Intervalos de confianza para los coeficientes de una regresión lineal.

Asumiendo que los errores del modelo de regresión lineal se distribuye en forma normal con media cero y varianza constante σ^2 y que además son independientes entre si, se puede llegar a establecer que un intervalo de confianza del 100(1-α)% para el coeficiente de regresión poblacional β_j es de la forma

$$(\hat{\beta}_j - t_{(\alpha/2)} \times se(\hat{\beta}_j), \hat{\beta}_j + t_{(1-\alpha/2)} \times se(\hat{\beta}_j)$$

donde $t_{(\alpha/2)}$ y $t_{(1-\alpha/2)}$ representa los percentiles del (α/2)100% y (1-α/2)100% de una distribución t con n-p-1 grados de libertad, donde p es el número de variables predictoras.
Todas las maneras de estimar intervalos de confianza por bootstrapping vistas anteriormente pueden ser aplicadas para hallar estimar intervalos de confianza para los coeficientes de un modelo de regresión. Aquí la suposición de normalidad ya no es requerida. Veremos solo algunas de ellas.

Métodos de los percentiles. Aquí simplemente se encuentran los percentiles del (α/2)100% y (1-α/2)100% de la distribución empírica acumulada de los estimados bootstrap $\hat{\beta}_j$. Luego el intervalo de confianza será de la forma

$$(\hat{F}^{-1}_{\hat{\beta}_j}(\alpha/2), \hat{F}^{-1}_{\hat{\beta}_j}(1-\alpha/2))$$

La siguiente función en R halla intervalos de confianza para los coeficientes de un modelo de regresión por el método de percentiles

```
boot.reg.ic.perc= function(X,conf,nboot)
{#Calcula intervalos de confianza por percentiles para los coeficientes de regresion
#usando bootstrapping en las observaciones
# level: nivel de confianza en porcentaje
# bint=intervalo de confianza
# nboot= numero de muestras bootstrap
nobs=dim(X)[1]
p=dim(X)[2]
alpha=1-.01*conf
y=X[,p]
xp=X[,-p]
betab=matrix(0,nboot,p)
for( k in 1:nboot)
{bootindex=sample(1:nobs,nobs,replace=T)
yb=X[bootindex,p]
xpb=X[bootindex,-p]
lm1=lm(yb~xpb)
betab[k,]=lm1$coeff
}
kperc=floor((nboot+1)*alpha*.5)
for( j in 1:p)
{
 betaboot=sort(betab[,j])
 tinf=betaboot[kperc]
```

```
  tsup=betaboot[nboot+1-kperc]

cat("\nEl intervalo de confianza para beta (",j-1,") es (",tinf,",",tsup,")\n")}
}
```

Ejemplo: Aplicar intervalo de confianza por percentiles a los datos millaje

```
boot.reg.ic.perc(as.matrix(milla1),95,1000)

El intervalo de confianza para beta ( 0 ) es ( 118.4289 , 257.8671 )

El intervalo de confianza para beta ( 1 ) es ( 0.1607931 , 0.6247816 )

El intervalo de confianza para beta ( 2 ) es ( -1.963997 , -0.570458 )

El intervalo de confianza para beta ( 3 ) es ( -2.493505 , -1.266671 )

El intervalo de confianza para beta ( 4 ) es ( -0.07411606 , 0.03582348 )
>
```

Peter Hall (1988, Annals of Statistics) mostró que el intervalo de confianza anterior es usualmente es sesgado y propuso la siguiente modificación.

$$(\hat{\beta}_j - [\hat{F}^{-1}_{\hat{\beta}^*_j}(1-\alpha/2) - \hat{\beta}_j], \hat{\beta}_j - [\hat{F}^{-1}_{\hat{\beta}^*_j}(\alpha/2) - \hat{\beta}_j])$$

El anterior hecho se justifica de la siguiente manera: Denotemos $\hat{\beta}^{*(\alpha/2)} = \hat{F}^{-1}_{\hat{\beta}^*}(\alpha/2)$ y $\hat{\beta}^{*(1-\alpha/2)} = \hat{F}^{-1}_{\hat{\beta}^*}(1-\alpha/2)$, donde por simplicidad se ha eliminado el índice j.

Consideremos,

$$\hat{\beta}^{(\alpha/2)} < \hat{\beta} < \hat{\beta}^{(1-\alpha/2)}$$

Luego,

$$\beta + (\hat{\beta}^{(\alpha/2)} - \beta) < \hat{\beta} < \beta + (\hat{\beta}^{(1-\alpha/2)} - \beta)$$

De donde,

$$-\beta - (\hat{\beta}^{(\alpha/2)} - \beta) > -\hat{\beta} > -\beta - (\hat{\beta}^{(1-\alpha/2)} - \beta)$$

Así,

$$\hat{\beta} - (\hat{\beta}^{(1-\alpha/2)} - \beta) < \beta < \hat{\beta} - (\hat{\beta}^{(\alpha/2)} - \beta)$$

Reemplazando por el estimador bootstrapping del percentil y el parámetro en cada extremo por su estimación correspondiente se obtiene el Intervalo de confianza

$$[\hat{\beta} - (\hat{\beta}^{*(1-\alpha/2)} - \hat{\beta}), \hat{\beta} - (\hat{\beta}^{*(\alpha/2)} - \hat{\beta})]]$$

Aplicando a los datos de millaje se obtiene

> boot.reg.ic.perc1(as.matrix(milla1),95,1000)

El intervalo de confianza para beta (0) es (121.6666 , 262.3008)

El intervalo de confianza para beta (1) es (0.1492236 , 0.6088384)

El intervalo de confianza para beta (2) es (-1.987184 , -0.576152)

El intervalo de confianza para beta (3) es (-2.429093 , -1.230867)

El intervalo de confianza para beta (4) es (-0.06648304 , 0.04360679)

Método de intervalos de confianza estudentizado

Aquí se estiman percentiles usando los valores de los coeficientes de regresión con sus desviaciones estándares en cada muestra bootstrap. Es decir,

$$t^* = \frac{\hat{\beta}_j^* - \hat{\beta}_j}{se(\hat{\beta}_j^*)}$$

el error estándar $se(\hat{\beta}_j)$ puede ser estimado de tres maneras:

a) usando el error estándar basado en la muestra original. Es decir, $se(\hat{\beta}_j) = \sqrt{c_{jj}}\, s$.

b) Usando la desviación estándar de la distribución de los valores bootstrap de $\hat{\beta}_j$.

c) Usando un rango interquartílico estándarizado, es decir $(Q_3-Q_1)/1.35$.

Los métodos b y c son estimadores más robustos.

El siguiente programa en R implementa el caso b, usando bootstrapping de las observaciones

```
boot.reg.ic=function(X,conf,nboot)
{#Calcula intervalos de confianza para los coeficientes de regresion
#usando bootstrapping en las observaciones
# level: nivel de confianza en porcentaje
# bint=intervalo de confianza
# nboot= numero de muestras bootstrap
nobs=dim(X)[1]
p=dim(X)[2]
alpha=1-.01*conf
y=X[,p]
xp=X[,-p]
beta0=lm(y~xp)$coef
betab=matrix(0,nboot,p)
tb=matrix(0,nboot,p)
```

```
for( k in 1:nboot)
{bootindex=sample(1:nobs,nobs,replace=T)
yb=X[bootindex,p]
xpb=X[bootindex,-p]
lm1=lm(yb~xpb)
betab[k,]=lm1$coeff
tb[k,]=(betab[k,]-beta0)/summary(lm1)$coeff[,2]
}
seboot=apply(betab,2,sd)
kperc=floor((nboot+1)*alpha*.5)
for( j in 1:p)
{
  pivott=sort(tb[,j])
  tinf=pivott[kperc]
  tsup=pivott[nboot+1-kperc]
  binf=beta0[j]+tinf*seboot[j]
bsup=beta0[j]+tsup*seboot[j]
cat("\nEl intervalo de confianza para beta (",j-1,") es (",binf,",",bsup,")\n")}
}
```

Ejemplo: Calcular usando boostrapping estudentizado los intervalos de confianza del 95% para los coeficientes de regresión de los datos de millaje y compararlos con los exactos.

Solución: Usando la función anterior se obtienen los siguientes resultados

El intervalo de confianza para beta (0) es (73.25095 , 296.0742)

El intervalo de confianza para beta (1) es (0.03027718 , 0.709605)

El intervalo de confianza para beta (2) es (-2.271097 , -0.1812522)

El intervalo de confianza para beta (3) es (-2.766586 , -0.8919825)

El intervalo de confianza para beta (4) es (-0.08375355 , 0.05121671)
>

En tanto que los valores exactos son:

```
> betahat
 (Intercept)          hp          sp          wt         vol
192.43775332   0.39221231  -1.29481848  -1.85980373  -0.01564501
> sebetahat
(Intercept)          hp          sp          wt         vol
23.53161326  0.08141236  0.24477349  0.21336303  0.02282527
>gl=summary(lm(mpg~.,data=milla1))$df[2]
> gl
[1]  77

> tcof=qt(.975,gl)
```

```
> tcof
 1.991254

> betahat-tcof[2]*sebetahat
 (Intercept)         hp         sp         wt         vol
145.58032499  0.23009960 -1.78222477 -2.28466379 -0.06109593
> betahat+tcof[2]*sebetahat
 (Intercept)         hp         sp         wt         vol
239.29518165  0.55432503 -0.80741219 -1.43494367  0.02980591
>
>#limites de confianza inferiores
>betahat-tcof[2]*sebetahat
 (Intercept)         hp         sp         wt         vol
145.58032499  0.23009960 -1.78222477 -2.28466379 -0.06109593
>#limites de confianza superiores
> betahat+tcof[2]*sebetahat
 (Intercept)         hp         sp         wt         vol
239.29518165  0.55432503 -0.80741219 -1.43494367  0.02980591
>
```

Notar que los intervalos de confianza por bootstrapping son bien amplios comparados con los exactos, debido a que la desviación estándar de los coeficientes estimada por bootstrapping es mucho mayor que su valor exacto.