

# Script\_05.R

jairo

2025-09-25

```
# JAIRO ALBERTO LEAL GOMEZ
# 25/09/2025
# SEMANA 8

# SCRIPT 4

# Regresion Lineal Simple

# Variable x es independiente, o explicativa o exogena
# Variable y es dependiente o respuesta o endogena

# Ejemplos
# Crecimiento de arboles, DAP y altura total
# Hidrologia, caudal del rio y la precipitacion acumulada
# Pesca, peso de un pez, longitud total

# Causalidad, entender la realcion causa y efecto

# Tipos de relacion: determinista y no determinista, lineal (positiva y negativa),
# no lineal (por ej. logaritmica), ausencia de correlacion

# El modelo de regresion lineal simple
# beta 0 es el intercepto
# beta 1 es la pendiente

# Metodo de minimos cuadrados

# Ejercicio 1

trigo <- c(30, 28, 32, 25, 25, 25, 22, 24, 35, 40)
precio <- c(25, 30, 27, 40, 42, 40, 50, 45, 30, 25)

datos <- data.frame(trigo, precio)

datos

##      trigo precio
## 1      30      25
## 2      28      30
## 3      32      27
## 4      25      40
## 5      25      42
## 6      25      40
```

```

## 7      22      50
## 8      24      45
## 9      35      30
## 10     40      25

sumxi_yi <- sum(datos$trigo*datos$precio)
sumxi_yi

## [1] 9734

nxy <- length(datos$trigo)*mean(datos$trigo)*mean(datos$precio)
nxy

## [1] 10124.4

# length para sacar n
length(datos$trigo)

## [1] 10

sumxi_yi - nxy

## [1] -390.4

x12 <- sum(datos$trigo^2)
x12

## [1] 8468

nx2 <- length(datos$trigo)*mean(datos$trigo)^2
nx2

## [1] 8179.6

b1 <- (sumxi_yi - nxy) / (x12 - nx2)
b1

## [1] -1.353675

bo <- mean(datos$precio) - b1*mean(datos$trigo)
bo

## [1] 74.11512

# Función de modelo lineal

lm(formula = datos$precio ~ datos$trigo)

##
## Call:
## lm(formula = datos$precio ~ datos$trigo)
##
## Coefficients:
## (Intercept)  datos$trigo
##      74.115      -1.354

fit.lm <- lm(datos$precio ~ datos$trigo)
summary(fit.lm)

##
## Call:
## lm(formula = datos$precio ~ datos$trigo)

```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.5049 -2.9164  0.7268  3.3457  5.6657
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  74.1151     8.7358   8.484 2.85e-05 ***
## datos$trigo  -1.3537     0.3002  -4.509  0.00198 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.098 on 8 degrees of freedom
## Multiple R-squared:  0.7176, Adjusted R-squared:  0.6824
## F-statistic: 20.33 on 1 and 8 DF,  p-value: 0.001978
```

```
sqrt(0.6824)
```

```
## [1] 0.8260751
```

```
# es igual a la correlacion y es bastante fuerte
```

```
anova(fit.lm)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: datos$precio
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## datos$trigo   1  528.47   528.47   20.333 0.001978 **
## Residuals     8  207.93    25.99
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Libreria de modelos de regresion
```

```
library(lmtest)
```

```
## Cargando paquete requerido: zoo
```

```
##
```

```
## Adjuntando el paquete: 'zoo'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      as.Date, as.Date.numeric
```

```
# Revisar la posible heterocedasticidad o homocedasticidad
```

```
bptest(fit.lm)
```

```
##
```

```
## studentized Breusch-Pagan test
```

```
##
```

```
## data: fit.lm
```

```
## BP = 0.33274, df = 1, p-value = 0.5641
```

```
fit.lm$model
```

```
##      datos$precio datos$trigo
```

```
## 1              25          30
```

```
## 2      30      28
## 3      27      32
## 4      40      25
## 5      42      25
## 6      40      25
## 7      50      22
## 8      45      24
## 9      30      35
## 10     25      40
```

```
fit.lm$coefficients
```

```
## (Intercept) datos$trigo
## 74.115118 -1.353675
```

```
fit.lm$residuals
```

```
##      1      2      3      4      5      6      7
## -8.5048544 -6.2122053 -3.7975035 -0.2732316 1.7267684 -0.2732316 5.6657420
##      8      9     10
## 3.3730929 3.2635229 5.0319001
```

```
mean(fit.lm$residuals)
```

```
## [1] -4.883247e-16
```

```
datos$yprima <- 74.11 - 1.3536 * datos$trigo
datos$recta <- fit.lm$fitted.values
datos$prima <- bo + b1 * datos$trigo
```

```
datos$residuales <- datos$precio-datos$recta
SSE <- sum(datos$residuales^2)
SSE/8
```

```
## [1] 25.99064
```

```
sqrt(SSE/8)
```

```
## [1] 5.098101
```