Estimación del modelo para los datos del ejemplo

Contents

1 Introducción

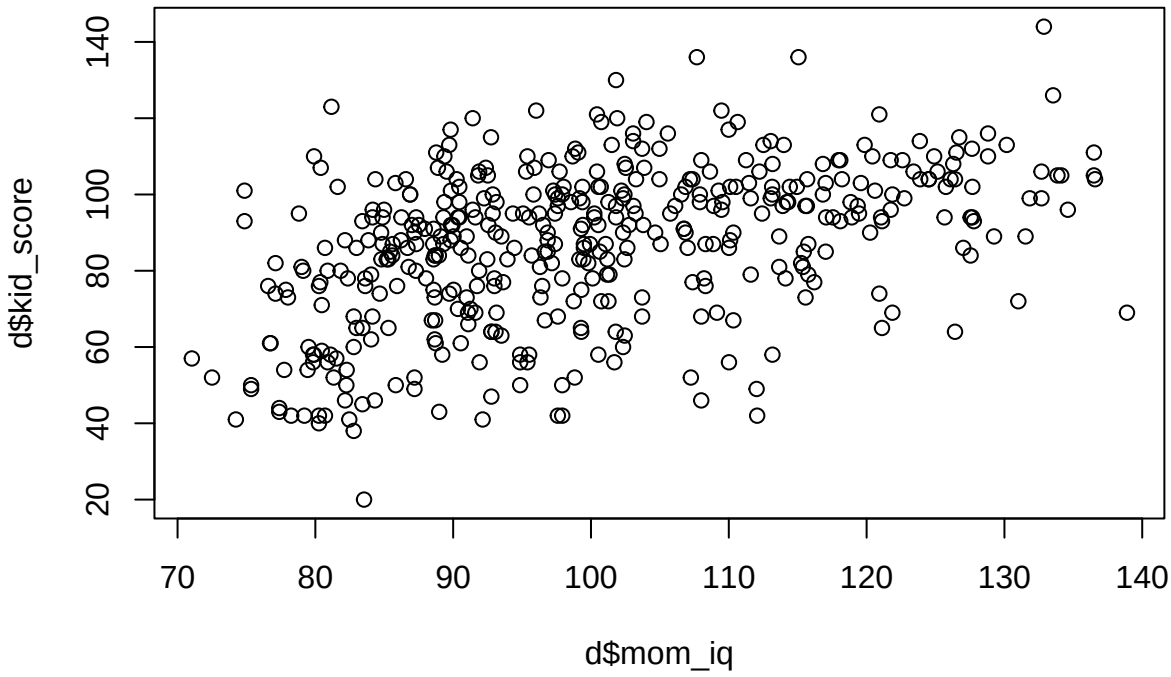
El primer paso es leer los datos correspondientes:

```
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 ...
```

Se quiere estudiar si la puntuación obtenida por los niños (variable *kid_score*) está relacionada con la puntuación obtenida por las madres (*mom_iq*) y su edad (*mom_age*). Primero se dibuja el gráfico de dispersión:

```
plot(d$mom_iq, d$kid_score)
```



Como se observa, en términos generales cuando mayor es la puntuación obtenida por las madres mayor es la puntuación de los niños.

2 Ecuación del modelo

El modelo más sencillo que relaciona ambas variables es el modelo lineal:

$$kid\_score_i = \beta_0 + \beta_1 mom\_iq_i + \beta_2 mom\_age_i + u_i,\ u_i \sim N(0, \sigma^2),\ i = 1, 2, \cdots, n$$

Ya se ha visto que el modelo se puede escribir matricialmente:

$$kid\_score_i = \beta_0 + \beta_1 mom\_iq_i + \beta_2 mom\_age_i + u_i,\ i = 1, 2, \cdots, n$$

Si escribimos la ecuación para todos los datos disponibles:

$$i = 1 \Rightarrow kid\_score_1 = \beta_0 + \beta_1 mom\_iq_1 + \beta_2 mom\_age_1 + u_1$$

$$i = 2 \Rightarrow kid\_score_2 = \beta_0 + \beta_1 mom\_iq_2 + u_2$$

$$\cdots$$

$$i = n \Rightarrow kid\_score_n = \beta_0 + \beta_1 mom\_iq_n + \beta_2 mom\_age_n + u_n$$

Agrupando:

$$\begin{bmatrix} kid\_score_1 \\ kid\_score_2 \\ \cdots \\ kid\_score_n \end{bmatrix} = \begin{bmatrix} 1 & mom\_iq_1 & \beta_2 mom\_age_1 \\ 1 & mom\_iq_2 & \beta_2 mom\_age_2 \\ \cdots & \cdots & \\ 1 & mom\_iq_n & \beta_2 mom\_age_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_3 \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \\ \cdots \\ u_n \end{bmatrix}$$

Finalmente, en notación matricial:

$$y = X\beta + u$$

Los residuos cumplen que

$$kid\_score_i = \hat{\beta}_0 + \hat{\beta}_1 mom\_iq_i + \beta_2 mom\_age_i + e_i,\ i = 1, 2, \cdots, n$$

es decir

$$e_i = y_i - \hat{y}_i,\ i = 1, 2, \ldots, n$$

o en forma matricial

$$e = y - \hat{y}$$

donde

$$\hat{y} = XB$$

3 Estimación usando matrices de datos

- Matrices del modelo

```
y = matrix(d$kid_score, ncol = 1)
head(y)
```

```
##      [,1]
## [1,]   65
## [2,]   98
## [3,]   85
## [4,]   83
## [5,]  115
## [6,]   98
```

```
n = nrow(d)
X = cbind(rep(1,n), d$mom_iq, d$mom_age)
head(X)
```

```
##      [,1]      [,2] [,3]
## [1,]    1 121.11753   27
## [2,]    1  89.36188   25
## [3,]    1 115.44316   27
## [4,]    1  99.44964   25
## [5,]    1  92.74571   27
## [6,]    1 107.90184   18
```

- Estimacion

```
Xt_X = t(X) %*% X
Xt_y = t(X) %*% y
( B = solve(Xt_X) %*% Xt_y )
```

```
##             [,1]
## [1,] 17.5962491
## [2,]  0.6035720
## [3,]  0.3881286
```

- valores de la recta

```
y_e = X %*% B
```

Los residuos se calculan haciendo

```
e = y - y_e
```

4 Bondad del modelo ajustado

Es conveniente medir como de bueno es el ajuste del modelo. Una posibilidad es usar la suma de los residuos al cuadrado o SRC:

```
(SRC = sum(e^2))
```

```
## [1] 143665.4
```

Vamos a comprobar que se cumple la fórmula:

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

```
sum(d$kid_score^2) - t(B) %*% Xt_y
```

```
##          [,1]
## [1,] 143665.4
```

Pero esta variable depende de las unidades de x e y. Por tanto es difícil saber si un SRC alto indica que el modelo es bueno o malo. Lo ideal es utilizar variables adimensionales. La manera mas usual es utilizar el coeficiente de determinación o R^2:

$$R^2 = 1 - \frac{SRC}{STC}$$

donde STC es la suma total de cuadrados

$$STC = \sum (y_i - \bar{y})^2$$

```
(STC = sum((y-mean(y))^2))
```

```
## [1] 180386.2
```

```
(R2 = 1 - SRC/STC)
```

```
## [1] 0.2035673
```

El coeficiente R^2 toma valores entre cero y uno. Si $R^2 \approx 1 \Rightarrow SRC \ll STC$, es decir, los residuos son muy pequeños en comparación a los datos, luego el modelo se ajusta muy bien a los datos. Cuando $R^2 \approx 0$, los residuos son muy grandes y el modelo no se ajusta bien a los datos.

La suma total de cuadrados de y está relacionado con su varianza, ya que

$$s_y^2 = \frac{\sum(y_i - \bar{y})^2}{n-1} \Rightarrow STC = (n-1)s_y^2$$

```
(n-1)*var(y)
```

```
##          [,1]
## [1,] 180386.2
```

5 Estimación usando matrices de varianzas

```
# en primer lugar vamos a calcular las matrices de covarianzas con la función de R cov()
(S = cov(d[,c(1,3,5)]) )
```

```
##            kid_score    mom_iq  mom_age
## kid_score 416.596205 137.24428 5.071923
## mom_iq    137.244279 225.00000 3.711610
## mom_age     5.071923   3.71161 7.295777
```

```
(Sxx = S[2:3,2:3])
```

```
##            mom_iq  mom_age
## mom_iq  225.00000 3.711610
## mom_age   3.71161 7.295777
```

```
(Sxy = S[2:3,1] )
```

```
##     mom_iq    mom_age
## 137.244279   5.071923
```

```
(Ba = solve(Sxx) %*% Sxy)
```

```
##              [,1]
## mom_iq  0.6035720
## mom_age 0.3881286
```

Vamos a comprobar que las matrices de covarianzas se pueden calcular a partirde X_a e Y_a:

```
ya = matrix(d$kid_score - mean(d$kid_score), ncol = 1)
Xa = cbind(d$mom_iq - mean(d$mom_iq), d$mom_age - mean(d$mom_age)) # sin columna de unos!!!!
```

```
n = nrow(d)
1/(n-1) * t(Xa) %*% Xa
```

```
##           [,1]     [,2]
## [1,] 225.00000 3.711610
## [2,]   3.71161 7.295777
```

```
1/(n-1) * t(Xa) %*% ya
```

```
##            [,1]
## [1,] 137.244279
## [2,]   5.071923
```

Falta calcular b_0. Utilizamos la fórmula

$$b_0 = \bar{y} - (\hat{\beta}_1\bar{x}_1 + \hat{\beta}_2\bar{x}_2 + \cdots + \hat{\beta}_k\bar{x}_k)$$

```r
( b0 = mean(d$kid_score) - colMeans(d[,c(3,5)]) %*% Ba )
```

```
##          [,1]
## [1,] 17.59625
```

Por último, para la suma de los residuos al cuadrado se tiene que cumplir que:

$$\sum e_i^2 = (n-1)s_y^2 - (n-1)\hat{\beta}_a^T S_{Xy}$$

```r
# varianza de y
(sy2 = S[1,1] )
```

```
## [1] 416.5962
```

```r
(SRC = (n-1)*sy2 - (n-1)*t(Ba) %*% Sxy)
```

```
##          [,1]
## [1,] 143665.4
```