Introducción a la regresión lineal

Ramon Ceballos

6/2/2021

1. Cálculo de una recta de regresión lineal

1.1 Planteamiento del problema

Como ya hemos dicho, el objetivo de este tema es estudiar si existe relación lineal entre las variables dependiente e independiente.

Por lo general, cuando tenemos una serie de observaciones emparejadas, $(x_i, y_i)_{i=1,\ldots,n}$, la forma natural de almacenarlas en R es mediante una tabla de datos, con dos columnas (una para "x" y otra para "y"). Y la que más conocemos nostros es el data frame.

Como recordaréis de temas anteriores, la ventaja de trabajar con este tipo de organización de datos es que luego se pueden hacer muchas cosas.

1.2 Ejemplo para el cálculo

1.2.1 Análisis exploratorio de los datos de estudio

En este ejemplo, nosotros haremos uso del siguiente data frame:

```
body = read.table("../../../../data/bodyfat.txt", header = TRUE)
head(body,3)
```

```
##   Density  Fat Age Weight Height Neck Chest Abdomen  Hip Thigh Knee Ankle
## 1  1.0708 12.3  23 154.25  67.75 36.2  93.1    85.2 94.5  59.0 37.3  21.9
## 2  1.0853  6.1  22 173.25  72.25 38.5  93.6    83.0 98.7  58.7 37.3  23.4
## 3  1.0414 25.3  22 154.00  66.25 34.0  95.8    87.9 99.2  59.6 38.9  24.0
##   Biceps Forearm Wrist
## 1   32.0    27.4  17.1
## 2   30.5    28.9  18.2
## 3   28.8    25.2  16.6
```

Más concretamente, trabajaremos con las variables **Fat** y **Weight**.

```
#seleccionamos las columnas Fat y Weight
body2 = body[,c(2,4)]

#Cambiamos los nombres de las columnas
names(body2) = c("Grasa","Peso")
```

```r
#Observamos la estructura
str(body2)
```

```
## 'data.frame':    252 obs. of  2 variables:
##  $ Grasa: num  12.3 6.1 25.3 10.4 28.7 20.9 19.2 12.4 4.1 11.7 ...
##  $ Peso : num  154 173 154 185 184 ...
```

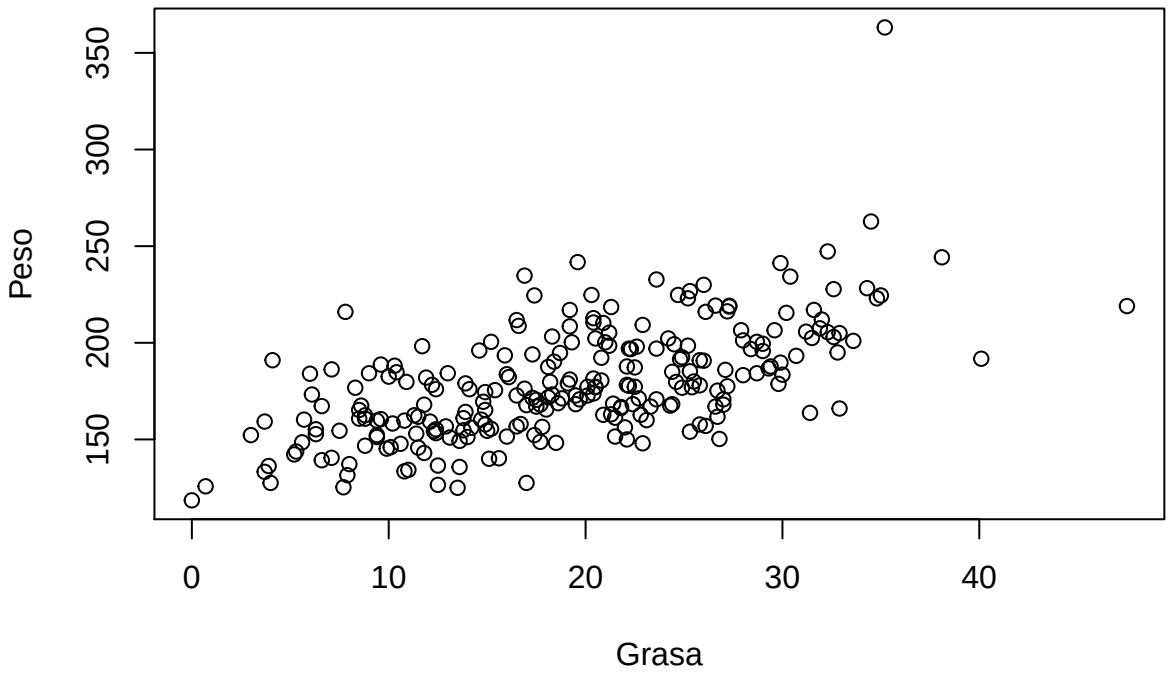
```r
head(body2,3)
```

```
##   Grasa   Peso
## 1  12.3 154.25
## 2   6.1 173.25
## 3  25.3 154.00
```

Al analizar datos, siempre es recomendable empezar con una representación gráfica que nos permita hacernos a la idea de lo que tenemos.

Esto se consigue haciendo uso de la función **plot**, que ya hemos estudiado en detalle en lecciones anteriores. No obstante, para lo que necesitamos en este tema nos conformamos con un gráfico básico de estos puntos que nos muestre su distribución.

```r
#Plot básico de ptos
plot(body2)
```



1.2.2 Cálculo de la recta de regresión

Para calcular la **recta de regresión** con R de la familia de puntos $(x_i, y_i)_{i=1,\dots,n}$, si `x` es el vector $(x_i)_{i=1,\dots,n}$ e `y` es el vector $(y_i)_{i=1,\dots,n}$, entonces, su recta de regresión se calcula mediante la instrucción:

`lm(y~x)` (linear model)

Cuidado con la sintaxis: primero va el vector de las variables dependientes "y" (lo que se quiere predecir), seguidamente después de una tilde `~`, va el vector de las variables independientes "X".

Esto se debe a que R toma el significado de la tilde como "en función de". Es decir, la interpretación de **`lm(y~x)`** en R es "la recta de regresión de y en función de x".

Si los vectores **`y`** y **`x`** son, en este orden, la primera y la segunda columna de un data frame de dos variables, entonces es suficiente aplicar la función **`lm`** al data frame.

En general, si **`x`** e **`y`** son dos variables de un data frame, para calcular la recta de regresión de **`y`** en función de **`x`** podemos usar la instrucción:

`lm(y~x, data = data fame)`

Si la varibale independiente es la grasa y la variable dependiente es el peso, se hace lo siguiente.

```
#Opción 1
lm(body2$Peso~body2$Grasa)
```

```
##
## Call:
## lm(formula = body2$Peso ~ body2$Grasa)
##
## Coefficients:
## (Intercept)  body2$Grasa
##     137.738        2.151
```

```
#Opción 2 (Mejor opción)
lm(Peso~Grasa, data = body2)
```

```
##
## Call:
## lm(formula = Peso ~ Grasa, data = body2)
##
## Coefficients:
## (Intercept)        Grasa
##     137.738        2.151
```

Como podéis observar, las dos formas de llamar a la función dan exactamente lo mismo. Ninguna es mejor que la otra.

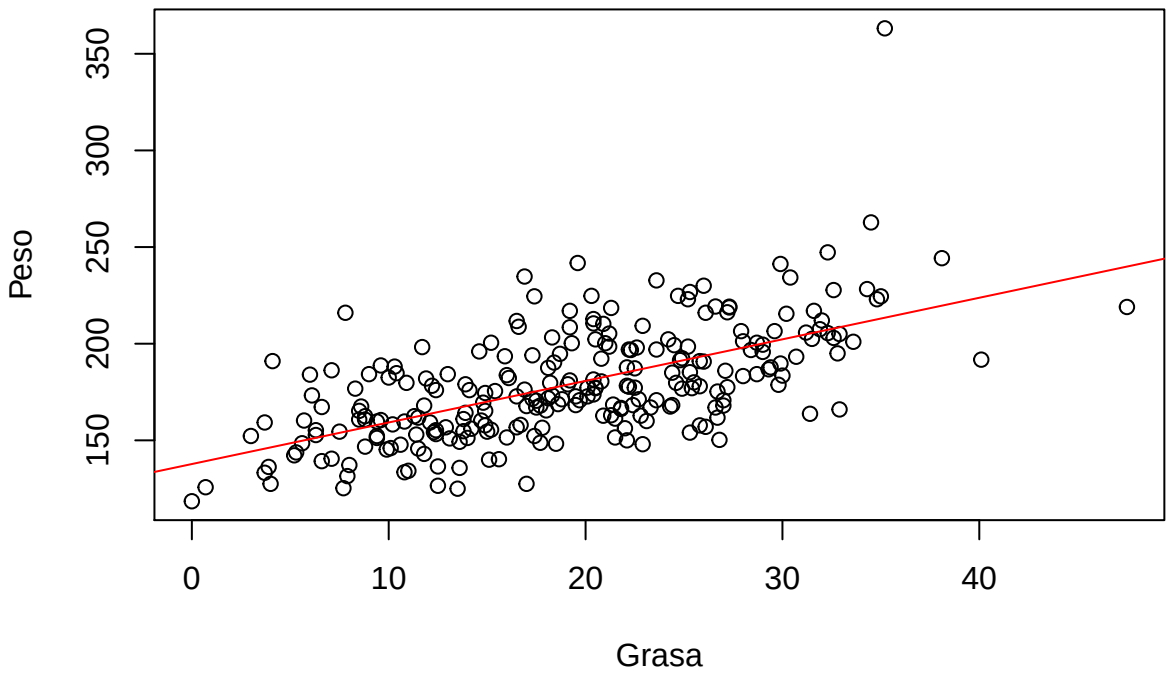
El resultado obtenido en ambos casos significa que la recta de regresión para nuestros datos es:

$$y = 2.151x + 137.738$$

Ahora, podemos superponer esta recta a nuestro gráfico anterior haciendo uso de la función **`abline()`**.

```
#Nube de ptos anterior
plot(body2)

#Añado la recta de regresión lineal
abline(lm(Peso~Grasa, data = body2), col = "red")
```



Hay que tener en cuenta que el análisis llevado a cabo hasta el momento de los pares de valores $(x_i, y_i)_{i=1,\ldots,n}$ ha sido puramente descriptivo.

Es decir, hemos mostrado que estos datos son consistentes con una función lineal, pero no hemos demostrado que la variable dependiente sea función aproximadamente lineal de la variable independiente. Esto último necesitaría una demostración matemática, o bien un argumento biológico, pero no basta con una simple comprobación numérica.

Eso sí, podemos utilizar todo lo hecho hasta ahora para predecir valores $\tilde{y}_i$ en función de los x_i resolviendo una simple ecuación lineal.

2. Coeficiente de determinación

El **coeficiente de determinación**, R^2, nos es útil para evaluar numéricamente si la relación lineal obtenida es significativa o no.

No explicaremos de momento como se define. Eso lo dejamos para curiosidad del usuario. Por el momento, es suficiente con saber que este coeficiente se encuentra en el intervalo $[0, 1]$. Si R^2 es mayor a 0.9, consideraremos que el ajuste es bueno. De lo contrario, no.

2.1 Cálculo del coeficiente de determinación (summary)

La función `summary` aplicada a `lm` nos muestra los contenidos de este objeto. Entre ellos encontramos `Multiple R-squared`, que no es ni más ni menos que el coeficiente de determinación, R^2.

Para facilitarnos las cosas y ahorrarnos información que, de momento, no nos resulta de interés, podemos aplicar **`summary(lm(...))$r.squared`**.

```
#todo lo que devuelve summary
summary(lm(Peso~Grasa, data = body2))
```

```
##
## Call:
## lm(formula = Peso ~ Grasa, data = body2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.799 -14.999  -3.469  11.860 149.709
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 137.7375     3.6684   37.55   <2e-16 ***
## Grasa         2.1507     0.1756   12.25   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.28 on 250 degrees of freedom
## Multiple R-squared:  0.3751, Adjusted R-squared:  0.3726
## F-statistic:   150 on 1 and 250 DF,  p-value: < 2.2e-16
```

```
#valor directo del coeficiente
summary(lm(Peso~Grasa, data = body2))$r.squared
```

```
## [1] 0.3750509
```

En este caso, hemos obtenido un coeficiente de determinación de 0.3751, cosa que confirma que la recta de regresión no aproxima nada bien nuestros datos.