Estimación del modelo con la función lm()

Contents

1 La función *lm()*

Para estimar modelos lineales en R se utiliza la función *lm()*, de *linear models*:

```
d = read.csv("datos/kidiq.csv")
```

```
m = lm(kid_score ~ mom_iq + mom_age, data = d)
```

El resultado del análisis se ha guardado en la variable m:

- Matriz X:

```
X = model.matrix(m)
head(X)
```

```
##   (Intercept)    mom_iq mom_age
## 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
```

- Parámetros estimados B:

```
coefficients(m)
```

```
## (Intercept)      mom_iq     mom_age
##  17.5962491   0.6035720   0.3881286
```

- Valores estimados por el modelo $\hat{y}$:

```
y_e = fitted(m)
head(y_e)
```

```
##         1         2         3         4         5         6
## 101.17887  81.23579  97.75398  87.32448  84.05443  89.70909
```

- Residuos e_i

```
e = residuals(m)
head(e)
```

```
##          1          2          3          4          5          6 
## -36.17887  16.76421 -12.75398  -4.32448  30.94557   8.29091
```

- RSS

```r
deviance(m)
```

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

Los valores anteriores también se pueden obtener con el símbolo $:

```r
m$coef
```

```
## (Intercept)      mom_iq     mom_age 
##  17.5962491   0.6035720   0.3881286
```

También se puede utilizar la función *summary()*, que además de los valores anteriores proporciona otros muchos que se irán viendo en los temas siguientes:

```r
summary(m)
```

```
## 
## Call:
## lm(formula = kid_score ~ mom_iq + mom_age, data = d)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -56.941 -12.493   2.257  11.614  46.711 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 17.59625    9.08397   1.937   0.0534 .  
## mom_iq       0.60357    0.05874  10.275   <2e-16 ***
## mom_age      0.38813    0.32620   1.190   0.2348    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 18.26 on 431 degrees of freedom
## Multiple R-squared:  0.2036, Adjusted R-squared:  0.1999 
## F-statistic: 55.08 on 2 and 431 DF,  p-value: < 2.2e-16
```

El resultado de summary también se puede guardar en una variable para tener, por ejemplo, el R2:

```r
m_summ = summary(m)
m_summ$r.squared
```

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

2 Regresión lineal sin ordenada en el origen

El modelo que queremos estimar es

$$y_i = b_1 x_{1i} + b_2 x_{2i} + b_3 x_{3i} + e_i, \ \ i = 1, 2, \cdots, n \tag{1}$$

es decir, tenemos que $b_0 = 0$. En forma matricial tendríamos $y = XB + e$, donde:

$$X = \begin{bmatrix} x_{11} & x_{21} & x_{31} \\ x_{12} & x_{22} & x_{32} \\ \cdots & \cdots & \cdots \\ x_{1n} & x_{2n} & x_{3n} \end{bmatrix}, \; B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{2}$$

En R:

```
y = matrix(d$kid_score, ncol = 1)
X = cbind(d$mom_iq, d$mom_age)

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

```
##           [,1]
## [1,] 0.6686188
## [2,] 0.8677182
```

Con la función *lm()*, este modelo se estima añadiendo un cero en la declaración de los regresores:

```
m = lm(kid_score ~ 0 + mom_iq + mom_age, data = d)
summary(m)
```

```
##
## Call:
## lm(formula = kid_score ~ 0 + mom_iq + mom_age, data = d)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -55.809 -12.076   2.487  12.310  47.515
##
## Coefficients:
##         Estimate Std. Error t value Pr(>|t|)
## mom_iq   0.66862    0.04835  13.829  < 2e-16 ***
## mom_age  0.86772    0.21307   4.073 5.53e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.32 on 432 degrees of freedom
## Multiple R-squared:  0.958,  Adjusted R-squared:  0.9578
## F-statistic:  4926 on 2 and 432 DF,  p-value: < 2.2e-16
```

Otra manera de indicarlo es:

```
m = lm(kid_score ~ -1 + mom_iq + mom_age, data = d)
summary(m)
```

```
##
## Call:
## lm(formula = kid_score ~ -1 + mom_iq + mom_age, data = d)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -55.809 -12.076   2.487  12.310  47.515
##
## Coefficients:
```

```
##         Estimate Std. Error t value Pr(>|t|)
## mom_iq   0.66862    0.04835  13.829  < 2e-16 ***
## mom_age  0.86772    0.21307   4.073 5.53e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.32 on 432 degrees of freedom
## Multiple R-squared:  0.958,  Adjusted R-squared:  0.9578
## F-statistic:  4926 on 2 and 432 DF,  p-value: < 2.2e-16
```

Es curioso el hecho de que R^2 ha aumentado y alcanza un valor cercano a uno. Esto es debido a que en este tipo de modelos la fórmula que emplea R para calcular R^2 es:

$$R_0^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum y_i^2} \tag{3}$$

en lugar de

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \tag{4}$$

Podemos calcular esos dos valores *a mano* para comprobarlo:

```
1 - sum(m$resid^2)/sum(d$kid_score^2)
```

```
## [1] 0.9579958
```

```
1 - sum(m$resid^2)/sum((d$kid_score - mean(d$kid_score))^2)
```

```
## [1] 0.1966337
```

Por tanto, el modelo sin ordenada en el origen tiene menor R^2.

3 Formulas y expresiones en la función lm()

Modelo *lineal* hace referencia a que la ecuación del modelo es una función lineal de los parámetros β_0, β_1, β_2, $\cdots$, β_k. Modelos en apariencia complicados pueden ser considerados como un modelo lineal. Por ejemplo:

- polinomios:

$$y = b_0 + b_1 x + b_2 x^2 + e \Rightarrow y = b_0 + b_1 x_1 + b_2 x_2 \tag{5}$$

- modelos con funciones en los regresores

$$y = b_0 + b_1 x + b_2 log x + e \Rightarrow y = b_0 + b_1 x_1 + b_2 x_2 \tag{6}$$

- modelos con interacción:

$$y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_1 x_2 + e \tag{7}$$

- este modelo no es lineal

$$y = b_0 + b_1 x^{b_2} + e \tag{8}$$

En este apartado se va a estudiar como introducir estas regresiones lineales en R. Por ejemplo, queremos realizar la siguiente regresión:

$$y_i = b_0 + b_1(x_{1i} + x_{2i}) + e_i \tag{9}$$

En R se expresa mediante el operador I():

```
y ~ I(x1 + x2)
```

ya que la expresión:

```
y ~ x1 + x2
```

corresponde al modelo:

$$y_i = b_0 + b_1 x_{1i} + b_2 x_{2i} + e_i \tag{10}$$

Otro ejemplo es el modelo:

$$y_i = b_0 + b_1 x_{1i} + b_2 x_{1i}^2 + e_i \tag{11}$$

En R este modelo se expresa utilizando:

```
y ~ x1 + I(x2^2)
```

ya que la expresión

```
y ~ x1 + x2^2
```

significa interacción y no el cuadrado del regresor.

Es frecuente incluir funciones matemáticas en el modelo de regresión. Por ejemplo:

$$\log(y_i) = b_0 + b_1 x_{1i} + b_2 e^{x_{2i}} + e_i \tag{12}$$

En R, este modelo se indica:

```
log(y) ~ x1 + exp(x2)
```

3.1 Ejemplo

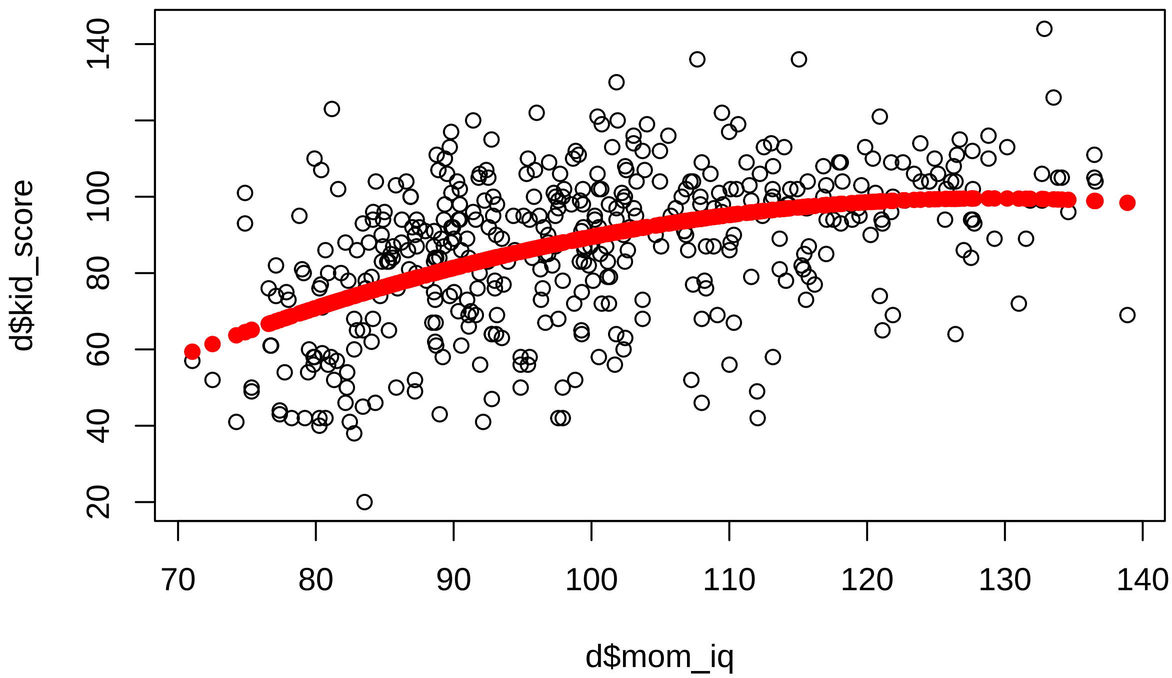
```
m = lm(kid_score ~ mom_iq + I(mom_iq^2), data = d)
summary(m)
```

```
##
## Call:
## lm(formula = kid_score ~ mom_iq + I(mom_iq^2), data = d)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -54.824 -11.640   2.883  11.372  50.813
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -99.033675  37.301385  -2.655 0.008226 **
## mom_iq        3.076800   0.730291   4.213 3.07e-05 ***
## I(mom_iq^2)  -0.011917   0.003517  -3.389 0.000767 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.05 on 431 degrees of freedom
## Multiple R-squared:  0.2217, Adjusted R-squared:  0.2181
## F-statistic: 61.38 on 2 and 431 DF,  p-value: < 2.2e-16
```

Podemos representar el modelo estimado haciendo

```r
plot(d$mom_iq, d$kid_score)
points(d$mom_iq, fitted.values(m), col = "red", pch = 19)
```



Es conveniente recordar que a pesar de que hay un término cuadrático se trata de un modelo lineal ya que el modelo que se estima es en realidad

$$kid\_score_i = b_0 + b_1 mom\_iq_i + b_2 z_i + e_i \tag{13}$$

donde $z_i = mom\_iq_i^2$.