

Ejemplo: validación del supuesto de homocedasticidad

Javier Santibáñez

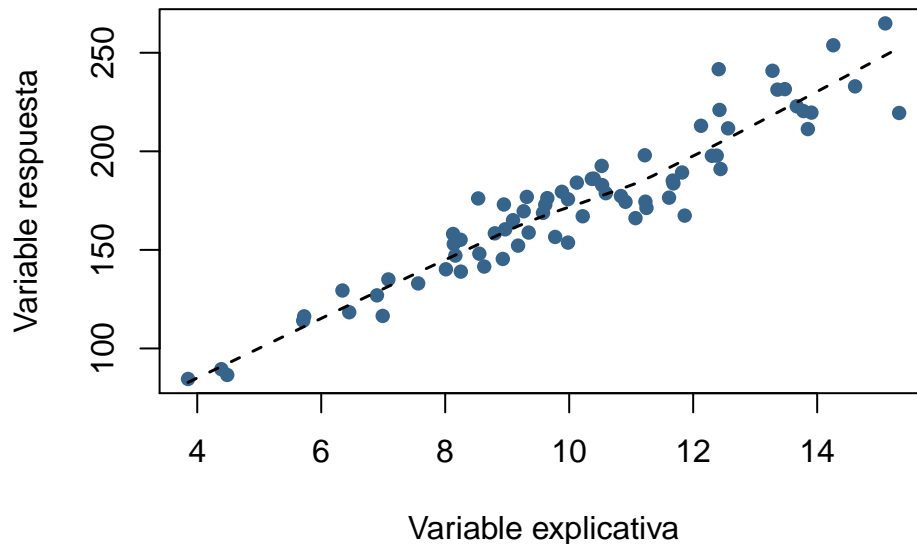
20/8/2018

Descripción del conjunto de datos

$$Y_i = 31 + 2.5x_i + \epsilon_i, \quad \epsilon_i \sim N(0, 2e^{0.4x_i}), \quad i = 1, \dots, 72.$$

```
set.seed(130419)
x <- rnorm(72, 10, 3)
y <- 31 + 14*x + exp(0.2*x)*rnorm(72, 0, sqrt(2))

par(mar = c(4, 4, 1, 1))
plot(x, y, pch = 16, col = 'steelblue4',
     xlab = 'Variable explicativa', ylab = 'Variable respuesta')
lines(loess.smooth(x,y), lwd = 1.5, lty = 2)
```



Ajuste del modelo lineal

```
lm.fit <- lm(y ~ x)
summary(lm.fit)

##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -30.631  -8.207  -0.929   8.710  35.954
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  31.9917     6.0173   5.317 1.2e-06
```

```
## x          13.9962    0.5755  24.320 < 2e-16
##
## Residual standard error: 12.58 on 70 degrees of freedom
## Multiple R-squared:  0.8942, Adjusted R-squared:  0.8927
## F-statistic: 591.5 on 1 and 70 DF,  p-value: < 2.2e-16
```

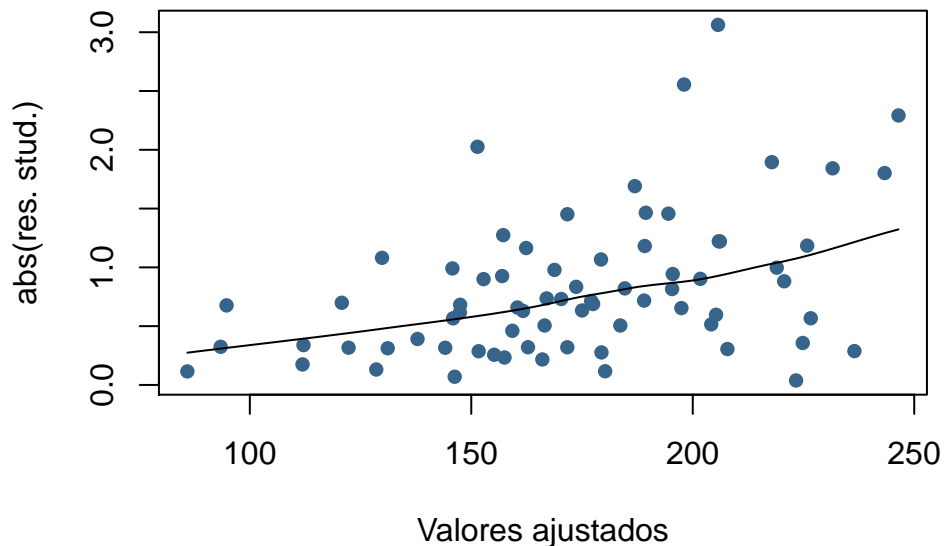
Validación del supuesto de homocedasticidad

```
library(car)
ncvTest(lm.fit)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 10.90862    Df = 1    p = 0.0009571756
```

Validación del supuesto de homocedasticidad (a mano)

```
par(mar = c(4, 4, 1, 1))
lm.res_t <- rstudent(lm.fit)
plot(fitted(lm.fit), abs(lm.res_t), pch=16, col='steelblue4',
     xlab = 'Valores ajustados', ylab = 'abs(res. stud.)')
lines(loess.smooth(fitted(lm.fit), abs(lm.res_t)))
```



```
het_test <- lm(I(abs(lm.res_t)) ~ fitted(lm.fit))
summary(het_test)
```

```
##
## Call:
## lm(formula = I(abs(lm.res_t)) ~ fitted(lm.fit))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.13441 -0.34156 -0.07615  0.23631  2.01544
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   -0.421142   0.320991  -1.312 0.193805
## fitted(lm.fit) 0.007136   0.001808   3.947 0.000186
##
## Residual standard error: 0.5532 on 70 degrees of freedom
## Multiple R-squared:  0.182, Adjusted R-squared:  0.1703
## F-statistic: 15.58 on 1 and 70 DF,  p-value: 0.0001863
```

Ajuste por MC ponderados

```
wlm.fit <- lm(y ~ x, weights = exp(-0.2*x))
summary(wlm.fit)
```

```
##
## Call:
## lm(formula = y ~ x, weights = exp(-0.2 * x))
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -9.4692 -3.2838 -0.3382  3.1216 10.5703
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30.6067     4.3774   6.992 1.29e-09
## x            14.1450     0.4785  29.563 < 2e-16
##
## Residual standard error: 4.174 on 70 degrees of freedom
## Multiple R-squared:  0.9258, Adjusted R-squared:  0.9248
## F-statistic:  874 on 1 and 70 DF,  p-value: < 2.2e-16
```

```
ncvTest(wlm.fit)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.7436495    Df = 1    p = 0.3884943
```