UNIVERSIDAD DE EL SALVADOR

FACULTAD MULDISCIPLINARIA DE OCCIDENTE

DEPARTAMENTO DE MATEMÁTICAS

LICENCIATURA EN ESTADÍSTICA

PRACTICAS REALIZADAS EN EL SOFTWARE R

DOCENTE:
LICENCIADO. JAIME ISAAC PEÑA

PRESENTADO POR:
MORIS SALVADOR HENRIQUEZ LIMA

Viernes 21 de Octubre del 2022

Índice

1. REGRESIÓN LINEAL SIMPLE

Los modelos de regresión lineal son modelos probabilísticos basados en una función lineal, expresamos el valor de nuestra variable de estudio (interés), a la que también llamamos variable dependiente, en función de una o más variables a quienes llamamos variables independientes o explicativas, y las cuales suponemos tienen un efecto sobre nuestra variable de estudio. Los pasos básicos a seguir en el estudio de un modelo lineal son:

- Escribir el modelo matemático con todas sus hipótesis.
- Estimación de los parámetros del modelo.
- Inferencias sobre los parámetros.
- Diagnóstico del modelo.

El modelo de regresión más simple que nos podemos encontrar es aquel en donde únicamente se considera a solamente una variable independiente, y se quiere estudiar su efecto sobre la variable dependiente; la ecuación del modelo es:

$$y_i = \beta_0 + \beta_1 x_i + u_i$$

Donde:

- y_i ; representa la observación i-ésima correspondiente de la variable dependiente, es decir, el valor de la variable dependiente para el i-ésimo individuo de la muestra.
- x_i ; representa la observación i-ésima correspondiente de la variable independiente.
- β_0 ; representa el intercepto del modelo, es decir, valor de la variable dependiente cuando nuestra variable independiente toma el valor de cero. En muchos casos no tendrá interpretación, pues la variable independiente no puede tomar el valor de 0.
- β_1 ;representa la pendiente del modelo, es decir, el cambio esperado en la variable dependiente por cada cambio unitario realizado a la variable independiente.
- u_i ; representa el efecto de las demás variables omitidas en el modelo.

Las hipótesis básicas del modelo, son las mismas a las consideradas en el Análisis de Varianza, que como recordarán son las siguientes:

- El promedio de las perturbaciones es cero, es decir, se cumple que:
  $E[u_i] = ; \forall_i$
- La varianza de las perturbaciones es constante, es decir, se cumple que:
  $var(u_i) = \sigma^2 \, ; \, \forall_i$
- La distribución de las perturbaciones debe ser normal, es decir se cumple que:
  $u_i \approx N(0; \sigma^2) \, ; \, \forall_i$
- Las perturbaciones son independientes, es decir se cumple que:
  $cov(u_i; u_j) = 0 \, ; \, \forall_i \neq j$

Las cuales pueden resumirse en: $u_i \sim NIID(0, \sigma^2); \forall_i$

En R la función a utilizar para realizar o ajustar un modelo de regresión es lm() (de lineal model). Esta función no nos ofrece ninguna salida en pantalla si no que nos crea un objeto, o mejor dicho, nosotros creamos un objeto que va a ser un modelo de regresión lineal, y el cual podemos referenciarlo posteriormente en nuestro análisis.

La función lm tiene la siguiente sintaxis:
lm(formula, data, subset)

- En formula escribimos: y $\sim$ x , lo cual significa que a la izquierda del símbolo $\sim$ especificamos quien es nuestra variable dependiente; mientras que a la derecha especificamos quien es nuestra variable independiente.
- En data especificamos el dataframe que contiene las variables del modelo, es recomendable que los datos se encuentren en un dataframe.
- En subset especificamos un subconjunto de observaciones para validar posteriormente el modelo. En caso que se desee utilizar conjuntos distinto para estimar y validar el modelo. Muy recomendado en muchas aplicaciones.

La función lm tiene muchas más opciones pero para conocer mejor su funcionamiento vamos a ver ejemplos.

1.1. EJEMPLO 1.

En el archivo "costes.dat" se encuentra la información correspondiente a 34 fábricas de producción en el montaje de placas para ordenador, el archivo contiene la información sobre el costo total (primera columna) y el número de unidades fabricadas (segunda columna). Suponga que deseamos ajustar un modelo de regresión simple a los datos para estimar el costo total en función del número de unidades fabricadas.

Ejecutamos lo siguiente:

Lectura de los datos:

```
> getwd()

[1] "C:/Moris_Henriquez/Practicas_R_Sweave_2022"

> Datos = read.table("costes.txt")
> Datos

           V1   V2
1   58.421666  482
2  179.000965 1154
3  126.396460  518
4   35.513232  145
5   52.638089  173
6   79.001568  175
7  123.304684  643
8   45.603328  143
9  203.540000  670
10  29.248519   95
11 151.155602  877
12  82.633085  217
13  72.079808  340
14  65.292189  605
```

```
15 230.495693 1198
16 251.549754 1738
17  39.187287  204
18  20.238405  192
19  11.599884  172
20 122.277107  880
21  81.340000 1917
22  90.972481 1026
23  13.281056  234
24  45.101104  370
25 145.766077  836
26 143.366143  823
27  32.749009  242
28  52.740539  108
29  -6.688746  347
30  53.857050  208
31  63.248448  380
32  81.309927  824
33  76.334521  133
34  38.475459  275
35  79.012335  149
```

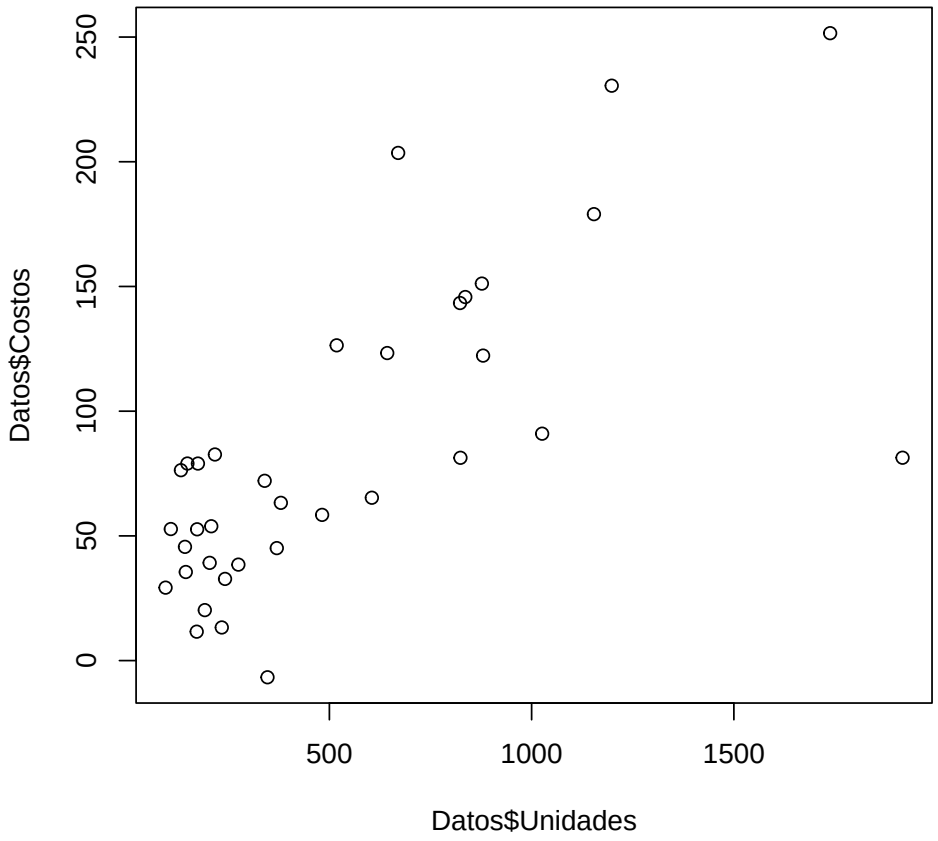
Renombrando las variables:

```
> names(Datos)=c("Costos","Unidades")
> head(Datos)

     Costos Unidades
1  58.42167      482
2 179.00096     1154
3 126.39646      518
4  35.51323      145
5  52.63809      173
6  79.00157      175
```

Diagrama de dispersion entre las dos variables:

```
> plot(Datos$Unidades,Datos$Costos)
```



Se aprecia una relación entre las variables por lo que se procede a ajustar el modelo de regresión:

```
> regresion <- lm(Datos$Costos ~ Datos$Unidades)
> summary(regresion)

Call:
lm(formula = Datos$Costos ~ Datos$Unidades)

Residuals:
     Min       1Q   Median       3Q      Max
-137.386  -24.496   -0.117   29.848  105.028

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    33.92200   11.57500   2.931   0.0061 **
Datos$Unidades  0.09640    0.01665   5.789  1.8e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 44.49 on 33 degrees of freedom
Multiple R-squared:  0.5039,        Adjusted R-squared:  0.4888
F-statistic: 33.51 on 1 and 33 DF,  p-value: 1.796e-06
```

En este caso el modelo resultante sería:
costos = 33.51

Con los resultados obtenidos tanto con el p-valor como el modelo, son resultados no significativos, por lo tanto, volvemos a realizar los cálculos con el software R:

```
> regresion2 <- lm(Datos$Costos ~ Datos$Unidades -1)
> summary(regresion2)

Call:
lm(formula = Datos$Costos ~ Datos$Unidades - 1)

Residuals:
     Min       1Q   Median       3Q      Max
-174.579   -4.844   19.527   35.812  114.095

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
Datos$Unidades  0.13350    0.01197   11.16 6.59e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 49.21 on 34 degrees of freedom
Multiple R-squared:  0.7854,        Adjusted R-squared:  0.7791
F-statistic: 124.5 on 1 and 34 DF,  p-value: 6.591e-13
```

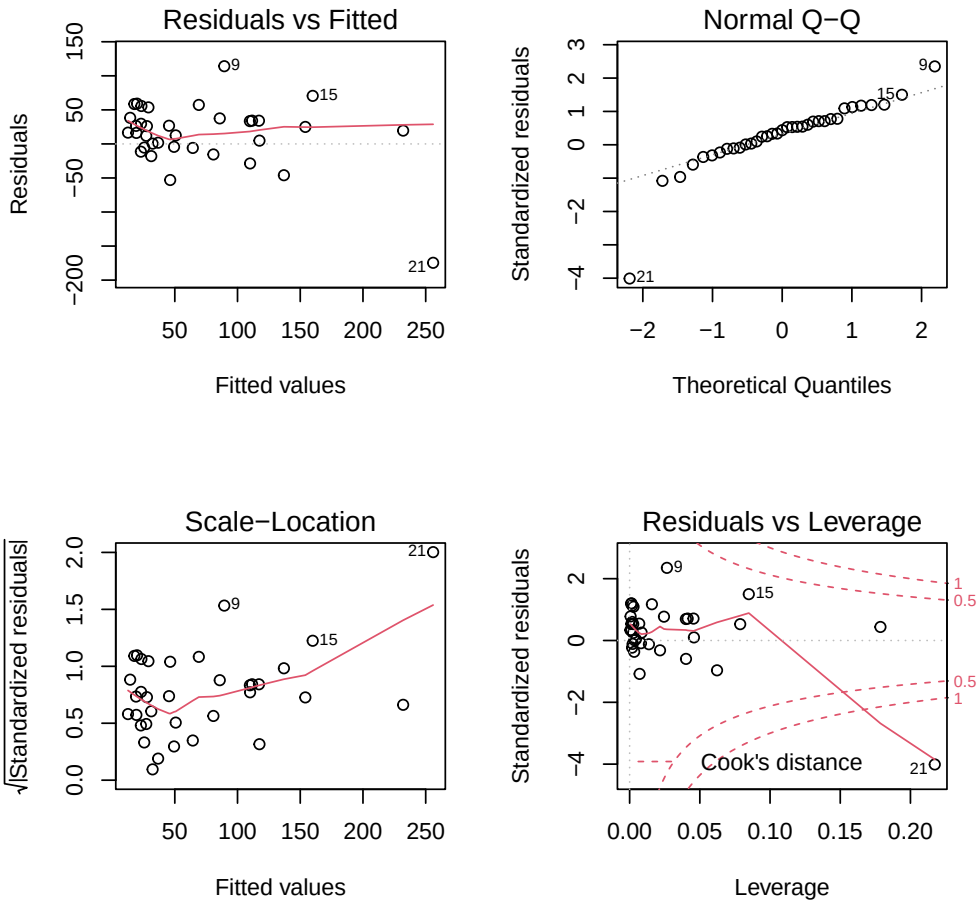
En este caso el modelo resultante sería: costos = 0.1335(unidades); el cual es un mejor modelo en términos de variabilidad explicada.

Una vez estimados los parámetros del modelo, el siguiente paso es validarlo, es decir verificar si se cumplen las cuatro hipótesis básicas del modelo (nulidad, normalidad, independencia y homocesdasticidad de los residuos). Para verificar esto, podríamos realizar los siguientes pasos:

Efectúa un análisis gráfico de bondad de ajuste del modelo

```
> par(mfrow = c(2, 2))
> plot(regresion2)
> par(oma=c(1,1,1,1), new=T, font=2, cex=0.5)
> mtext(outer=T, "Gráficos para validación del modelo: Costos en función de las unidades",
+ side=3)
```

Gráficos para validación del modelo: Costos en función de las unidades



En los gráficos que se muestra en la parte superior se contrasta los cuatro supuestos. En el de la izquierda se verifican: nulidad, independencia y homocedasticidad; a partir del gráfico mostrado parece existir indicios de falta de homocedasticidad, por su parte los residuos pueden considerarse constante pues no muestran ningún patrón; sin embargo, la media de los residuos no parece ser nula, lo cual indica falta de linealidad en el modelo (es decir, es necesario incorporar más variables o tal vez términos cuadráticos). En la figura de la derecha se contrasta la normalidad, y puede apreciarse que los residuos parecen seguir una distribución normal.

Por su parte, también es de mencionar que en el gráfico se muestran puntos que posiblemente sean observaciones atípicas, por lo que habría que estudiarlas.

Información sobre el modelo ajustado que proporciona la función lm():

- **formula(regresion2)**: Extrae la fórmula del modelo.
- **coef(regresion2)**: Extrae el vector de coeficientes de regresión.
- **residuals(regresion2)**: Extrae el vector de residuos.
- **modelo2ted.values(regresion2)**: Extrae un vector con los valores estimados.
- **vcov(regresion2)**: Extrae la matriz de covarianzas de los parámetros.
- **ls.diag(regresion2)**: Calcula los residuales, errores estándar de los parámetros, distancias Cook.

- **step(regresion2)**: Permite obtener el mejor conjunto de regresión y proporciona la estimación de los coeficientes (válido únicamente en modelos de regresión múltiple).

De todos los resultados anteriores nos concentraremos en la instrucción: ls.diag(regresion2). Con esta instrucción obtenemos para cada observación en el conjunto de datos, medidas que nos ayudarán a identificar observación atípicas (tienen un impacto únicamente en las medidas resumen del modelo) y observaciones influyentes (tienen un efecto marcado en la estimación de los parámetros).

2. REGRESIÓN LINEAL MÚLTIPLE

Al igual que en el modelo de regresión simple, el modelo de regresión múltiple trata de ajustar una ecuación matemática en la que se relacione a una única variable dependiente en función de dos o más variables independientes. La forma general del modelo es la siguiente:

$$Y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_k x_{ki} + \mu_i$$

Como siempre debe cumplirse que:

$$\mu_i \sim NIID(0, \sigma^2); \forall i$$

La función para estimar cada uno de los parámetros del modelo, a partir de la información suministrada por la muestra, los datos disponibles, es como siempre lm(), sin embargo, en la expresión fórmula debemos escribir $y \sim x_1 + x_2 + ... + x_k$. Todas las instrucciones utilizadas en regresión simple son válidas también para regresión múltiple (diagnosis de los residuos e identificación de puntos influyentes).

Veamos el siguiente ejemplo.

2.1. EJEMPLO 2.

En el archivo "preciocasas.dat" tienen la información sobre 100 datos de precios de viviendas y sus características, el archivo se encuentra estructurado de la siguiente forma:

- Primera columna: precios de viviendas en euros.
- Segunda columna: superficie en metros cuadrados.
- Tercera: numero de cuartos de baño.
- Cuarta: número de dormitorios.
- Quinta: número de plazas de garaje.
- Sexta: edad de la vivienda .
- Séptima: 1 =buenas vistas y 0 =vistas corrientes

Suponga que deseamos estimar un modelo de regresión en el cual relacionemos el precio de una vivienda en función de sus características.

Realizamos los siguiente:

Lectura de los Datos:

```
> datos <- read.table(file="preciocasas.txt")
> datos
```

```
        V1  V2 V3 V4 V5 V6 V7
1   221333 154  2  3  1 13  0
2   229979 117  2  4  1 17  1
3   229979 119  2  3  1 11  1
4   231363 109  2  3  0  6  0
5   235167 117  2  3  1 17  1
6   236896 159  2  3  1 20  1
7   238625 155  3  4  1 12  1
8   245369 161  2  4  1 18  1
9   245542 116  2  3  1 13  0
```

10	251393	123	2	3	1	9	1
11	251770	105	2	3	0	8	0
12	260091	180	2	4	1	13	0
13	262833	159	2	4	1	13	1
14	262833	127	2	3	1	8	0
15	268021	156	2	4	1	15	0
16	276321	125	2	3	1	8	1
17	276321	150	3	3	1	11	0
18	280125	120	2	3	1	14	1
19	286350	158	3	3	1	18	0
20	293613	195	3	3	1	13	1
21	293958	150	2	3	1	9	0
22	297417	178	3	4	1	13	0
23	308760	125	2	3	1	5	0
24	310904	129	2	3	1	7	1
25	311250	176	3	5	1	13	0
26	318167	131	2	3	1	4	1
27	319685	165	2	3	1	10	0
28	327317	149	3	4	1	7	1
29	336472	136	2	3	1	10	1
30	339262	180	3	4	1	17	1
31	339432	179	3	4	1	9	1
32	363125	166	3	3	1	9	1
33	380417	164	3	4	1	3	1
34	386988	194	3	5	1	12	1
35	390792	189	3	3	1	9	1
36	428833	199	3	4	2	17	1
37	459958	214	3	4	1	11	1
38	674375	253	3	4	2	6	1
39	204042	82	2	4	1	19	1
40	212688	73	2	3	1	19	1
41	219604	91	2	5	1	17	1
42	231535	82	2	4	1	16	1
43	235167	91	2	5	1	17	1
44	237567	75	2	3	0	11	1
45	238625	91	3	4	1	19	1
46	243647	101	2	3	1	11	1
47	259375	112	3	4	1	12	1
48	265946	82	2	4	1	18	1
49	287042	118	2	3	1	8	1
50	304333	134	2	3	1	13	1
51	307373	101	3	4	1	16	1
52	308992	115	2	3	1	11	1
53	310212	134	2	3	1	10	1
54	311250	128	2	4	1	14	1
55	314017	113	3	4	1	17	1
56	316438	169	3	4	1	9	1
57	328542	122	2	3	1	7	1
58	338917	137	2	3	1	22	1
59	344104	145	3	4	1	13	1
60	370042	191	2	4	1	14	1
61	393385	189	3	4	1	13	1

```
62  432292 201  3  5  1  9  1
63  456500 145  3  5  2 13  1
64  262833 119  2  3  1 13  1
65  276321 151  3  3  1  7  0
66  276321 129  3  3  1 12  1
67  293958 150  2  3  1  7  1
68  301421 124  2  3  1  8  1
69  301567 147  3  3  1 10  1
70  316438 126  3  3  1  1  0
71  316438 150  4  4  1 16  1
72  319896 139  3  3  1  8  1
73  323354 184  3  4  1 15  1
74  325083 161  3  4  1  9  1
75  334421 168  3  3  1 12  1
76  336842 165  3  3  1  7  1
77  340646 146  3  3  1  3  1
78  342375 198  3  4  1 12  1
79  342375 154  3  3  1 12  1
80  354133 189  3  3  1  7  1
81  359667 180  3  4  1 13  1
82  370042 164  3  4  1 19  1
83  373154 174  3  3  1 11  1
84  376958 214  3  3  1 17  0
85  380417 178  3  3  1  9  1
86  382146 209  3  5  1  9  1
87  387333 186  3  4  1 11  1
88  390446 190  3  3  1  7  1
89  392867 185  4  4  1  9  1
90  395979 186  3  3  1  5  1
91  396671 193  4  5  1  8  1
92  397708 195  4  6  1 21  1
93  418458 179  3  4  1 11  1
94  418458 192  3  4  1 20  1
95  432292 184  3  4  1 19  1
96  442667 183  3  3  2  6  1
97  449583 204  3  3  2  7  1
98  453042 204  3  3  2 10  1
99  456500 213  3  4  1  6  1
100 458229 204  3  5  1 12  1
```

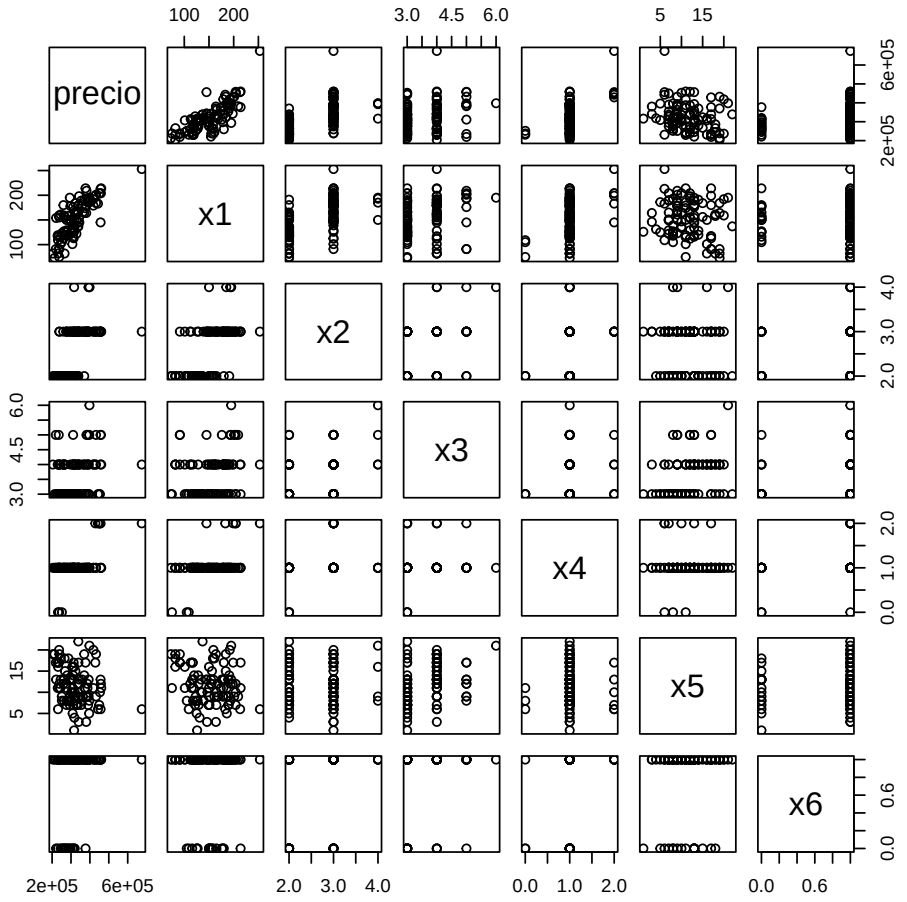
Nombrando a las Columnas:

```
> names(datos) <- c("precio", "x1", "x2", "x3", "x4", "x5", "x6" )
> head(datos)

  precio  x1 x2 x3 x4 x5 x6
1 221333 154  2  3  1 13  0
2 229979 117  2  4  1 17  1
3 229979 119  2  3  1 11  1
4 231363 109  2  3  0  6  0
5 235167 117  2  3  1 17  1
6 236896 159  2  3  1 20  1
```

Matriz de diagramas de dispersión

```
> plot(datos)
```



Se observa gráficamente que las variables independientes parecen influir en el comportamiento de nuestra variable dependiente.

Ajustamos el modelo de regresión

```
> modelo1 <- lm( precio ~ x1 + x2 + x3 + x4 + x5 + x6 , data = datos)
```

Resumen del modelo

```
> summary(modelo1)

Call:
lm(formula = precio ~ x1 + x2 + x3 + x4 + x5 + x6, data = datos)

Residuals:
    Min      1Q  Median      3Q     Max
-101248  -23050    -345   18036  141928
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  29844.7    26365.3   1.132  0.26056
x1            1159.3      142.9   8.112 1.98e-12 ***
x2           13284.5     9286.2   1.431  0.15591
x3            8695.2     6708.7   1.296  0.19814
x4           59777.1    14604.0   4.093 9.06e-05 ***
x5           -3198.4      974.3  -3.283  0.00145 **
x6           34312.9    10963.6   3.130  0.00234 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 38920 on 93 degrees of freedom
Multiple R-squared:  0.7505,        Adjusted R-squared:  0.7344
F-statistic: 46.61 on 6 and 93 DF,  p-value: < 2.2e-16
```

De los resultados anteriores puede apreciarse que el intercepto, y las variables x2 (número de cuarto de baño) y x3 (número de dormitorios) no parecen influir en la estimación del precio de la vivienda por lo podrían descartarse de la ecuación.

Una forma alternativa y mucho más eficiente para seleccionar el mejor conjunto de variables independientes es utilizar la instrucción step(), con la cual se utilizan los algoritmos conocidos para seleccionar variables (selección hacia adelante -"forward"-, hacia atrás -"backward"- o selección por pasos -"both"-).

```
> step(modelo1, direction="both")

Start:  AIC=2120.58
precio ~ x1 + x2 + x3 + x4 + x5 + x6

       Df  Sum of Sq        RSS    AIC
- x3    1 2.5444e+09 1.4340e+11 2120.4
<none>               1.4086e+11 2120.6
- x2    1 3.0996e+09 1.4395e+11 2120.8
- x6    1 1.4835e+10 1.5569e+11 2128.6
- x5    1 1.6322e+10 1.5718e+11 2129.6
- x4    1 2.5376e+10 1.6623e+11 2135.2
- x1    1 9.9664e+10 2.4052e+11 2172.1

Step:  AIC=2120.37
precio ~ x1 + x2 + x4 + x5 + x6

       Df  Sum of Sq        RSS    AIC
<none>               1.4340e+11 2120.4
+ x3    1 2.5444e+09 1.4086e+11 2120.6
- x2    1 5.3488e+09 1.4875e+11 2122.0
- x5    1 1.3780e+10 1.5718e+11 2127.6
- x6    1 1.6460e+10 1.5986e+11 2129.2
- x4    1 2.4664e+10 1.6806e+11 2134.2
- x1    1 1.0510e+11 2.4850e+11 2173.4

Call:
lm(formula = precio ~ x1 + x2 + x4 + x5 + x6, data = datos)
```

```
Coefficients:
(Intercept)           x1           x2           x4           x5           x6
      42224         1182        16724        58864        -2686        35913
```