

## Apuntes sobre el programa “R”: *Instalación e introducción al uso*

En el primer apartado de la primera práctica se pide realizar una regresión lineal de una serie de datos experimentales. Existen muchos programas (software), tanto comerciales como “freeware”, que se podrían utilizar para cumplir este fin. Aquí se describe un freeware: el programa “R”. Es un programa que se usa en los ámbitos académico y profesional para computación estadística y gráficos. Tras explicar el sencillo proceso de instalación del programa, se da una pequeña introducción hasta el nivel que nos va a hacer falta este software en la primera práctica. Se encuentra disponible en internet amplia información (por ejemplo, tutoriales o videos en youtube) para el alumno interesado en continuar su aprendizaje de “R”.

### 1. Instalación

A continuación se describe cómo instalar el programa “R” para Windows. Si eres usuario de Mac o Linux, también existen versiones del programa para tu sistema operativo. Las características del programa R (que se exponen en la sección 2 “Introducción al uso”), en general, serán independientes del sistema operativo. Aun así puede que haya pequeñas diferencias con respecto, por ejemplo, al formato de la ventanilla principal y los componentes de sus menús.

*Paso 1.* Vamos a la dirección <http://ftp.cixug.es/CRAN/>

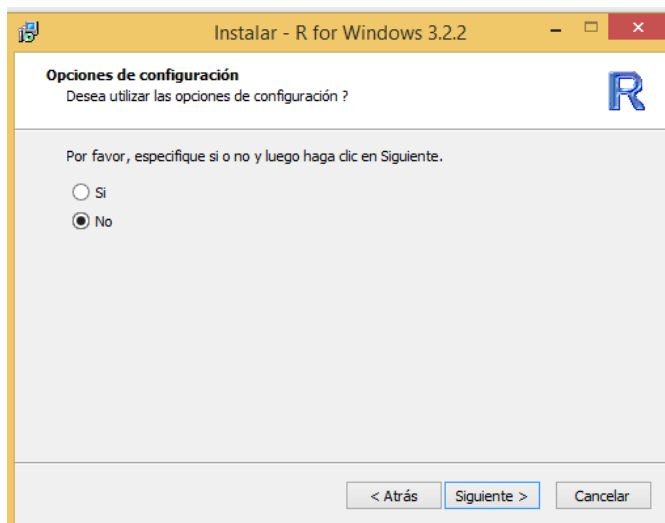
*Paso 2.* Pinchamos en [Download R for Windows](#)

*Paso 3.* Pinchamos en [install R for the first time](#)

*Paso 4.* Pinchamos en [DownloadR 3.2.2 for Windows](#)

*Paso 5.* Pinchamos en “Ejecutar” en la ventanilla que sale (probablemente en la parte de debajo de la pantalla):

Durante el proceso de instalación, contestamos “No” si sale la siguiente ventanilla:



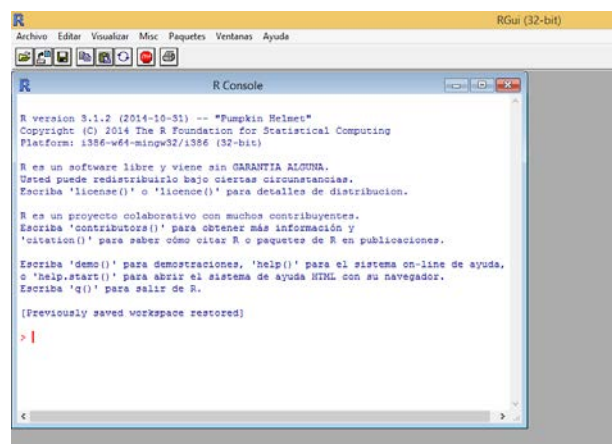
Una vez terminada la instalación, puede que encontremos dos iconos del programa en el escritorio. Si es así, corresponderán a dos versiones: la versión “i386” y la versión “x64”. Usamos la primera y borramos el segundo icono.



## 2. Introducción al uso

Cuando abrimos el programa “R”, sale automáticamente una ventanilla titulada en inglés “R console” – es decir, la consola de R – que es una interfaz de LÍNEA DE COMANDOS. En esta ventanilla (o interfaz) introducimos órdenes (o “comandos”) línea por línea.

Como uso más sencillo, podemos utilizar el programa R como una calculadora, donde introducimos operaciones en la línea de comandos, por ejemplo “uno más uno” y le damos a la tecla RETURN:



```
[Previously saved workspace restored]
```

```
> 1+1
[1] 2
> |
```

y el programa nos da el resultado. Dos! Como otro ejemplo, calculamos el seno de 90 (el seno se escribe “sin” en vez de “sen” porque el programa reconoce solamente los nombres matemáticos del sistema anglosajón):

```
> sin(90)
[1] 0.8939967
> |
```

Vemos que no nos sale “uno” como resultado, lo que nos indica que la función ha interpretado el valor (de noventa) en radianes. Para confirmarlo, calculamos el seno de  $\pi/2$ :

```
> sin(pi/2)
[1] 1
> |
```

Incrementando un poco el grado de dificultad, podemos usar “R” para guardar valores numéricos en la forma de variables. Por ejemplo, definimos la variable “x” igual a uno:

```
> x=1
> |
```

Luego para comprobar que “R” ha guardado correctamente el valor de x en su memoria, tecleamos “x” y le damos a RETURN:

```
> x
[1] 1
```

Ya que tenemos x definido, podemos definir una variable nueva “y” como función de “x”:

```
> y = 2 * x
> y
[1] 2
```

y una tercera variable “z”, función de “x” e “y”:

```
> z = x + y
>
> z
[1] 3
```

Como hemos dicho al principio, el propósito de nuestro uso de “R” en el laboratorio será la representación gráfica de una serie de datos experimentales y la realización de la regresión lineal de esta serie. Como ejemplo, imaginemos una serie de valores de una magnitud hipotética “x”, *la variable independiente* en un “experimento”. Esta serie la guardamos en R incluyendo los valores entre paréntesis y separados por comas:

```
> x = c(1,2,3,4,5)
```

(Por cierto, como separador decimal se usa un punto en R, y no una coma.) Siempre va la letra “c” antes de los paréntesis cuando se define una serie (o “vector”) de valores. Si tecleamos “x” podemos comprobar que está guardado correctamente en la memoria:

```
> x
[1] 1 2 3 4 5
```

Nos podemos referir a la serie de valores de “x” como el “vector x”. A continuación podemos generar el vector de los valores de una hipotética *variable dependiente*, “y”. Por ejemplo, “y” igual a dos por “x”:

```
> y = 2 * x
```

y comprobamos que se ha creado correctamente:

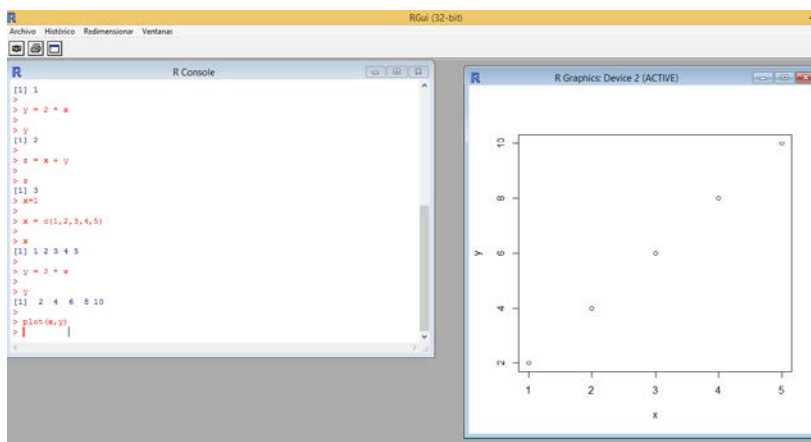
```
> y
[1] 2 4 6 8 10
```

Ya podemos pintar el gráfico de  $y$  frente a  $x$ , utilizando el comando de R "plot":

```
> plot(x,y)
```

y vemos que sale automáticamente una ventanilla con la gráfica.

La función  $y = 2x$  es un ejemplo de una relación lineal. Es decir, es la función de una línea recta cuya pendiente es 2 y cuya ordenada es nula.



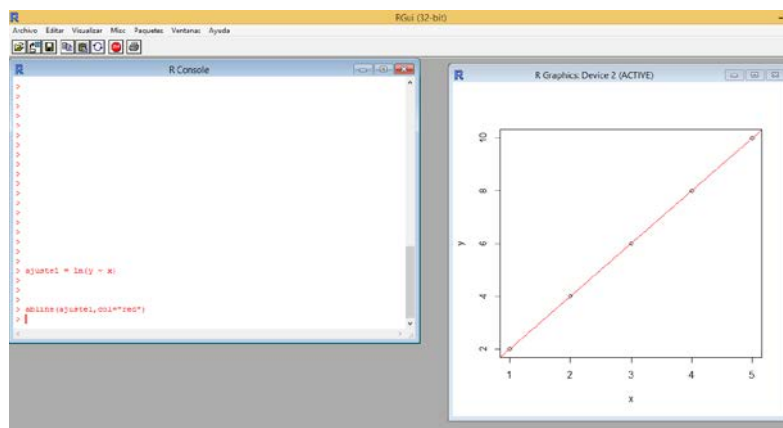
Vamos a realizar una regresión lineal de estos datos ficticios, "y" frente a "x". Para ello, usamos la función (o comando) "lm" que quiere decir (en inglés) "Linear Model". Dentro de los paréntesis de la función escribimos primero la variable dependiente (en este caso "y"), seguida por "un tilde de eñe", y después la variable independiente (en es este caso "x"):

```
> ajustel = lm(y ~ x)
```

En este ejemplo hemos guardado el resultado de la regresión en un "objeto" llamado "ajustel" (podemos elegir el nombre que más nos guste). Para pintar la línea de regresión sobre el gráfico anterior, usamos el comando "abline" con, como primer argumento, el nombre del objeto donde hemos guardado el resultado de la regresión, y, como segundo, el color que queramos para esta línea:

```
> abline(ajustel,col="red")
```

Y vemos que se le añade una línea roja a la gráfica anterior.



Hay muchos parámetros correspondientes a una regresión lineal. El comando "summary" nos aporta todos ellos:

```
> summary(ajustel)

Call:
lm(formula = y ~ x)

Residuals:
    1        2        3        4        5 
-2.385e-17  6.159e-17  1.411e-17 -1.176e-16  6.572e-17 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.332e-15   9.124e-17 -1.46e+01  0.000696 ***
x            2.000e+00   2.751e-17  7.27e+16  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.699e-17 on 3 degrees of freedom
Multiple R-squared:  1, Adjusted R-squared:  1 
F-statistic: 5.286e+33 on 1 and 3 DF, p-value: < 2.2e-16

Mensajes de aviso perdidos
In summary.lm(ajustel) : essentially perfect fit: summary may be unreliable
```

En la práctica 1.A. los parámetros de la regresión que nos van a interesar son solamente la pendiente y su error. En el ejemplo arriba el valor de la pendiente está circulado en rojo y su error estándar está circulado en verde. Los comandos que se usan para solamente acceder a estos parámetros son los siguientes:

```
> coef(ajustel)[2]
x
2
```

para ver la pendiente, y

```
> summary(ajustel)$coef[4]
[1] 2.750893e-17
```

para ver el error estándar de la pendiente. En nuestro ejemplo el valor del error es tan pequeño (2.750893e-17) comparado con el de la pendiente (2) porque nuestros datos ficticios "x" e "y" constituyen una relación *exactamente* lineal.

Todos los comandos que vayamos ejecutando en la ventanilla "R console" los podemos guardar en un fichero (tipo ASCII) de R, lo que llamaremos un "script de R". Se ha preparado un script con todos los comandos necesarios para que el alumno pueda introducir sus datos experimentales, pintarlos y realizar la regresión lineal correspondiente. El script está disponible en la carpeta "Ficheros para el programa R" situada el apartado LABORATORIO de Moodle, con el nombre "Practical.R".

- El alumno bajará este fichero a una carpeta local de su portátil antes del día de la primera práctica. Durante la práctica lo abrirá dentro del programa R seleccionando "Abrir script..." del menú "Archivo".
- Una herramienta útil es la de "Divida Verticalmente" en el menú "Ventanas". Ésta nos separará el "R console" del "script", posicionándoles a la izquierda y a la derecha de la pantalla, respectivamente (ver la Figura 2, abajo).
- Lo único que el alumno tendrá que hacer antes de ejecutar el script es introducir los valores experimentales de intensidad y voltaje en los vectores "I" y "V" respectivamente (en las posiciones indicadas con las flechas en la Figura 3, abajo). De nuevo, se le recuerda al alumno que los valores van separados por comas (las comas ya están escritas) y se usa un punto como separador decimal en cada valor

(y no una coma). Finalmente se selecciona “Ejecutar todo” del menú “Editar” (Figura 4).

En la carpeta “Ficheros para el programa R” de Moodle, el alumno encontrará también un vídeo (archivo .mp4) que muestra una “Demostracion del uso de Practica1.R”.

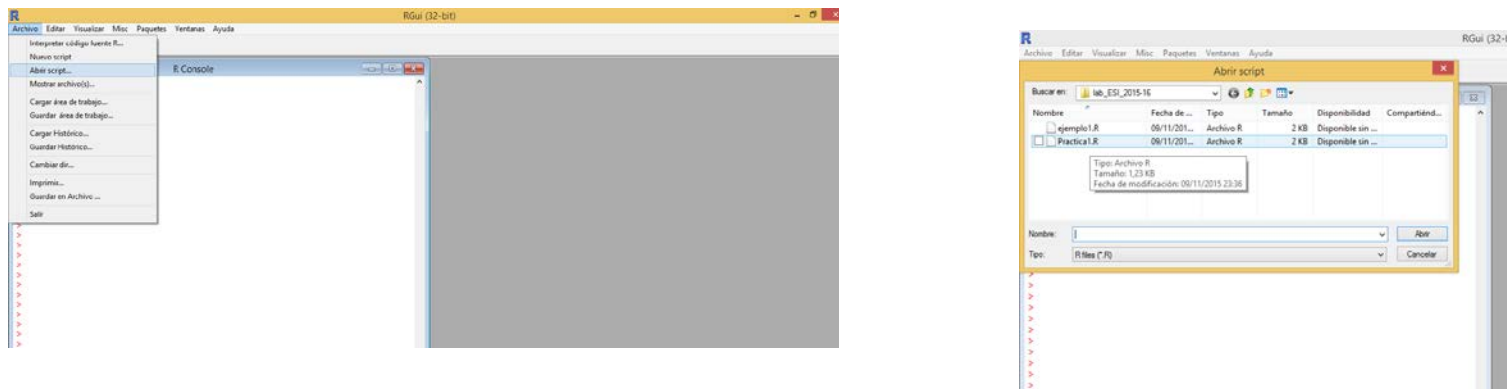


Figura 1.

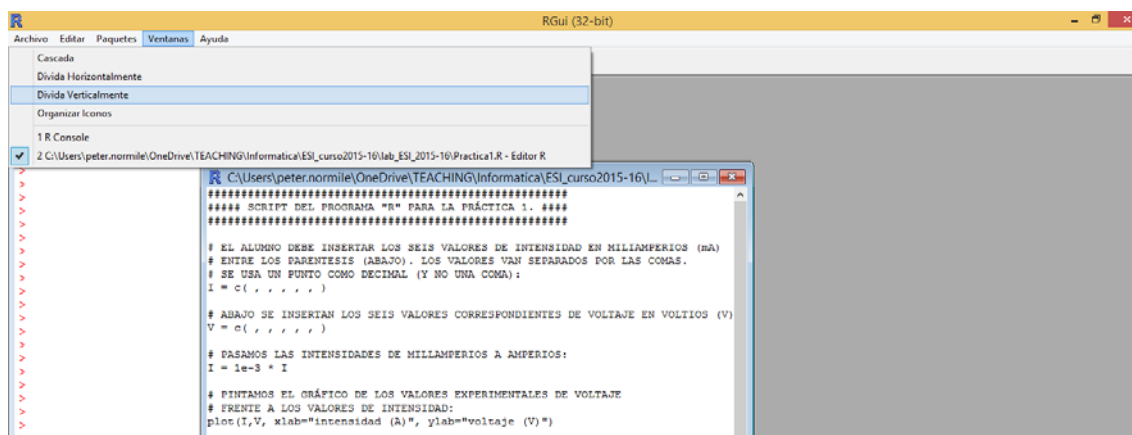


Figura 2.

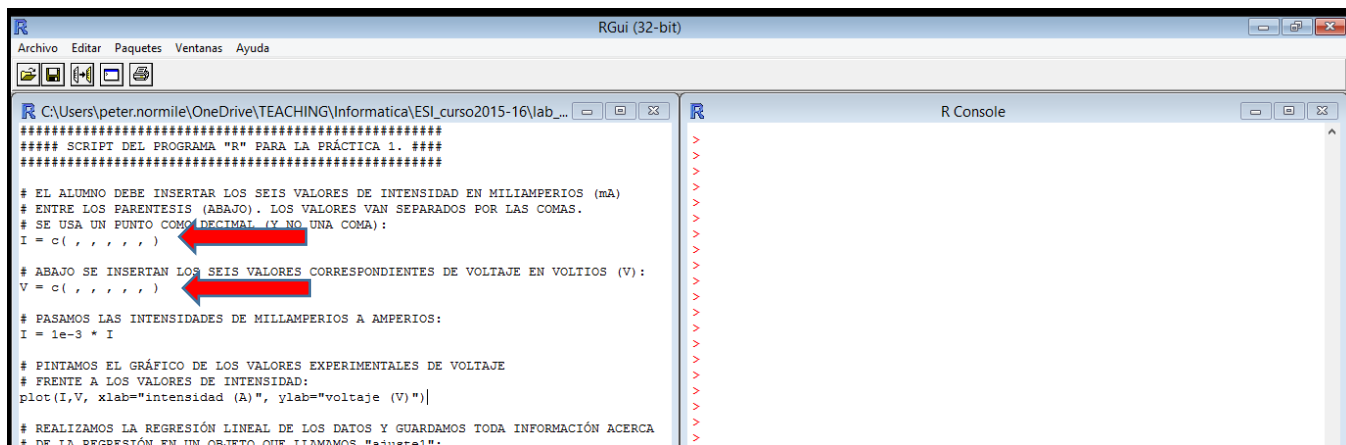


Figura 3.

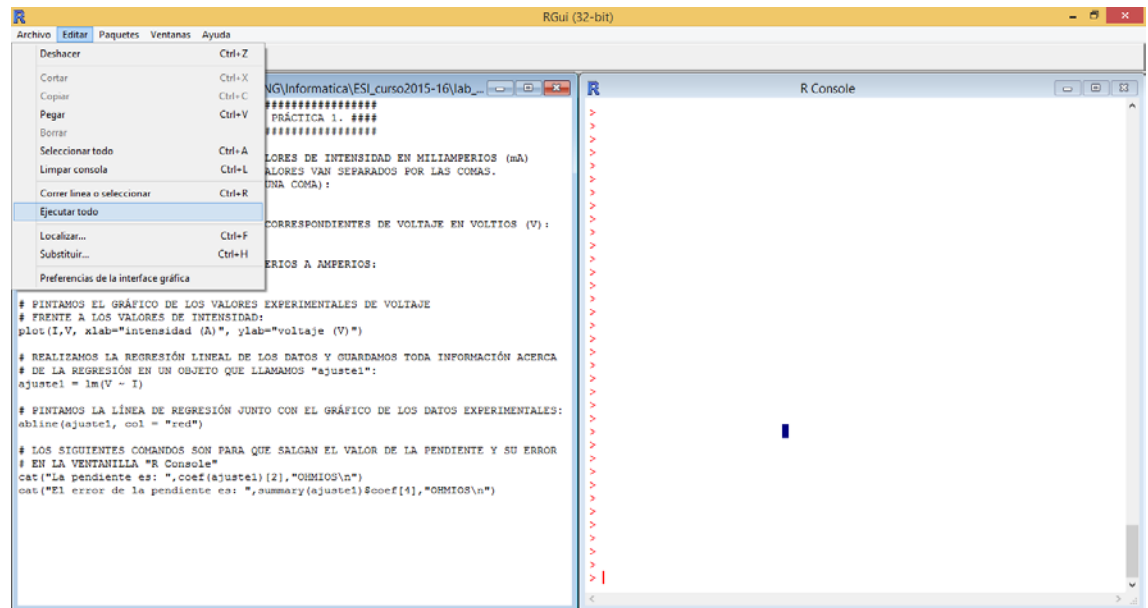


Figura 4.