Introdución a la Regresión Lineal

Santiago Pérez Moncada

24/7/2020

Seguramente, en algún momento de nuestra vida ya sea hojeando un libro de matemáticas, curioseando artículos científicos... haz visto una línea recta o algún tipo de curva en un gráfico que se ajusta a las observaciónes representadas por medio de puntos en el plano.

En general, la situación es la siguiente:Supongamos que tenemos una serie de puntos en el plano cartesiano $\mathbb{R}^2$ de la forma.

$$(x_1, y_1), ..., (x_n, y_n)$$

que representan las observaciones de dos variable numéricas. Digamos que x es la edad e y el peso de n estudiantes.

Nuestro objetivo: Describir la relación entre la **variable independiente**, x, y la **variable dependiente**, y, apartir de estas observaciones.

Para ello, lo que haremos será buscar una función $y = f(x)$ cuya gráfica se aproxime lo máximo posible a nuestro pares ordenados $(x_i, y_i) \quad i = 1, ..., n$

Esta función nos dará un modelo matemático de cómo se comportan estas observaciones, lo cual nos permitirá entender mejor los mecanismos que relacionan las variables estudiadad o incluso, nos dara la oportunidad de hacer predicciones sobre futuras observaciones.

El modelo lineal es el modelo mas simple que existe. Consiste en estudiar si los puntos $(x_i, y_i) \quad i = 1, ..., n$ satisfacen una relación de la forma.

$$y = ax + b \quad a, b \in \mathbb{R}$$

En este caso, se busca la recta $y = ax + b$ que mejor se aproxime a los puntos dados imponiendo que la suma de los cuadrados de las diferencias entre sus valore y_i y sus aproximaciones $\tilde{y_i} = ax_i + b$ sea mínima. Es decir que

$$\sum_{i=1}^{n} (y_i - \tilde{y_i})^2$$

El objetivo de este tema no es más que enseñarnos hacer uso de R para obtener esa recta de regresión.

Veremos tambien cómo se puede evaluar numéricamente si esta recta se ajusta bien a las observaciones dadas.

Para ello, introduciremos algunas funciones de R y haremos uso de trasformaciones logarítmicas para tratar casos en los que los puntos dados se aproximen mejor mediante una función exponencial o potencial.

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, $(x_i, y_i) \quad i = 1, ..., n$, la forma natural de almacenarlas en R es mediante una tabla de datos. Y la que más conocemos es un data frame.

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

Usaremos los datos de body fat

```
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`

```
body2 = body[,c(2,4)]
names(body2) = c("Grasa","Peso")
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 ...
```

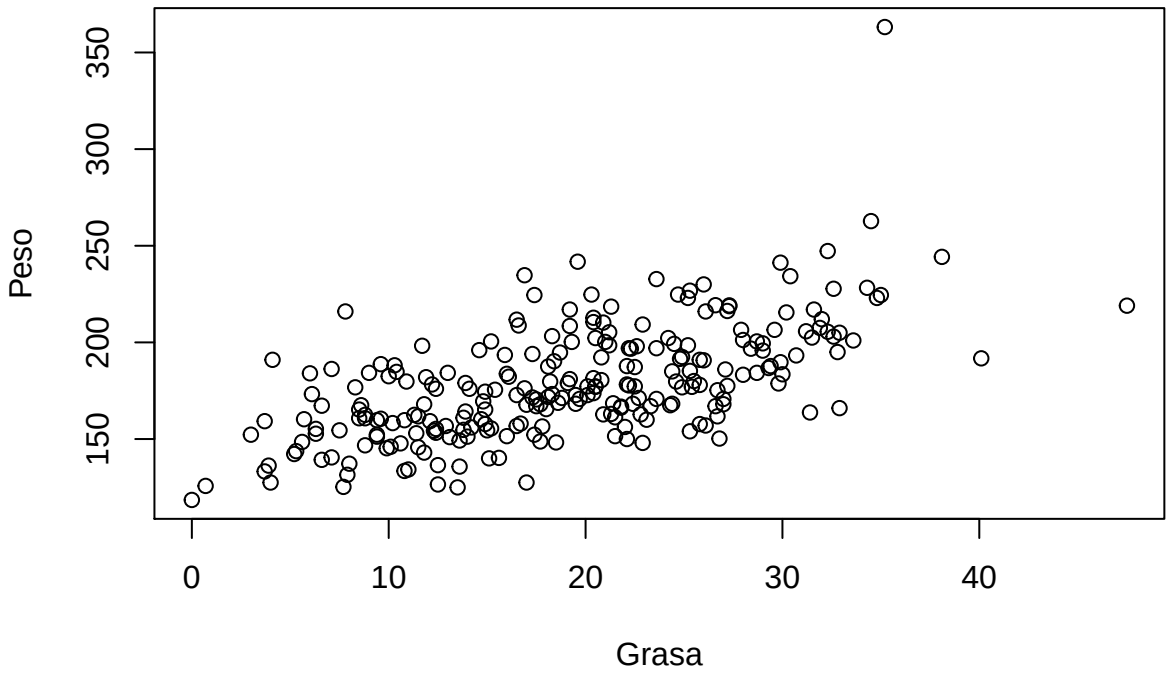
```
head(body2)
```

```
##   Grasa   Peso
## 1  12.3 154.25
## 2   6.1 173.25
## 3  25.3 154.00
## 4  10.4 184.75
## 5  28.7 184.25
## 6  20.9 210.25
```

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 a detalle en lecciones anteriores. No obstante, para lo que necesitamos en este tema nos conformaremos con un gráfico básico de estos puntos que nos muestre su distribución.

```r
plot(body2)
```



Para calcular la **recta de regresión** con `R` de la familia de puntos $(x_i, y_i) \quad i = 1, ..., n$ si $\vec{x}$ es el vector de datos $\vec{x} = (x_1, ..., x_n)$ e $\vec{y}$ es el vector de datos $\vec{y} = (y_1, ..., y_n)$, entonces su recta de regresión en `R` se calcula mediante la instrucción `lm(y~x)`

Cuidado con la sintaxis: primero va el vector de las variables dependientes y, segudamente después de una tilde ~, va el vector de las variables independientes.

Esto se debe a que `R` toma el signidicado 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 $\vec{x}$ e $\vec{y}$ 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 $\vec{x}$ e $\vec{y}$ son dos variables de un data frames, para calcular la recta de regresión de $\vec{y}$ en función de $\vec{x}$ podemos usar la instrucción. `lm(y~x, data=DataFrame)`.

```r
lm(body2$Peso~body2$Grasa) #Opcion 1
```

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

```r
lm(Peso~Grasa, data = body2)#Opcion 2
```

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

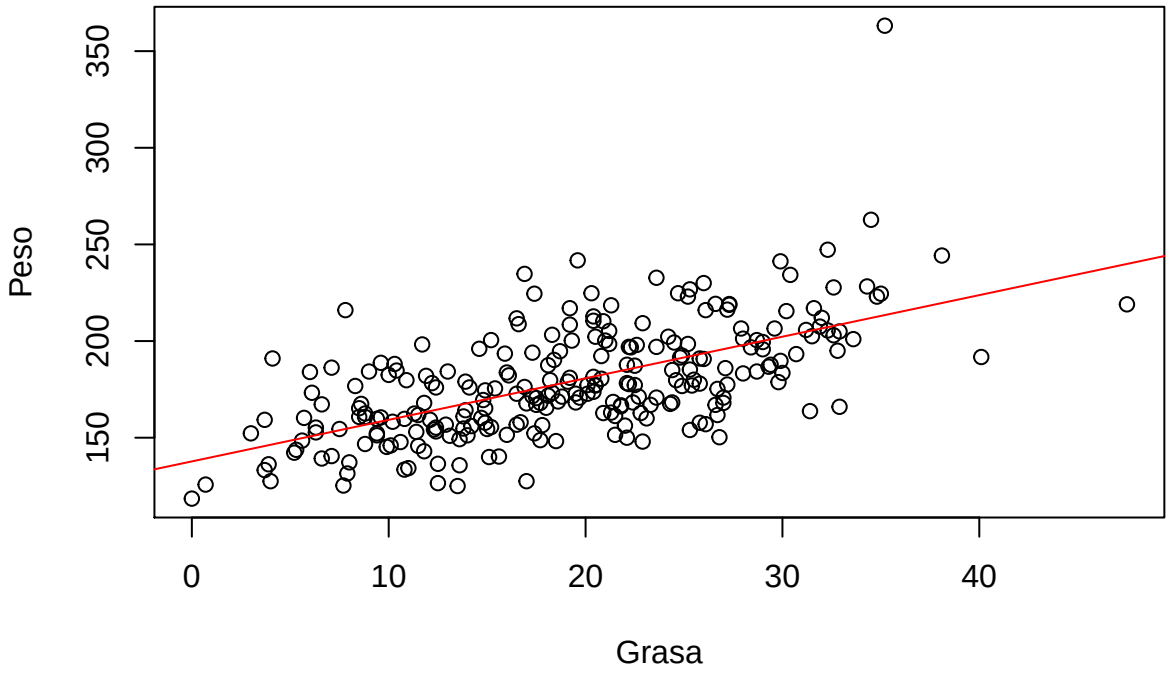
Como podemos observar, las dos formas de llamar a la función `lm` dan exactamente los mismo.

El resultado obtenido en ambos caso significa que la recta de regresión para nuestro datos es.

$$\tilde{y} = 2.151x + 137.738$$

donde y es el peso y x es la grasa. Ahora podemos superponer esta recta a nuestro gráfico anterior haciendo uso de la función `abline()`.

```r
plot(body2)
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 $(x_i, y_i) \quad i = 1, ..., n$ ha sido pueramente descriptivo.

Es decir, hemos mostrado que los datos son consistente con una función lineal, pero no hemos mostrado 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 si, podemos utilizar lo hecho hasta ahora para predecir valores $\tilde{y}_i$ en función de los valores x_i resolviendo una simple ecuación lineal.

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 $R^2 > 0.9$, consideramos que el ajuste es bueno. De lo contrario, no.

La funcion `summary` aplicada a `lm` nos muestra los contenidos de este objeto. Entre ellos encontraremos `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`.

```
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
```

Si hacemos la siguiente instrucción vamos al elemento que nos interesa.

```
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.

Tranformaciones Logarítmicas

No siempre encontraremos dependencias lineales. Aveces nos encontraremos otro tipo de dependencias, como por ejemplo potencias o exponenciales.

Estas se pueden transformar a lineales mediante un **cambio de escala**.

Por lo general, es habitual encontrarnos gráficos con sus ejes en una **escala lineal**. Es decir, las marcas de los ejes estan igualmente espaciadas.

A veces, es conveniente dibujar alguno de los ejes en **escala logarítmica**, de modo que la misma distancia entre las marcas significa el mismo cociente entre sus valores.En otras palabrasm un eje en escala logaritmica representa el **logaritmo** de sus valores en escala lineal.

Diremos que un gráfico está en **escala semilogaritmica** cuando su eje de abcisas está en escala lineal y el de las ordenadas, en escala logaritmica.

Diremos que un gráfico está en **escala doble logarítmica** cuando ambos ejes están es escala logarítmica.

Interpretación Gráfica

Si al representar unos puntos $(x_i, y_i) \quad i = 1, ..., n$ en escala semilogarítmica observamos que siguen aproximadamente una recta, esto querrá decir que los valores $\log(y)$ siguen una ley aproximadamente lineal en los valores x y por lo tanto y sigue una **ley aproximadamente exponencial** en x.

En efecto si $\log(y) = ax + b$, entonces,

$$y = 10^{\log(y)} = 10^{ax+b} = 10^{ax} \cdot 10^b = \alpha^x \beta$$

con $\alpha = 10^a$ y $\beta = 10^b$

SI al representar unos puntos $(x_i, y_i) \quad i = 1, ..., n$ en escala doble logaritmica observamos que siguen aproximadamente una recta, esto querra decir que los valores $\log(y)$ siguen una ley aproximadamente lineal en los valores $\log(x)$, y por lo tanto que y sigue una **ley aproximadamente potencial** en x.

En efecto, si $\log(y) = a \log(x) + b$, entonces, por propiedades de logaritmos.

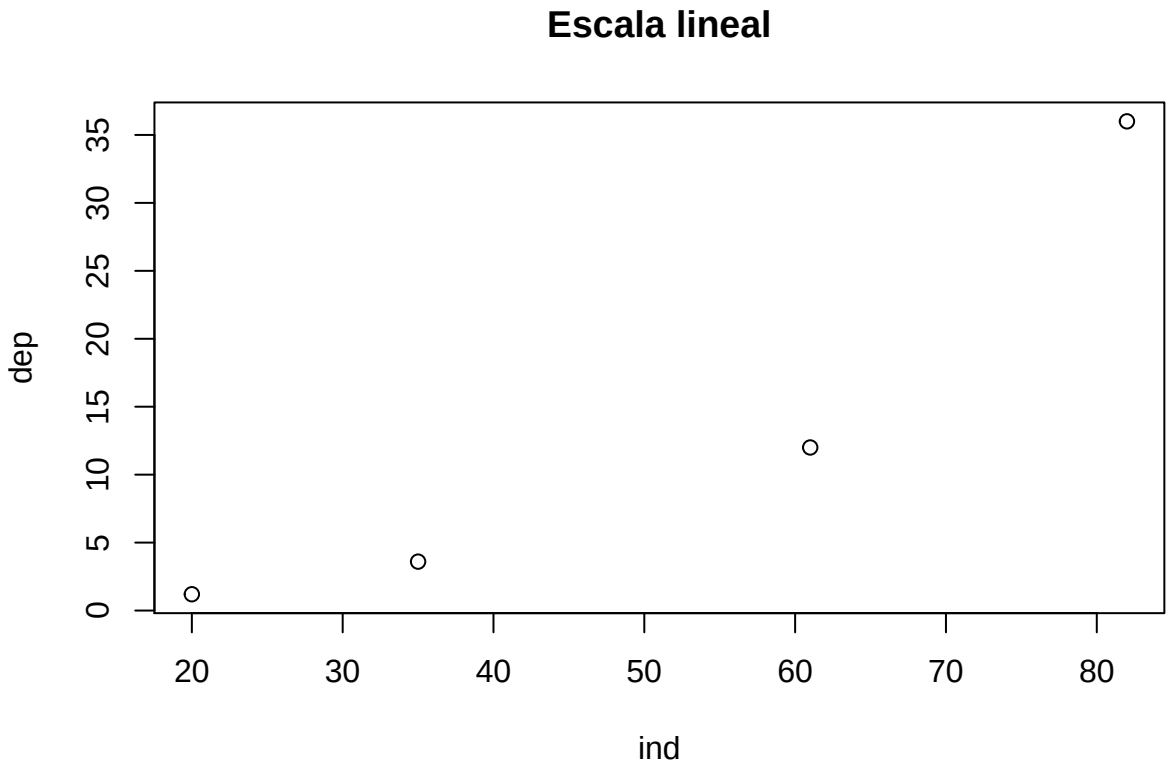
$$y = 10^{\log(y)} = 10^{a \log(x)+b} = (10^{\log(x)})^a \cdot 10^b = x^a \beta$$

con $\beta = 10^b$

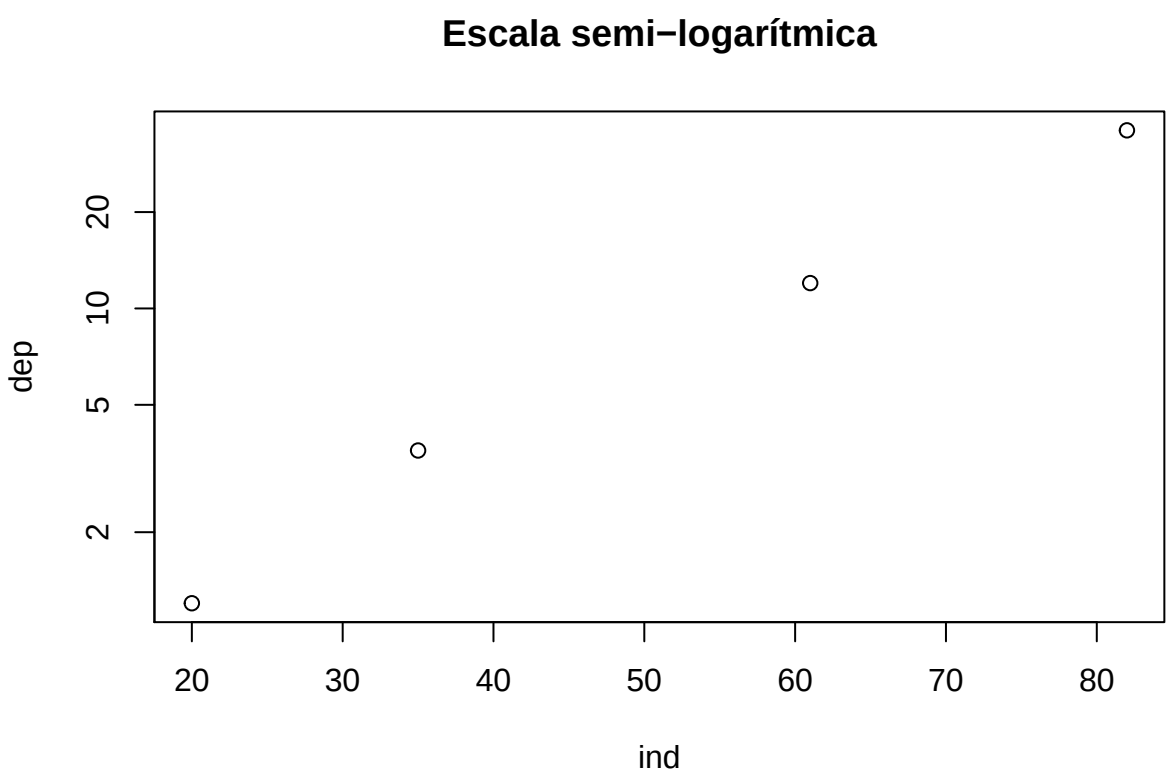
Ejemplo

En este caso trabajaremos no con un data frame, si no directamente con los dos siguientes vectores.

```
dep = c(1.2,3.6,12,36)
ind = c(20,35,61,82)
plot(ind,dep, main = "Escala lineal")
```




```
plot(ind,dep, log = "y", main = "Escala semi-logarítmica")
```




```
lm(log10(dep)~ind)
```

```
##
## Call:
## lm(formula = log10(dep) ~ ind)
##
## Coefficients:
## (Intercept)          ind
##    -0.32951      0.02318
```

```
summary(lm(log10(dep)~ind))$r.squared
```

```
## [1] 0.9928168
```

Podemos afirmar que en efecto el modelo lineal que ajusta el logaritmo en base 10 de la variable dependiente en función de la independiente es en efecto una recta.

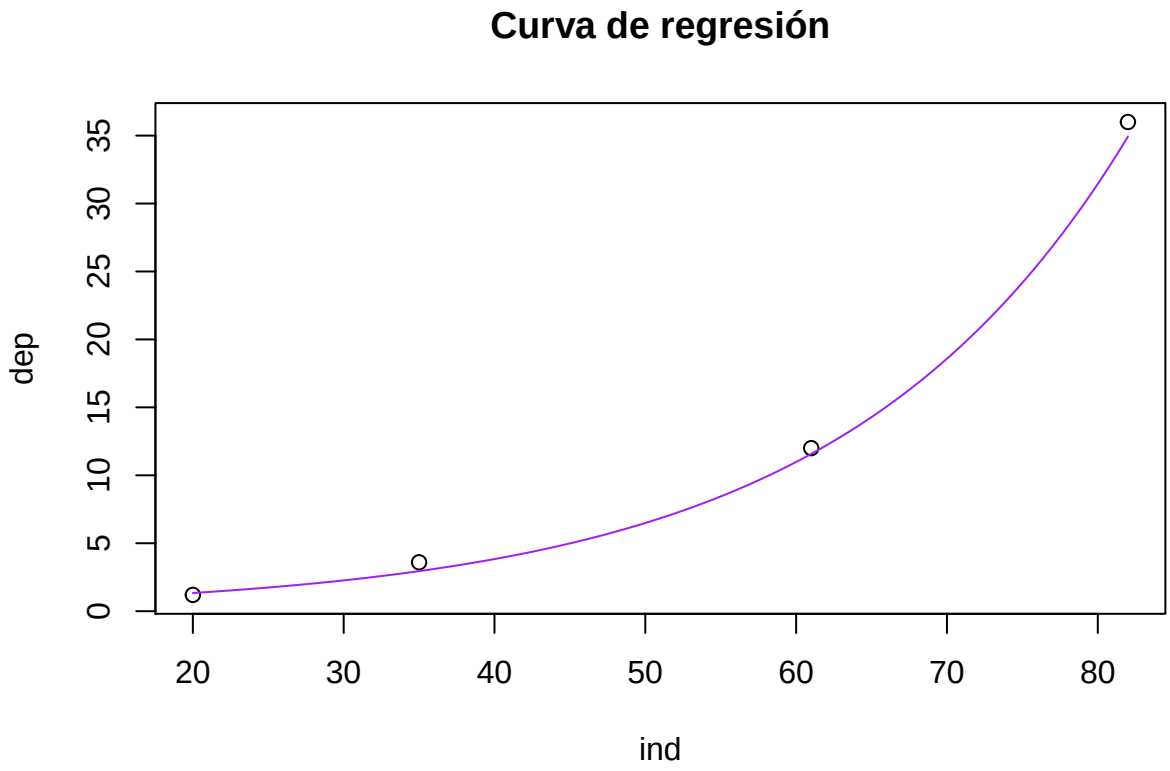
Lo que acabamos de obtener es que

$$\log(dep) = 0.023 \cdot ind - 0.33$$

es una buena aproximación de nuestro datos. Con lo cual obtenemos el siguiente modelo exponencial.

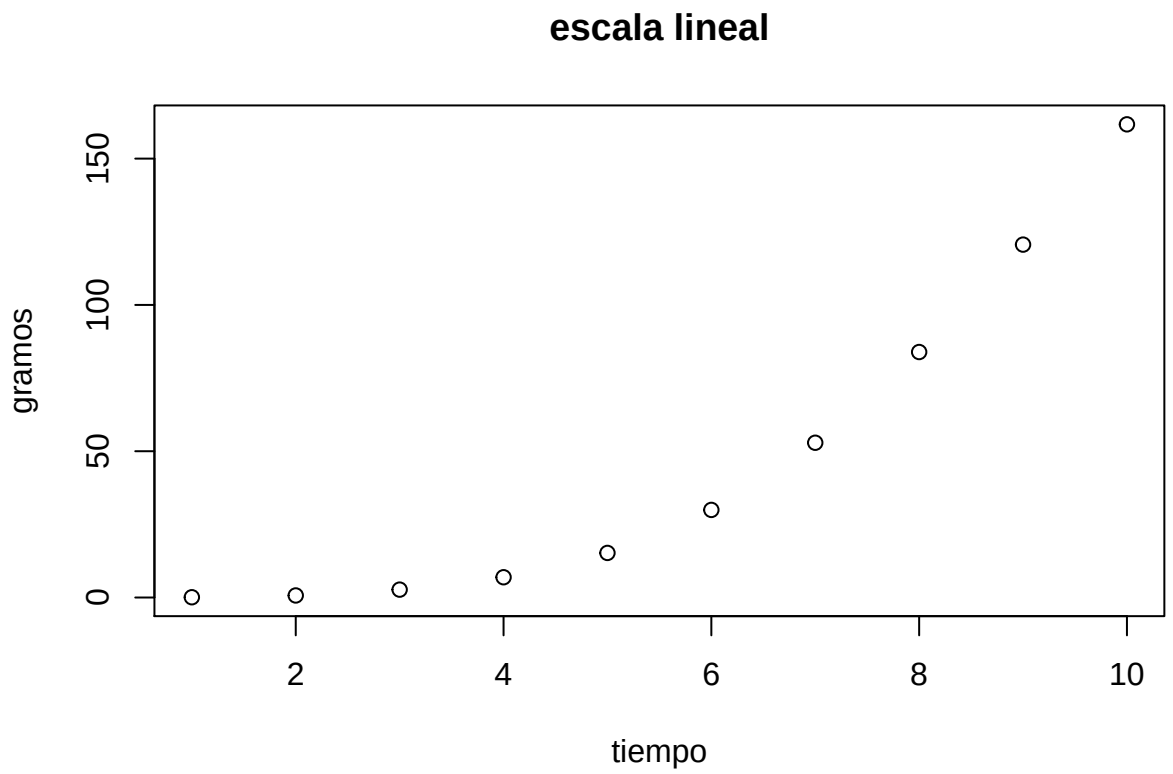
$$dep = 10^{0.023 \cdot ind} \cdot 10^{-0.33} = 1.054^{ind} \cdot 0.468$$

```r
plot(ind,dep, main = "Curva de regresión")
curve(1.054^x * 0.468, add = TRUE, col="purple")
```

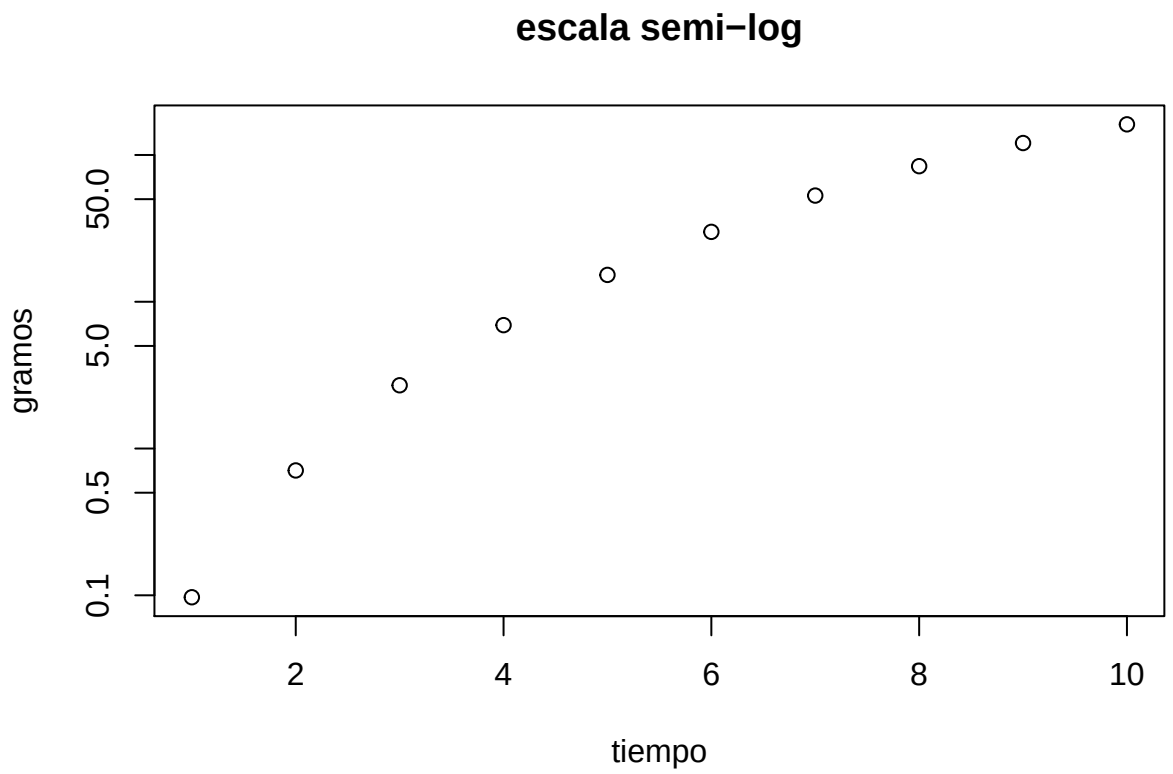


Ejemplo

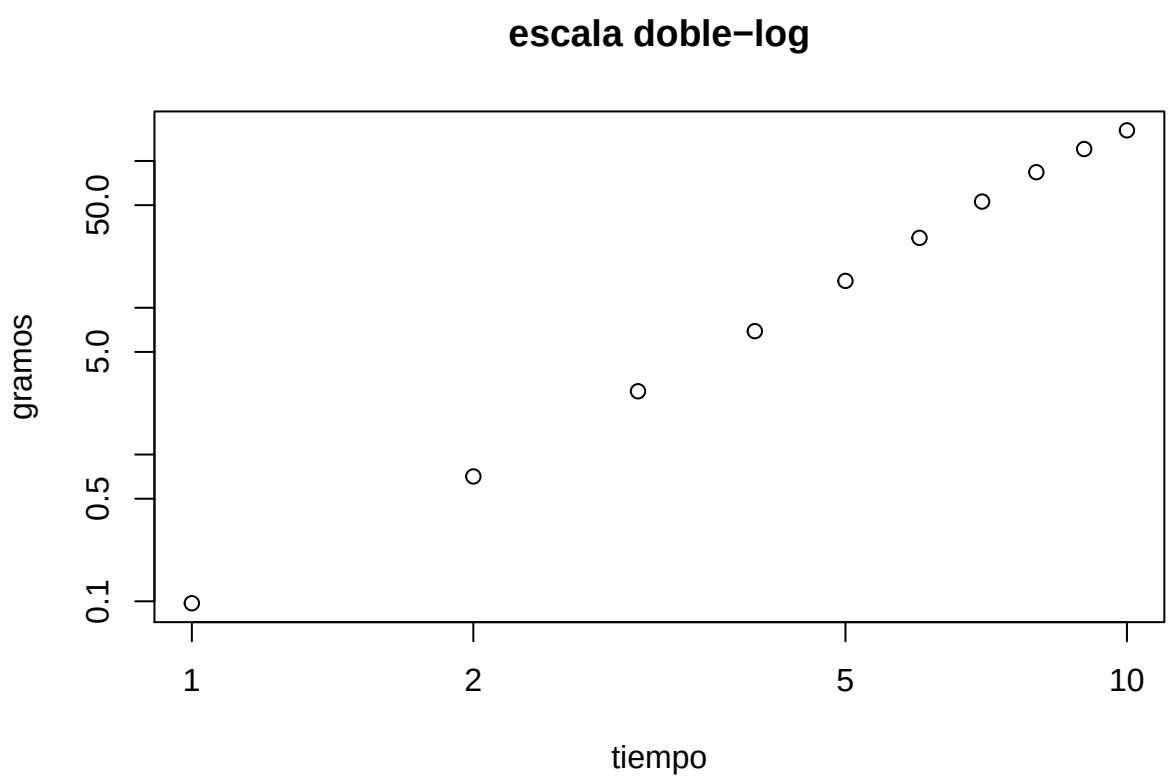
```r
tiempo = 1:10
gramos = c(0.097,0.709,2.698,6.928,15.242,29.944,52.902,83.903,120.612,161.711)
d.f = data.frame(tiempo,gramos)

plot(d.f, main = "escala lineal")
```




```
plot(d.f, log="y", main = "escala semi-log")
```




```
plot(d.f,log = "xy", main = "escala doble-log")
```




```
lm(log10(gramos)~log(tiempo), data = d.f)
```

```
##
## Call:
## lm(formula = log10(gramos) ~ log(tiempo), data = d.f)
##
## Coefficients:
## (Intercept)  log(tiempo)
##      -1.093        1.432
```

```
summary(lm(log10(gramos)~log(tiempo), data = d.f))$r.squared
```

```
## [1] 0.9982009
```

Lo que acabamos de obtener es que

$$\log(gramos) = 3.298 \cdot \log(tiempo) - 1.093$$

es una buena aproximación que sigue un **modelo potencial.** Con lo cual

$$gramos = 10^{3.298 \cdot \log(tiempo)} \cdot 10^{-1.093} = tiempo^{3.298} \cdot 0.081$$

```r
plot(d.f, main = "Curva de regresión")
curve(x^(3.298)*0.081, add = TRUE, col="purple")
```