Aplicación 2.7: Ventas en una cadena de supermercados

J. Ramajo

2020

Supongamos que somos el gerente de una cadena de 75 supermercados ubicados en distintas ciudades españolas. Las ventas totales en cada tienda dependen básicamente del precio de los productos que se venden en la misma y del gasto en publicidad que se realice en cada ciudad. El objetivo de este ejemplo consiste en determinar el efecto que tendrán sobre las ventas de cada supermercado distintas políticas de precios, así como distintas políticas de gasto en publicidad.

## Datos

En primer lugar, leeremos los datos y los examinaremos:

> library(readr)  
> VENTAS <- read\_csv("VENTAS\_SUPER.csv")

Parsed with column specification:  
cols(  
 A = col\_double(),  
 P = col\_double(),  
 V = col\_double()  
)

> dim(VENTAS)

[1] 75 3

> VENTAS

# A tibble: 75 x 3  
 A P V  
 <dbl> <dbl> <dbl>  
 1 1.3 5.69 73.2  
 2 2.9 6.49 71.8  
 3 0.8 5.63 62.4  
 4 0.7 6.22 67.4  
 5 1.5 5.02 89.3  
 6 1.3 6.41 70.3  
 7 1.8 5.85 73.2  
 8 2.4 5.41 86.1  
 9 0.7 6.24 81   
10 3 6.2 76.4  
# … with 65 more rows

> summary(VENTAS)

A P V   
 Min. :0.500 Min. :4.830 Min. :62.40   
 1st Qu.:1.100 1st Qu.:5.220 1st Qu.:73.20   
 Median :1.800 Median :5.690 Median :76.50   
 Mean :1.844 Mean :5.687 Mean :77.37   
 3rd Qu.:2.700 3rd Qu.:6.210 3rd Qu.:82.20   
 Max. :3.100 Max. :6.490 Max. :91.20

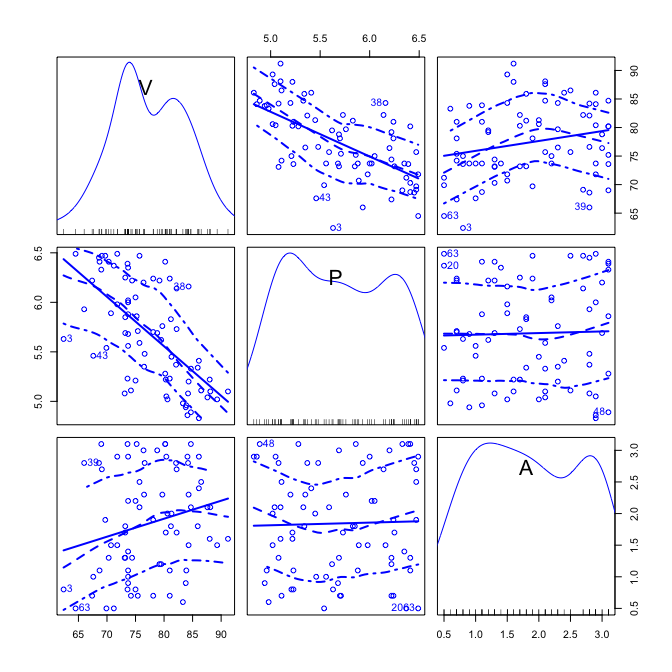
## Gráfica de las variables

Haremos un análisis gráfico a través de una *scatterplot matrix* de las tres variables del modelo, empleando la función scatterplotMatrix() en la librería **car**:

> library("car")

Loading required package: carData

> scatterplotMatrix(~ V + P + A, id=list(n=3),  
+ smooth=list(span=0.7), data=VENTAS)



## Modelo econométrico

La ecuación que se ajusta es la siguiente:

donde representan los ingresos mensuales por ventas en cada ciudad, es el índice de precios en el supermercado de esa ciudad, y son los gastos mensuales en publicidad para promocionar los productos de la tienda. *V* y *A* se miden en miles de euros, mientras que *P* es un índice ponderado del precio (en euros) de todos los productos que se venden en cada tienda.

> modelo.ventas <- lm(V ~ P + A, data=VENTAS)

> S(modelo.ventas)

Call: lm(formula = V ~ P + A, data = VENTAS)  
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 118.9136 6.3516 18.722 < 2e-16 \*\*\*  
P -7.9079 1.0960 -7.215 4.42e-10 \*\*\*  
A 1.8626 0.6832 2.726 0.00804 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard deviation: 4.886 on 72 degrees of freedom  
Multiple R-squared: 0.4483  
F-statistic: 29.25 on 2 and 72 DF, p-value: 5.041e-10   
 AIC BIC   
455.74 465.01

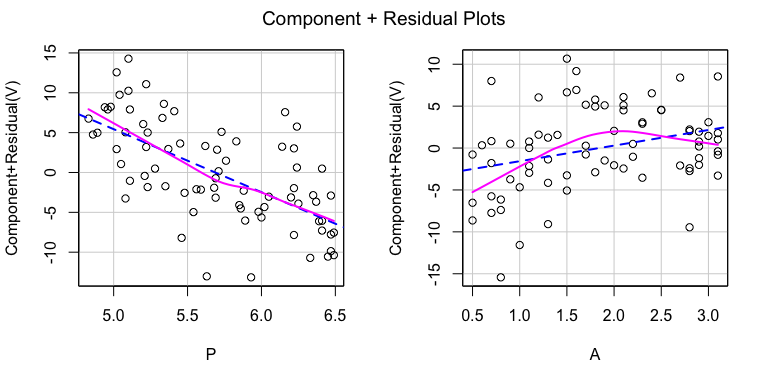
## Diagnósticos de la regresión

Para comprobar si el modelo estimado ajusta adecuadamente los datos llevaremos a cabo diferentes contrastes de diagnóstico:

### No-linealidad

Gráficos C+R (*Component-plus-Residual*) para *P* y *A* para el análisis de la linealidad de las regresiones parciales:

> crPlots(modelo.ventas, smooth=list(span=0.7))

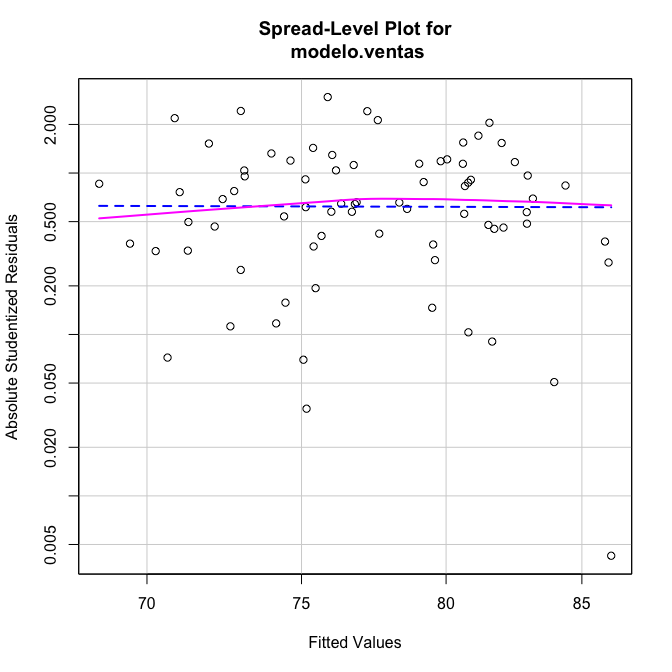


NOTA: Cuando el tamaño de la muestra es pequeño es mejor tomar un parámetro de suavizado relativamente elevado (el parámetro por defecto es 0.5, pero aquí se ha tomado el valor 0.7).

### Heteroscedasticidad (varianza no constante)

En nuestro caso mediremos si la varianza residual cambia con el nivel de la variable respuesta o de los regresores:

> spreadLevelPlot(modelo.ventas, smooth=list(span=1))



Suggested power transformation: 1.087202

> ncvTest(modelo.ventas)

Non-constant Variance Score Test   
Variance formula: ~ fitted.values   
Chisquare = 0.3372845, Df = 1, p = 0.5614

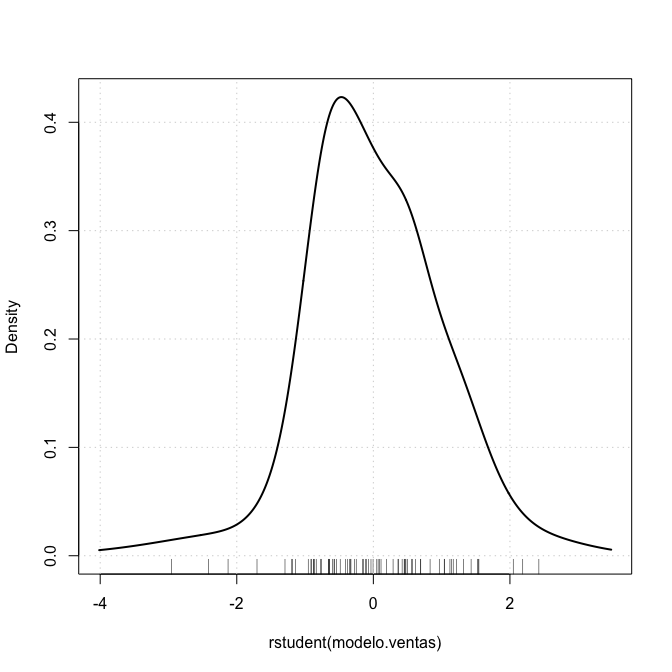
> ncvTest(modelo.ventas, var.formula= ~ P + A)

Non-constant Variance Score Test   
Variance formula: ~ P + A   
Chisquare = 2.796449, Df = 2, p = 0.24704

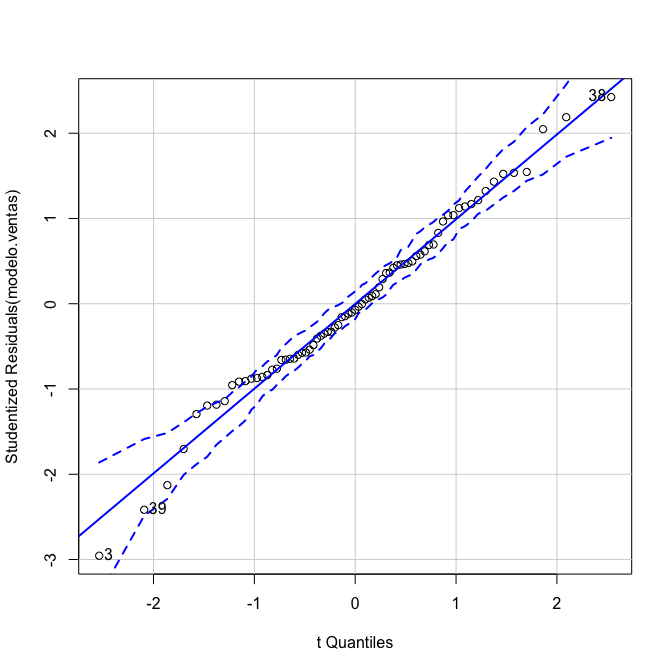
### Datos atípicos

Se realizará un examen de los residuos *estudentizados* para el análisis de outliers:

> densityPlot(rstudent(modelo.ventas))



> qqPlot(modelo.ventas, id=list(n=3))



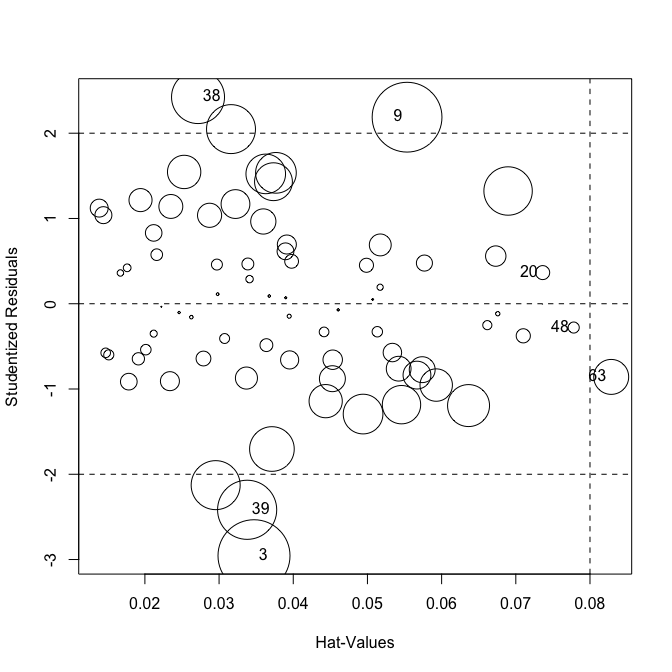
[1] 3 38 39

> outlierTest(modelo.ventas)

No Studentized residuals with Bonferroni p < 0.05  
Largest |rstudent|:  
 rstudent unadjusted p-value Bonferroni p  
3 -2.955519 0.0042334 0.31751

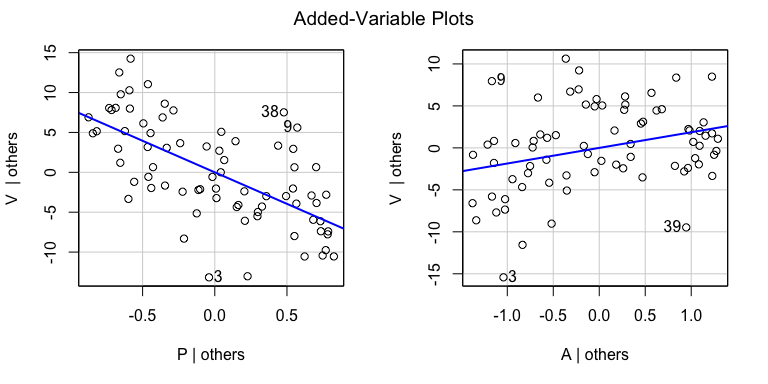
Y, finalmente, un examen conjunto de las observaciones influyentes (residuos grandes y leverages elevados) y del efecto de estas sobre los coeficientes estimados para *P* y *A*:

> influencePlot(modelo.ventas, id=list(n=3))



StudRes Hat CookD  
3 -2.9555191 0.03470814 0.094537027  
9 2.1879407 0.05534113 0.088809643  
20 0.3652486 0.07362313 0.003577184  
38 2.4238667 0.02716299 0.051213014  
39 -2.4166263 0.03378357 0.063778386  
48 -0.2791333 0.07780693 0.002219710  
63 -0.8581504 0.08285940 0.022258926

> avPlots(modelo.ventas, id=list(n=3, method="mahal"))



Si se eliminan de la regresión las observaciones atípicas el resultado sería el siguiente:

> modelo.ventas.2 <- update(modelo.ventas, subset=-c(3,9,38,39))  
> S(modelo.ventas.2)

Call: lm(formula = V ~ P + A, data = VENTAS, subset = -c(3, 9, 38, 39))  
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 121.1966 5.4576 22.207 < 2e-16 \*\*\*  
P -8.3901 0.9469 -8.861 6.02e-13 \*\*\*  
A 2.1293 0.6018 3.538 0.000732 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard deviation: 4.153 on 68 degrees of freedom  
Multiple R-squared: 0.5652  
F-statistic: 44.19 on 2 and 68 DF, p-value: 5.052e-13   
 AIC BIC   
408.61 417.66

> compareCoefs(modelo.ventas, modelo.ventas.2)

Calls:  
1: lm(formula = V ~ P + A, data = VENTAS)  
2: lm(formula = V ~ P + A, data = VENTAS, subset = -c(3, 9, 38, 39))  
  
 Model 1 Model 2  
(Intercept) 118.91 121.20  
SE 6.35 5.46  
   
P -7.908 -8.390  
SE 1.096 0.947  
   
A 1.863 2.129  
SE 0.683 0.602