

Lección 7: Correlación y regresión lineal

Bioestadística con R

Ejercicios

Ejercicio 1: En un estudio de la síntesis de proteína en el oocito de la rana *Xenopus laevis*, un biólogo inyectó oocitos con leucina marcada radiactivamente. En distintos tiempos tras la inyección, hizo las mediciones de radioactividad y calculó cuánto de la leucina había sido incorporada a proteínas. Los resultados son los siguientes. Cada valor de leucina es la cantidad de leucina marcada en dos oocitos. Todos los oocitos fueron de la misma hembra.

Tiempo (min)	Leucina
0	0.02
10	0.25
20	0.54
30	0.69
40	1.07
50	1.50
60	1.74

A partir de estos datos:

- Hacer un gráfico de dispersión donde se muestre la relación entre el tiempo y la cantidad de leucina.
- Hacer una regresión lineal para estimar el ritmo al que se incorpora la leucina.
- Obtener el error estándar de los residuales.
- Calcular el coeficiente de correlación.
- *Actividad extra:* Realizar una interpolación para estimar el valor esperado pasado 45 minutos.
- *Actividad extra:* Calcular los residuales.

Soluciones

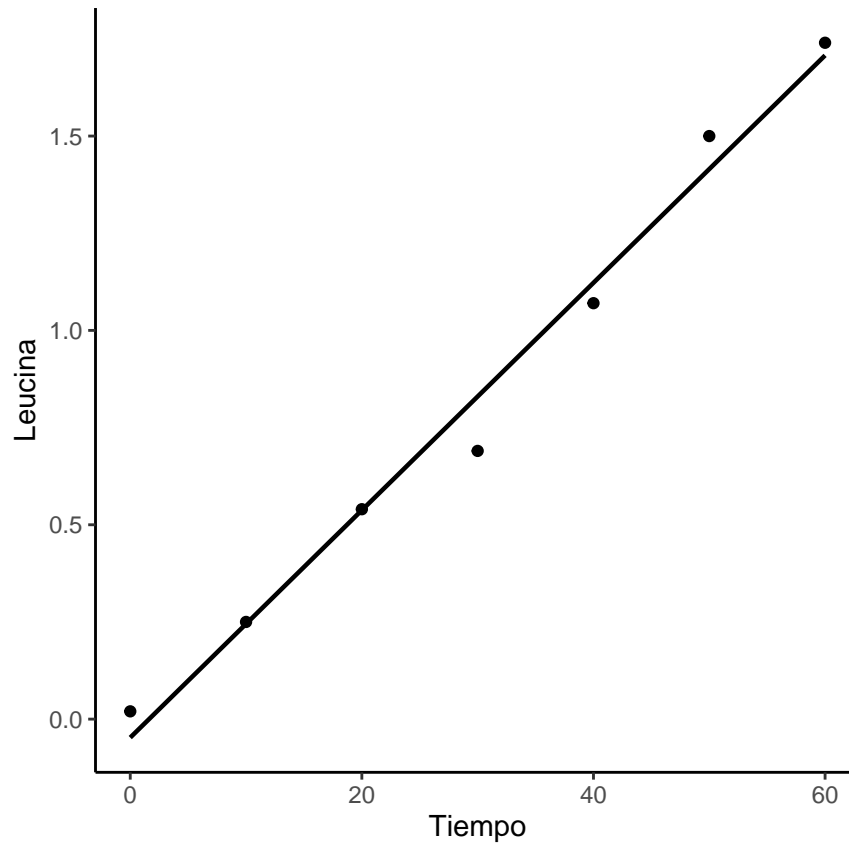
Ejercicio 1: Primero vamos a realizar el gráfico de nuestra relación entre las dos variables, para esto vamos a utilizar `ggplot2`.

```
# Cargamos nuestros datos
tiempo <- c(0, 10, 20, 30, 40, 50, 60)
leucina <- c(0.02, 0.25, 0.54, 0.69, 1.07, 1.50, 1.74)
ej1 <- data.frame(Tiempo = tiempo, Leucina = leucina)

# Realizamos el gráfico
ggplot(ej1, aes(x = Tiempo, y = Leucina)) +
  geom_point(size = 1.5) +
  labs(x = "Tiempo", y = "Leucina") +
  geom_smooth(method = "lm", se = F, color = "black", size = 0.8) +
```

```
theme_classic() +
  theme(aspect.ratio = 1)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Como podemos ver, parece que existe una relación lineal en cuanto al tiempo y la cantidad de leucina. Vamos a generar un modelo lineal para corroborar esto.

```
ej1.lm <- lm(Leucina ~ Tiempo, ej1)
summary(ej1.lm)
```

```
##
## Call:
## lm(formula = Leucina ~ Tiempo, data = ej1)
##
## Residuals:
##      1      2      3      4      5      6      7
## 0.0675 0.0050 0.0025 -0.1400 -0.0525 0.0850 0.0325
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.047500   0.057192  -0.831   0.444
## Tiempo       0.029250   0.001586  18.440 8.63e-06 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08393 on 5 degrees of freedom
## Multiple R-squared:  0.9855, Adjusted R-squared:  0.9826
## F-statistic: 340 on 1 and 5 DF, p-value: 8.628e-06
```

El ritmo de incorporación viene dado por el valor de la pendiente b_1 , que en este caso corresponde a 0.2925 como se pueden observar en los resultados de nuestro modelo lineal.

También podemos observar el error estándar de nuestros residuales cuando tenemos $df = 5$, que corresponde a s_e . En este caso vale 0.0839. Por último calculamos el coeficiente de correlación.

```
cor.test(ej1$Tiempo, ej1$Leucina)

##
## Pearson's product-moment correlation
##
## data: ej1$Tiempo and ej1$Leucina
## t = 18.44, df = 5, p-value = 8.628e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9494956 0.9989724
## sample estimates:
##      cor
## 0.992728
```

Como podemos ver, el valor de nuestra correlación es de 0.9927, un valor casi perfecto y positivo. Esto quiere decir que conforme aumenta el tiempo aumenta el nivel de leucina, al menos con los datos observados, siguiendo un patrón lineal.

Actividad extra: Para calcular el valor estimado pasado 45 minutos, es decir, entre los valores conocidos de 40 y 50, necesitamos realizar una *interpolación*. Para esto utilizamos la función `approx()`, con el argumento `xout` para indicar qué valor es el que queremos calcular.

```
approx(ej1$Tiempo, ej1$Leucina, xout = 45)

## $x
## [1] 45
##
## $y
## [1] 1.285
```

El valor aproximado pasados 45 minutos es de 1.285.

Para calcular los residuales utilizamos la función `residuals()`, que nos dará los residuales de cada uno de nuestros puntos.

```
residuals(ej1.lm)

##      1      2      3      4      5      6      7
## 0.0675 0.0050 0.0025 -0.1400 -0.0525 0.0850 0.0325
```