Inferencia en el modelo de regresión lineal: intervalos de confianza

Contents

1 Intervalo de confianza para las β_i

Se quiere obtener un intervalo (LI, LS) tal que

$$P(LI \leq \beta_i \leq LS) = 1 - \alpha$$

A dicho intervalo se le llama *intervalo de confianza con nivel de confianza* $1 - \alpha$. Un intervalo de confianza para un parámetro se puede entender como **un rango de valores posibles para dicho parámetro**.

1.1 Con matrices de datos

Hemos visto que

$$\hat{\beta} \rightarrow N(\beta, \sigma^2 Q)$$

donde $Q = (X^T X)^{-1}$:

$$Q = \begin{bmatrix} q_{00} & q_{01} & \cdots & q_{0k} \\ q_{10} & q_{11} & \cdots & q_{1k} \\ \cdots & \cdots & \cdots & \cdots \\ q_{k0} & q_{k1} & \cdots & q_{kk} \end{bmatrix}$$

Es decir:

$$\begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \cdots \\ \hat{\beta}_k \end{bmatrix} \sim N \left(\begin{bmatrix} \beta_0 \\ \beta_1 \\ \cdots \\ \beta_k \end{bmatrix}, \sigma^2 \begin{bmatrix} q_{00} & q_{01} & \cdots & q_{0k} \\ q_{10} & q_{11} & \cdots & q_{1k} \\ \cdots & \cdots & \cdots & \cdots \\ q_{k0} & q_{k1} & \cdots & q_{kk} \end{bmatrix} \right)$$

Esto implica que:

$$\hat{\beta}_i \to N(\beta_i, \sigma^2 q_{ii}), \quad i = 0, 1, 2, \ldots, k$$

Aplicando las propiedades de la distribución normal

$$\frac{\hat{\beta}_i - \beta_i}{\sqrt{\sigma^2 q_{ii}}} \to N(0,1), \quad i = 0, 1, 2, \cdots, k.$$

Si se tiene en cuenta que:

$$\frac{(n-k-1)\hat{s}_R^2}{\sigma^2} \sim \chi^2_{n-k-1}$$

y de acuerdo con el Apendice: Distribución t-student, se puede obtener que

$$\frac{\hat{\beta}_i - \beta_i}{\sqrt{\hat{s}_R^2 q_{ii}}} \sim t_{n-k-1}, \quad i = 0, 1, 2, \cdots, k.$$

Al denominador se le suele llamar *desviación típica del estimador*, o *standard error del estimador*:

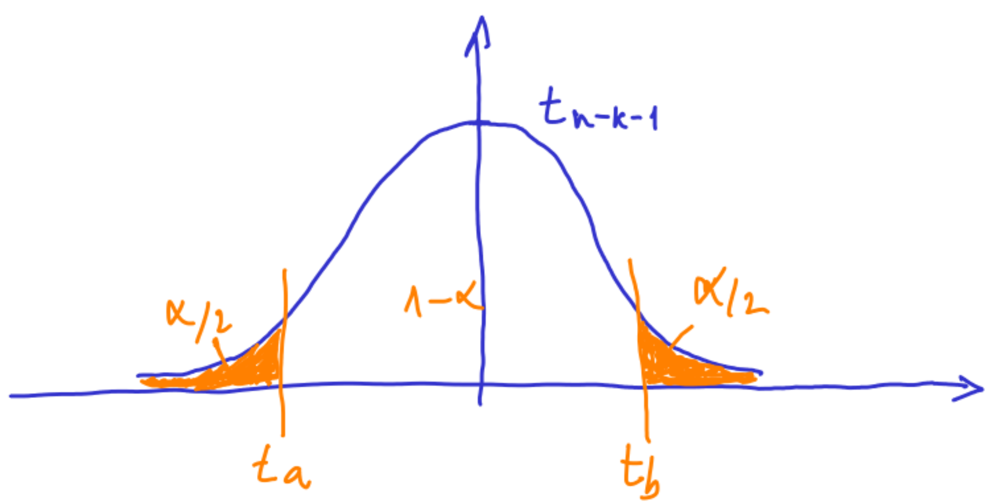
$$se(\hat{\beta}_i) = \sqrt{\hat{s}_R^2 q_{ii}}, \quad i = 0, 1, 2, \cdots, k.$$

Por lo que se puede escribir

$$\frac{\hat{\beta}_i - \beta_i}{se(\hat{\beta}_i)} \to t_{n-k-1}, \quad i = 0, 1, 2, \cdots, k.$$

Se eligen dos valores t_a y t_b que cumplen que:

$$P(t_a \leq t_{n-k-1} \leq t_b) = 1 - \alpha$$



Por tanto se tiene que cumplir que:

$$P\left(t_a \leq \frac{\hat{\beta}_i - \beta_i}{se(\hat{\beta}_i)} \leq t_b\right) = 1 - \alpha$$

Reordenando términos:

$$P\left(\hat{\beta}_i - t_b se(\hat{\beta}_i) \leq \beta_i \leq \hat{\beta}_i - t_a se(\hat{\beta}_i)\right) = 1 - \alpha$$

Como la distribución es simétrica se cumple que

$$-ta = tb = t_{\alpha/2}$$

$$\Rightarrow P\left(\hat{\beta}_i - t_{\alpha/2} se(\hat{\beta}_i) \leq \beta_i \leq \hat{\beta}_i + t_{\alpha/2} se(\hat{\beta}_i)\right) = 1 - \alpha$$

Comparando esta ecuación con $P(LI \leq \beta_i \leq LS) = 1 - \alpha$, se cumple que

$$LI = \hat{\beta}_i - t_{\alpha/2} se(\hat{\beta}_i)$$

$$LS = \hat{\beta}_i + t_{\alpha/2} se(\hat{\beta}_i)$$

Por tanto, el intervalo de confianza $100(1-\alpha)\%$ se obtiene como

$$\hat{\beta}_i \pm t_{\alpha/2} se(\hat{\beta}_i), \quad i = 0, 1, 2, \cdots, k.$$

1.2 Con matrices de covarianzas

Tenemos que

$$\hat{\beta}_a \to N(\beta_a, \sigma^2 Q_a)$$

donde

$$Q_a = \frac{1}{n-1} S_{XX}^{-1}$$

Esto implica que:

$$\hat{\beta}_i \to N(\beta_i, \sigma^2 Q_{a(i,i)}), \quad i = 1, 2, \ldots, k$$

donde $Q_{a(i,j)}$ es el elemento *(i,j)* de la matriz Q_a. Por tanto, siguiendo el razonamiento del apartado anterior:

$$se(\hat{\beta}_i) = \sqrt{\hat{s}_R^2 Q_{a(i,i)}}, \quad i = 1, 2, \cdots, k$$

Para $\hat{\beta}_0$ tenemos que

$$\hat{\beta}_0 \sim N\left(\beta_0, \sigma^2\left(\frac{1}{n} + \frac{1}{n-1}\bar{x}^T S_{XX}^{-1}\bar{x}\right)\right)$$

Por tanto

$$se(\hat{\beta}_0) = \sqrt{\hat{s}_R^2 \left(\frac{1}{n} + \frac{1}{n-1} \bar{x}^T S_{XX}^{-1} \bar{x} \right)}$$

Finalmente, el intervalo de confianza $100(1-\alpha)\%$ se obtiene como

$$\hat{\beta}_i \pm t_{n-k-1;\alpha/2} se(\hat{\beta}_i), \quad i = 0, 1, 2, \cdots, k.$$

2 Intervalo de confianza para σ^2

En este caso se quiere obtener un intervalo (LI, LS) tal que

$$P(LI \leq \sigma^2 \leq LS) = 1 - \alpha$$

Partimos de la distribución en el muestreo:

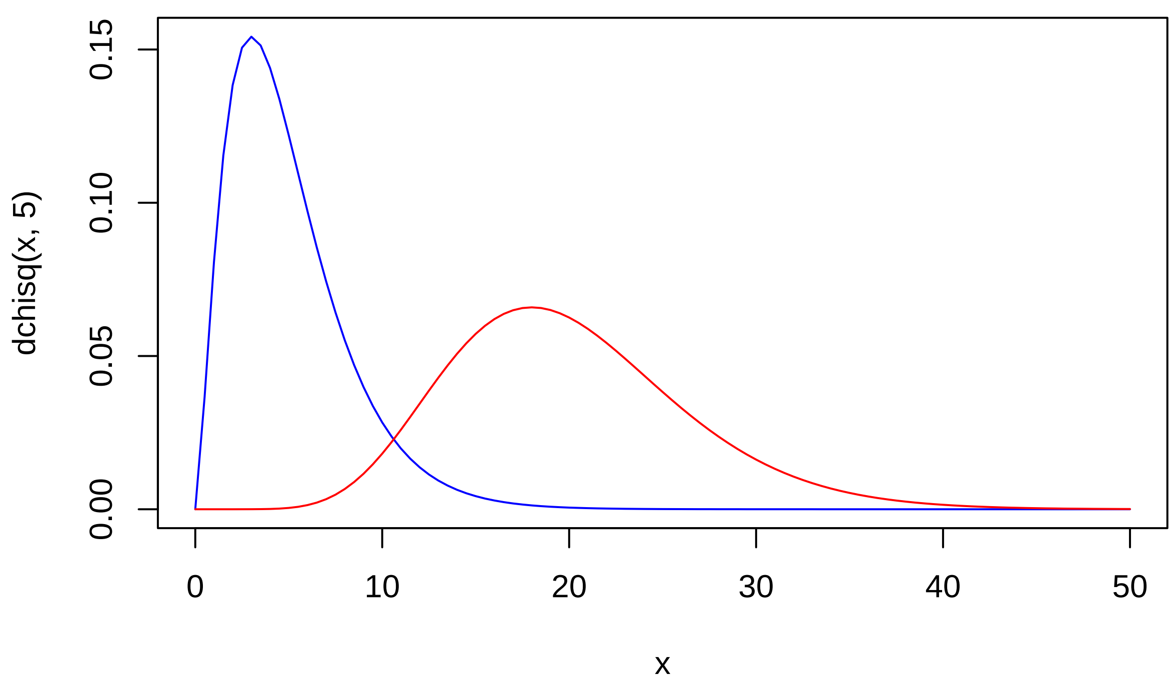
$$\frac{(n-k-1)\hat{s}_R^2}{\sigma^2} \to \chi^2_{n-k-1}$$

Despejando:

$$\frac{(n-k-1)\hat{s}_R^2}{\chi^2_{n-k-1;\alpha/2}} \leq \sigma^2 \leq \frac{(n-k-1)\hat{s}_R^2}{\chi^2_{n-k-1;1-\alpha/2}}$$

Podemos dibujar la distribución χ^2:

```
curve(dchisq(x,5), from = 0, to =50, col = "blue")
curve(dchisq(x,20), add = T, col = "red")
```



3 Ejemplo

Vamos a calcular de manera detallada los intervalos de confianza para el modelo $kid\_score \sim mom\_iq + mom\_hs$:

```
load("datos/kidiq.Rdata")
str(d)
```

```
## 'data.frame':    434 obs. of  5 variables:
##  $ kid_score: int  65 98 85 83 115 98 69 106 102 95 ...
##  $ mom_hs   : Factor w/ 2 levels "no","si": 2 2 2 2 2 1 2 2 2 2 ...
##  $ mom_iq   : num  121.1 89.4 115.4 99.4 92.7 ...
##  $ mom_work : Factor w/ 4 levels "notrabaja","trabaja23",..: 4 4 4 3 4 1 4 3 1 1 ...
##  $ mom_age  : int  27 25 27 25 27 18 20 23 24 19 ...
```

3.1 Varianza de $\hat{\beta}$

3.1.1 Usando matrices de datos

La varianza de los parámetros estimados es $var(\hat{\beta}_i) = \hat{s}_R^2 q_{i,i}$:

```
n = nrow(d)
# variable auxiliar
d$mom_hssi = ifelse(d$mom_hs == "si",1,0)
# matriz X
X = cbind(rep(1,n), d$mom_iq, d$mom_hssi)
Xt_X = crossprod(X) # crossprod es otra manera de calcular t(X) %*% X
Xt_y = crossprod(X,d$kid_score)
# parametros estimados
(beta = solve(Xt_X) %*% Xt_y)
```

```
##           [,1]
## [1,] 25.731538
## [2,]  0.563906
## [3,]  5.950117
```

```
# matriz Q = inv(t(X)*X)
(Q = solve(Xt_X) )
```

```
##               [,1]          [,2]          [,3]
## [1,]  0.1049491626 -1.025110e-03 -0.0001705848
## [2,] -0.0010251098  1.115594e-05 -0.0001151616
## [3,] -0.0001705848 -1.151616e-04  0.0148740410
```

La varianza residual es

$$\hat{s}_R^2 = \frac{\sum e_i^2}{n-k-1}$$

donde la suma de los residuos al cuadrado se puede calcular con la ecuación

$$\sum e_i^2 = y^T y - \hat{\beta}^T (X^T y)$$

```
(SRC = crossprod(d$kid_score) - t(beta) %*% Xt_y )
```

```
##          [,1]
## [1,] 141757.1
```

```
# numero de regresores
k = 2
(sR2 = SRC[1,1]/(n-k-1)) # [1,1] porque es mejor tener un numero que no una matriz de tamaño [1,1]
```

```
## [1] 328.9028
```

Por tanto, la matriz de varianzas de los estimadores será

```
(beta_var = sR2 * Q)
```

```
##             [,1]         [,2]        [,3]
## [1,] 34.51806922 -0.337161456 -0.05610582
## [2,] -0.33716146  0.003669219 -0.03787697
## [3,] -0.05610582 -0.037876974  4.89211314
```

Y el standard error de los estimadores, $se(\hat{\beta}_i)$:

```
(beta_se = sqrt(diag(beta_var)))
```

```
## [1] 5.87520802 0.06057408 2.21181218
```

3.1.2 Usando matrices de varianzas

Vamos a calcular ahora el standard error de los estimadores con la matriz de varianzas de los regresores:

```
Xa = cbind(d$mom_iq, d$mom_hssi)
(Qa = 1/(n-1)*solve(var(Xa)))
```

```
##               [,1]          [,2]
## [1,]  1.115594e-05 -0.0001151616
## [2,] -1.151616e-04  0.0148740410
```

El standard error de los estimadores $\hat{\beta}_1$ y $\hat{\beta}_2$ son:

```
sqrt(diag(Qa)*sR2)
```

```
## [1] 0.06057408 2.21181218
```

Para $\hat{\beta}_0$:

```
# vector con las medias de cada x
( xmed = matrix(colMeans(Xa), ncol = 1) )
```

```
##             [,1]
## [1,] 100.0000000
## [2,]   0.7857143
```

```
sqrt( sR2*(1/n + 1/(n-1)*t(xmed) %*% solve(var(Xa)) %*% xmed ) )
```

```
##          [,1]
## [1,] 5.875208
```

3.1.3 Con R

```
# Se estima el modelo
m = lm(kid_score ~ mom_iq + mom_hs, data = d)
```

```
# como curiosidad, la matriz X se puede obtener como
X1 = model.matrix(m)
# y los residuos
e = m$residuals
```

```r
# por lo que podríamos obtener la matriz de varianzas de beta como
(sum(e^2)/(n-k-1))*solve(crossprod(X1))
```

```
##             (Intercept)       mom_iq    mom_hssi
## (Intercept) 34.51806922 -0.337161456 -0.05610582
## mom_iq      -0.33716146  0.003669219 -0.03787697
## mom_hssi    -0.05610582 -0.037876974  4.89211314
```

Sin embargo R dispone de una función para calcular directamente esa matriz:

```r
vcov(m)
```

```
##             (Intercept)       mom_iq    mom_hssi
## (Intercept) 34.51806922 -0.337161456 -0.05610582
## mom_iq      -0.33716146  0.003669219 -0.03787697
## mom_hssi    -0.05610582 -0.037876974  4.89211314
```

Por tanto, el standard error de los estimadores será

```r
sqrt(diag(vcov(m)))
```

```
## (Intercept)      mom_iq    mom_hssi
##  5.87520802  0.06057408  2.21181218
```

Como vemos, los tres métodos dan el mismo resultado.

3.2 Intervalo de confianza para β_i

El valor de la t con n-k-1 = 431 grados de libertad es

```r
(ta = qt(1-0.05/2, df = n-k-1))
```

```
## [1] 1.965483
```

El límite inferior (LI) y el límite superior de los intervalos será:

```r
(LI = coef(m) - qt(1-0.05/2, df = n-k-1)*beta_se)
```

```
## (Intercept)      mom_iq    mom_hssi
##  14.1839148   0.4448487   1.6028370
```

```r
(LS = coef(m) + qt(1-0.05/2, df = n-k-1)*beta_se)
```

```
## (Intercept)      mom_iq    mom_hssi
##  37.2791615   0.6829634  10.2973969
```

Si lo juntamos todo en una tabla

```r
data.frame(estimacion = coef(m), se = beta_se, LI, LS)
```

```
##             estimacion         se         LI         LS
## (Intercept)  25.731538 5.87520802 14.1839148 37.2791615
## mom_iq        0.563906 0.06057408  0.4448487  0.6829634
## mom_hssi      5.950117 2.21181218  1.6028370 10.2973969
```

Directamente, mediante la función *confint()* de R se pueden obtener dichos valores:

```r
confint(m)
```

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

```
## mom_hssi     1.6028370 10.2973969
```

Si queremos otro nivel de confianza, por ejemplo, 90%:

```
confint(m, level = 0.90)
```

```
##                    5 %       95 %
## (Intercept) 16.0468646 35.4162117
## mom_iq       0.4640559  0.6637562
## mom_hssi     2.3041730  9.5960608
```

3.3 Intervalo de confianza para σ^2

El estimador de la varianza es:

```
sR2
```

```
## [1] 328.9028
```

Y su intervalo de confianza:

```
c((n-k-1)*sR2/qchisq(1-0.05/2, df = n-k-1), (n-k-1)*sR2/qchisq(0.05/2, df = n-k-1))
```

```
## [1] 289.0557 377.6434
```