

# Módulo 2. Entorno RStudio

Curso Herramientas para Data Science

*Prof. Jose Jacobo Zubcoff Vallejo, PhD  
Universidad de Alicante*

*2017 @ Licencia Creative Common BY*

## Módulo 4. El entorno RStudio.

### Lección 4.1 Entorno de trabajo

La interfaz de usuario para R denominado RStudio IDE (le llamaremos RStudio en adelante) <https://www.rstudio.com> es una aplicación gratuita (y open source con licencia AGPL <https://www.gnu.org/licenses/agpl-3.0.en.html>) y está disponible para varios sistemas operativos. El uso de entornos integrados (que integran herramientas y utilidades) facilitan el desarrollo de análisis con lenguajes como R. Además, añade varias características útiles como tener en un simple vistazo varios paneles en una misma ventana, el uso de proyectos para agrupar tareas de análisis, la selección amigable de paquetes o recorrer el histórico de comandos o de gráficos, etc.

RStudio permite tener hasta 4 paneles abiertos (ver figura 1) con distintas opciones, como la consola de R, un visor de gráficos, un editor de comandos con habilidad para resaltar sintaxis de R y facilitar la escritura de comandos de R sugiriendo funciones, la ayuda, la gestión de paquetes instalados, editor de guiones (scripts), el histórico de comandos o los datos en memoria, entre otros. Así, el entorno RStudio permite mantener a la vista toda la información necesaria para el análisis.

RStudio tiene una interfaz similar en todos los sistemas operativos (Windows, Mac OS X y Linux).

#### El entorno de RStudio

La interfaz de RStudio puede estar dividida en cuatro paneles, aunque puede personalizarse maximizando o minimizando algunos de ellos:

- consola de comandos de R
- panel de guiones de R
- panel de variables de entorno y cargadas en memoria, historial de comandos, etc.
- panel de carpetas y ficheros, paquetes de R, ayuda, y gráficos.

Se puede seleccionar la cantidad de paneles abiertos, minimizando o ampliando algunos de ellos y se pueden intercambiar algunas pestañas entre los paneles visibles desde las preferencias de RStudio.

#### Consola de comandos y resultados

En la consola (Figura 2) podemos introducir directamente los comandos, además, es donde aparecerán los resultados que no sean gráficos. También se puede seleccionar una parte de guión de comandos y ejecutarlos en la consola (a través de las opciones de menú, la barra de herramientas del propio panel o con atajos de teclado).

#### Histórico de comandos y datos en memoria

En R las variables se almacenan en memoria mientras dure la sesión. RStudio mantiene el histórico de comandos para la sesión. Además, mantiene a la vista las variables cargadas en memoria. Todo esto está en un panel, generalmente situado a la derecha como se puede observar en la figura 3, que contiene varias

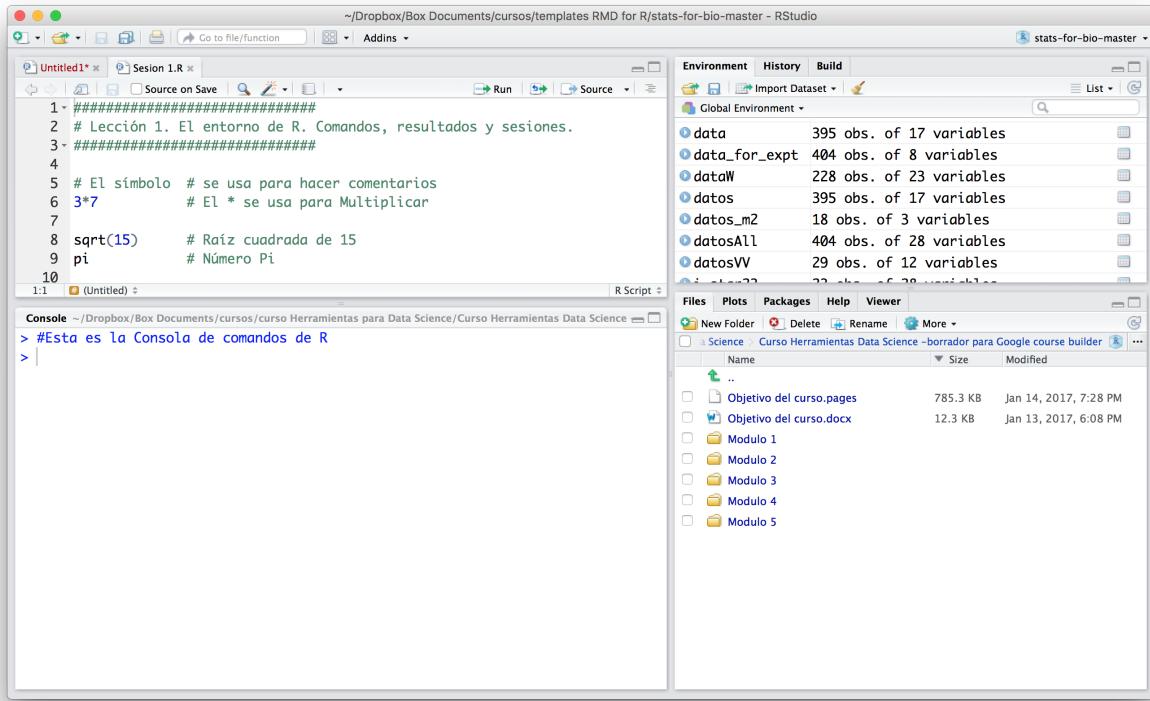


Figure 1: Interfaz de RStudio

pestañas: Environment (donde están las variables en memoria), History (donde está el histórico de comandos) y Build (para opciones de desarrollo).

### Files (pestaña de ficheros)

En la pestaña “Files” se puede navegar por las carpetas y ficheros de nuestro equipo de trabajo (Figura 4). Se puede seleccionar o establecer el directorio de trabajo desde este panel (en la barra de herramientas del panel en la opción “More” o desde el menú “Session” + “Set Working Directory”). Así, facilita la gestión de guiones de comandos o scripts y proyectos.

### Packages (pestaña de paquetes)

En la pestaña Packages, que se encuentra en el mismo panel que ficheros, figuras y ayuda) aparecerán todos los paquetes instalados estén cargados en memoria o no. Se pueden cargar en memoria o descargar simplemente haciendo click en la casilla de verificación que hay al lado de cada paquete. También se puede instalar nuevos paquetes (desde el botón dedicado que hay en la barra de herramientas de esta pestaña o desde el menú). Tiene una opción de actualizar los paquetes. En el caso de querer actualizar uno o varios paquetes, se abrirá una ventana donde aparece la versión instalada y la última versión disponible de cada paquete instalado, se puede seleccionar individualmente los paquetes a actualizar o seleccionar todos.

### Help (pestaña de ayuda)

La pestaña de ayuda se encuentra en el mismo panel que ficheros, figuras y paquetes. Esta pestaña es un acceso directo a la ayuda de R. Permite hacer consultas sobre comandos y paquetes, y mantener visible

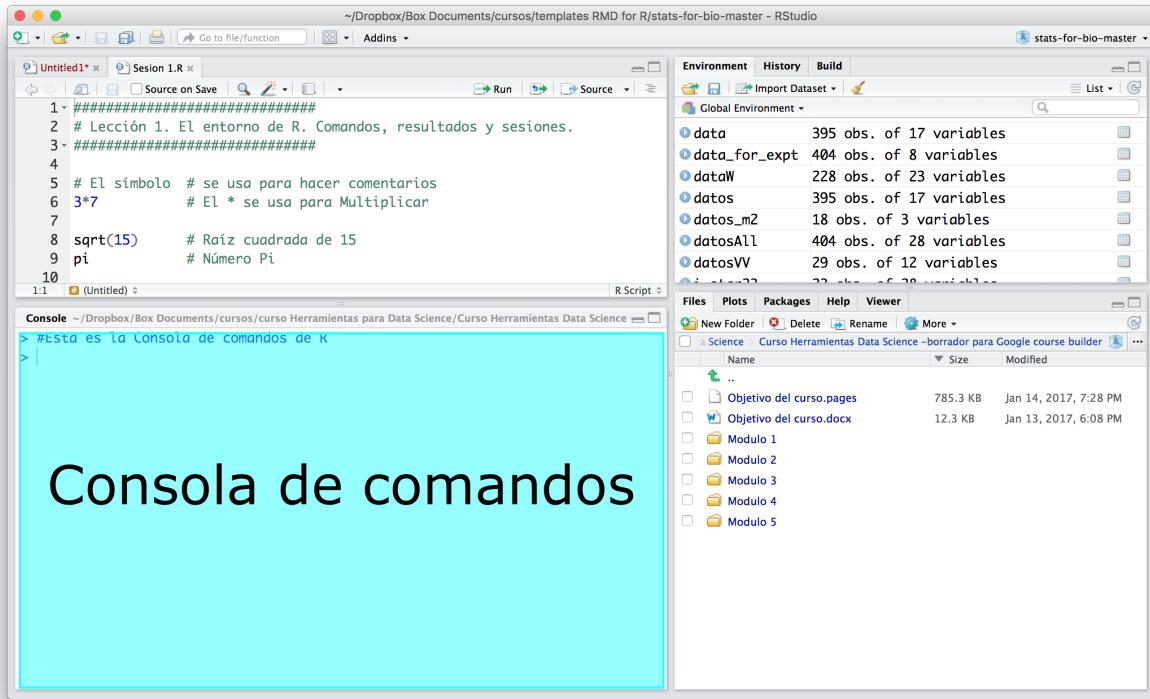


Figure 2: Panel de la consola de comandos dentro de RStudio

la ayuda junto con otros paneles, por ejemplo para visualizar la ayuda de una función mientras se está preparando el guión del análisis.

### Working directory (Directorio de trabajo)

El lenguaje R permite trabajar con ficheros. Estos ficheros, pueden ser el origen de datos, o bien, pueden ser guiones de comandos que se usan para un determinado análisis. De este modo, si el usuario prefiere trabajar con una estructura de ficheros en una carpeta y tener todos los ficheros (datos, guiones de comandos y otros ficheros) que sirvan para un análisis en una carpeta puede establecer lo que en R se denomina el “Working directory” (o directorio de trabajo). En RStudio se hace por medio de menú “Session, Set working directory” o bien en el panel de ficheros existe un botón denominado “More” que da las mismas opciones (5).

### Lección 2. Área de trabajo: variables, vectores y matrices. Importar datos

Todas las variables que se crean en una sesión de trabajo de R se van almacenando en memoria. RStudio permite tener un panel dedicado que muestra todas las variables que están actualmente en memoria. En la figura 6 se puede ver el panel del área de trabajo, las variables que hay en memoria y los botones disponibles.

Desde este panel se puede abrir un área de trabajo “Workspace” o guardar el área de trabajo actual. Además, se puede importar unos datos desde fuentes diversas (como CSV, Excel, SPSS, Stata, etc.), como se muestra en la figura 7.

Todas estas acciones también pueden llevarse a cabo desde opciones de menú (marcado como “a” en la Figura 7), en la opción “File Import”. Aunque resulta muy fácil acostumbrarse a usar los botones dedicados de los paneles que están siempre visibles (marcado como “b” en la Figura 7).

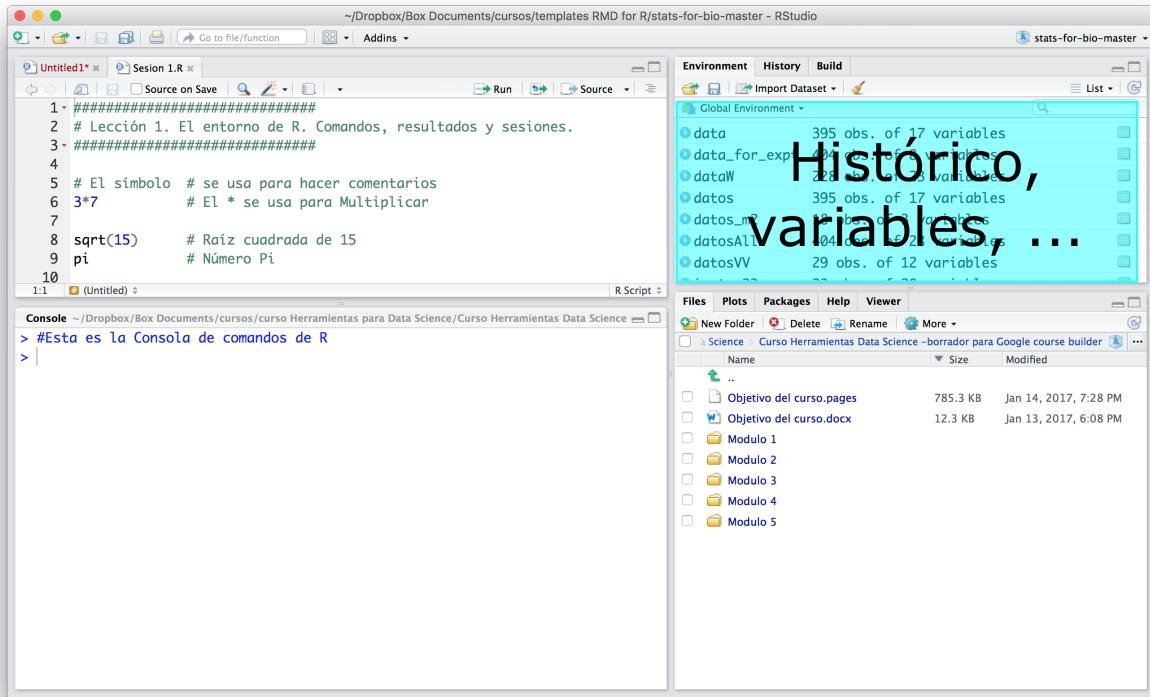


Figure 3: Panel de histórico de comandos de R y pestaña de variables en memoria en RStudio

Por ejemplo, para importar unos datos que están en una hoja de cálculo de Excel, aparecerá un cuadro de diálogo que permitirá elegir el fichero Excel, y una vez elegido se podrá seleccionar la hoja que se quiera importar. Conviene recordar que un libro de Excel puede tener muchas hojas disponibles, pero para importar los datos y almacenarlos en una matriz, solo podemos importar una hoja del libro de Excel. En el caso de trabajar con ficheros CSV solo hay que seleccionar el fichero porque solo tiene una “hoja” de datos.

En la figura 8 se puede ver el cuadro de dialogo que aparece cuando elegimos la opción de “Import Dataset” (del menú o desde el botón dedicado en el panel de “Environment”). En primer lugar debemos seleccionar el fichero (paso 1 de la figura). Una vez seleccionado el fichero Excel, debemos seleccionar la hoja de cálculo donde están los datos a importar (paso 2 de la figura). Hay algunas opciones para importar los datos, por ejemplo si la primera fila contiene nombres de las columnas, si queremos saltar las primeras n filas (se puede indicar el número de filas a saltar), o elegir cual es el valor cuyo significado es “NA” (“Not available”, o “No Disponible”). Es importante destacar que cada vez que cambiemos algo en las opciones de importación se refrescará la ventana denominada “Data Preview” y podremos “ver” lo que se pretende importar. Además, nos muestra en tiempo real cuál es el comando que se usará para importar los datos (número 3 en la figura 8). Una vez que hayamos seleccionado las opciones que consideremos adecuadas para la importación, procedemos a importar los datos usando el botón “Import” (paso 4 de la figura).

### Variables, vectores y matrices

El uso de RStudio implica que tenemos instalado una versión de R en nuestro ordenador. Así, en la consola podríamos introducir, por ejemplo, todos los comandos que ya conocemos del módulo 1 de este curso de herramientas para data science.

Podemos crear variables usando los operadores de asignación:

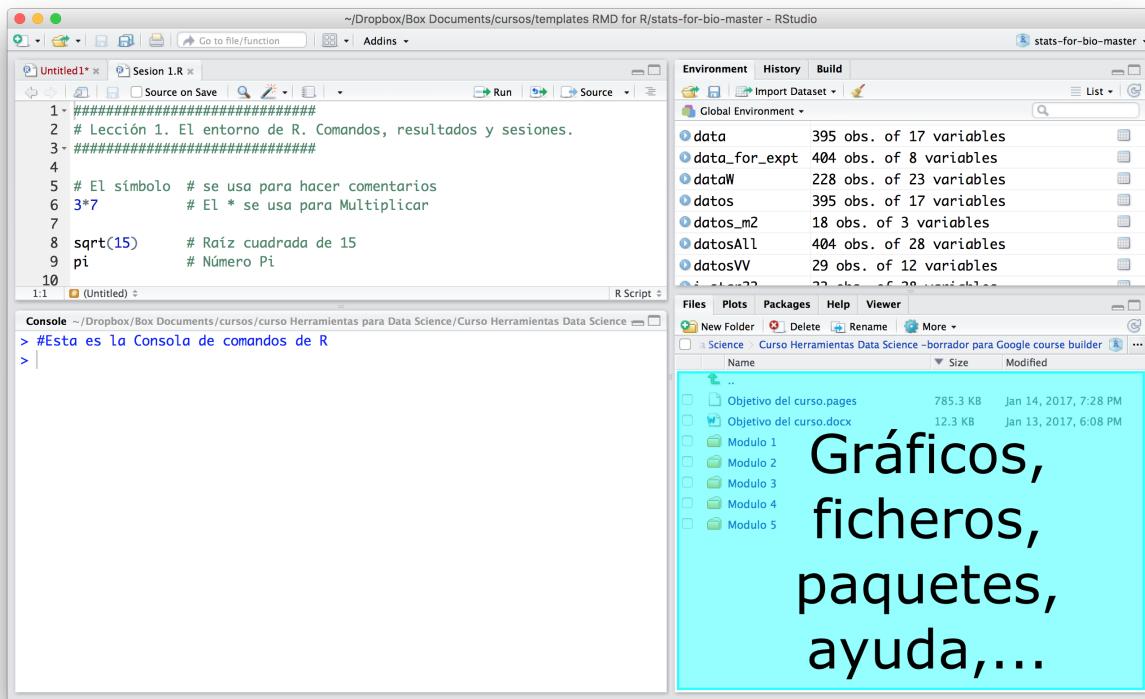


Figure 4: Panel de figuras, ficheros, paquetes y ayuda en RStudio

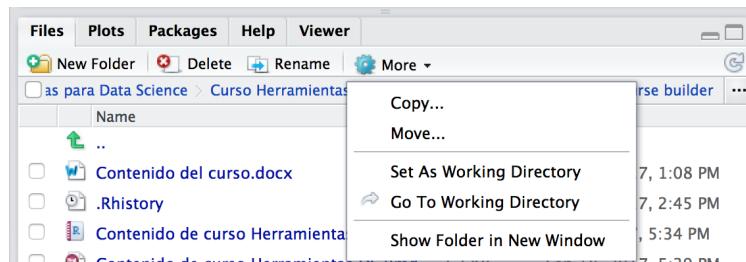


Figure 5: Working directory o directorio de trabajo en RStudio

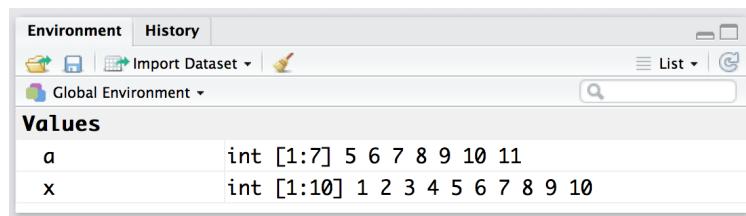


Figure 6: Environment o variables en memoria en RStudio

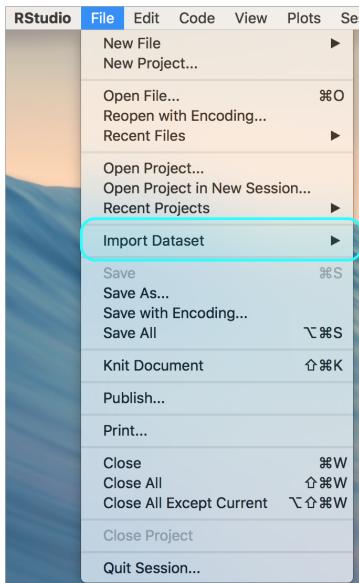
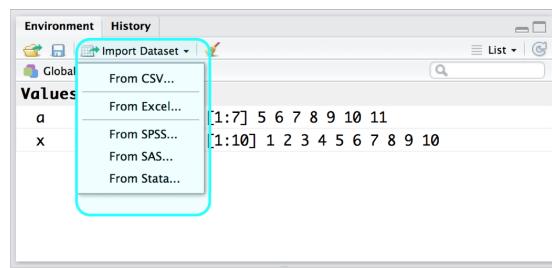
**a****b**

Figure 7: Importar datos en RStudio

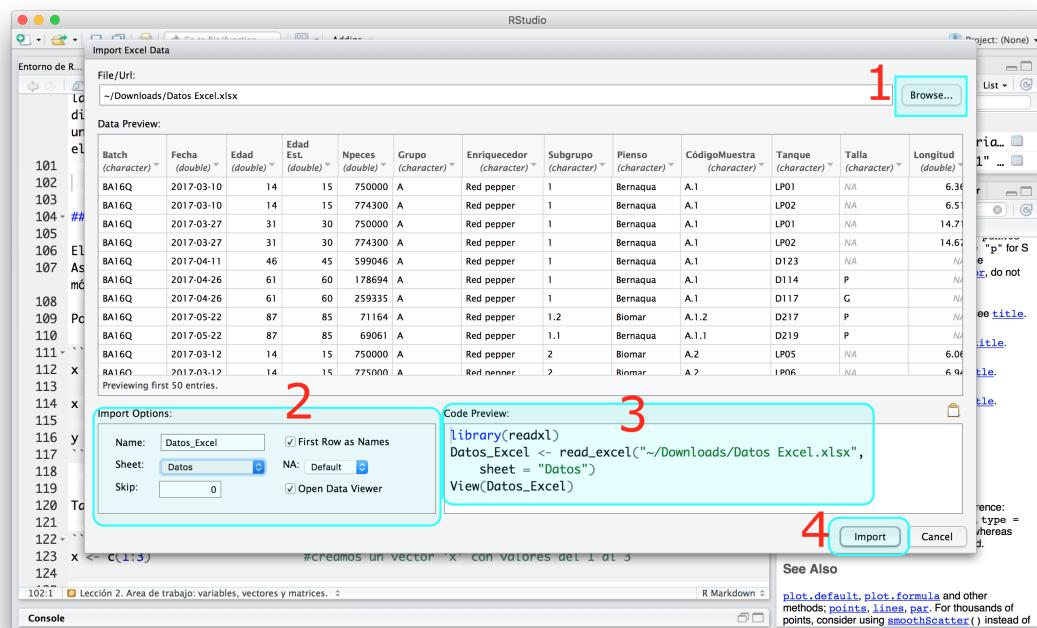


Figure 8: Importar datos desde Excel en RStudio

```

x <- 1                         #asignamos el valor 1 a 'x'

x <- 2*3                        #asignamos 6 a 'x' (resultado de multiplicar 2 * 3)

y <- TRUE                        #asignamos el valor TRUE o verdadero a 'y'

```

También se puede crear vectores usando operadores de asignación:

```

x <- c(1:3)                      #Creamos un vector 'x' con valores del 1 al 3

y <- c("a","b","c")                #asignamos los valores "a", "b" y "c" al vector 'y'

z <- matrix(cbind(x,y),nrow=3,ncol=2) #Creamos la matriz "z" con 3 filas y 2 columnas

```

## Lección 3: Gráficos

### Plots (pestaña de gráficos o figuras)

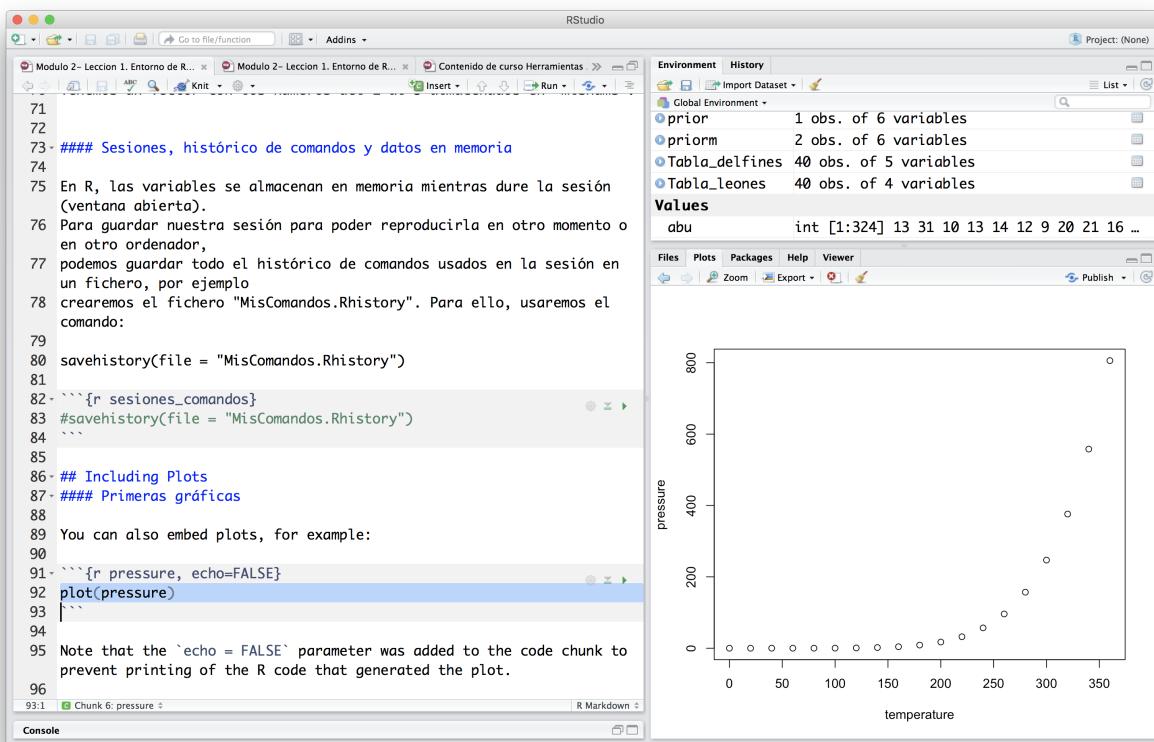


Figure 9: Panel de figuras en RStudio

Las figuras aparecen en la pestaña denominada Plots (Figura 9). Se puede ampliar (o hacer zoom) con el botón dedicado a ello y se mostrará el gráfico en una ventana a tamaño real. Además, permite navegar por las figuras que hayamos hecho durante la sesión. Esto puede ser de gran ayuda cuando estamos preparando un informe o cuando tenemos varios gráficos hechos en una misma sesión y necesitamos recorrer los últimos gráficos que hemos hecho.

Podemos guardar en algún formato para imágenes o en PDF usando la opción “Export” como se muestra en

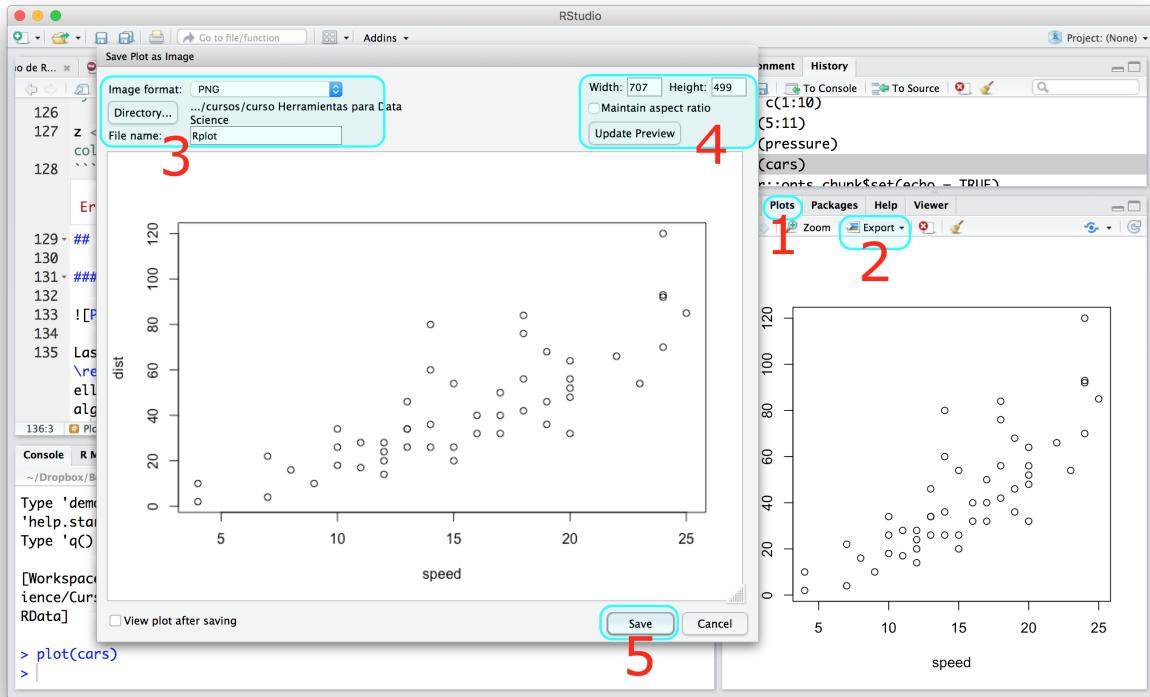


Figure 10: Guardar o exportar un gráfico en RStudio

la figura 10 o bien desde el menú “Plots”. Primero seleccionamos el panel de “Plots” (paso 1 de la figura 10), luego “Export” (paso 2 de la figura) y elegimos “Save as Image” para elegir las opciones del gráfico y del fichero de destino (pasos 3 y 4 de la figura 10). Finalmente, guardamos el gráfico (paso 5 de la figura).

## Lección 4. Scripts. Crear, ejecutar parcial/totalmente scripts

Se denomina script a un conjunto de comandos de R que se guardan en un fichero de texto. La ventaja de usar scripts (o guiones con comandos) es que podemos reproducir una sesión de trabajo cambiando los datos de entrada muy fácilmente. También es de utilidad cuando se hacen análisis similares y hay partes de guiones que podemos reutilizar en gran medida. Esto es una ventaja de R.

RStudio permite tener a la vista simultáneamente el script mientras observamos los resultados en la consola y vemos los gráficos o las variables en memoria, o la ayuda, todo a la vista.

Un script es un fichero de texto que contiene comandos y anotaciones para R. Para crear un script en RStudio, podemos hacerlo desde la opción de menú “File” + “New File” + “R Script” (figura 11).

Se abrirá una nueva pestaña vacía, donde se podrán introducir los comandos que se quieran almacenar como “Scripts” o guiones de comandos.

Para ejecutar parcial/totalmente scripts se puede hacer de tres formas: \* desde el botón dedicado “Run” \* desde la opción de menú “Code” + “Run selected line(s)” \* con atajo de teclado (Ctrl + R) en Windows ó (CMD + Intro) en Mac

Se pueden seleccionar la líneas que se quieren ejecutar, o bien se puede ejecutar todo el código, entre las opciones que permite RStudio.

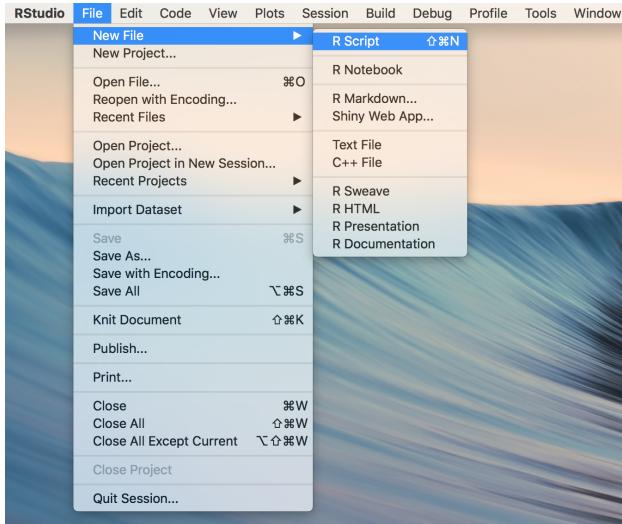


Figure 11: Crear un script en RStudio

The screenshot shows the RStudio interface with a script editor containing R code. The code is as follows:

```

27 ## Why do we need ANOVA models?
28
29 ````{r, echo=FALSE}
30 means <- c(1.5, 2.0, 1.8, 1.4, 1.9)
31 plans <- c("None", "Y-plan", "F-plan", "S1")
32 sim.wl <- function(variables) {
33   #
34   wloss.data <-
35     data.frame(
36       Plan = rep(plans, each = 8),
37       WeightLoss = round(rep(means, each =
38         0.4), 1),
39       stringsAsFactors = FALSE
40     ) %>% mutate(Plan = factor(Plan, levels =
41       plans))

```

To the right of the editor, a context menu is open under the 'Run' button in the toolbar. The menu is titled 'Run' and contains several options for running R code:

- Run Selected Line(s)
- Run Current Chunk
- Run Next Chunk
- Run Setup Chunk
- Run Setup Chunk Automatically
- Run All Chunks Above
- Run All Chunks Below
- Restart R and Run All Chunks
- Restart R and Clear Output
- Run All

Figure 12: Ejecutar guiones de comandos o scripts en RStudio

Es importante recalcar que el uso de guiones de comandos facilita la reproducibilidad de los análisis. Por ello es recomendable añadir comentarios a los comandos, para que quede documentado todo el proceso de análisis. Esto facilita la ciencia reproducible.

## Lección 5: Trabajo con Proyectos.

RStudio permite trabajar con proyectos. Un proyecto contendrá las siguientes características:

- Una carpeta donde se almacenan los ficheros de una forma estructurada
- El directorio de trabajo se establece en la carpeta principal del proyecto
- Una sesión de trabajo (incluye las variables en memoria y el historial de comandos de dicha sesión)
- Un fichero .Rprofile en la carpeta principal del proyecto que contendrá el perfil específico para ese proyecto
- Un fichero .RData con los datos y variable en memoria de la sesión de trabajo
- Un fichero .Rhistory con el historial de comandos ejecutados en el ámbito del proyecto
- Se guarda la configuración de los paneles, las pestañas y se abrirán los scripts que se hayan estado editando en la última sesión del proyecto

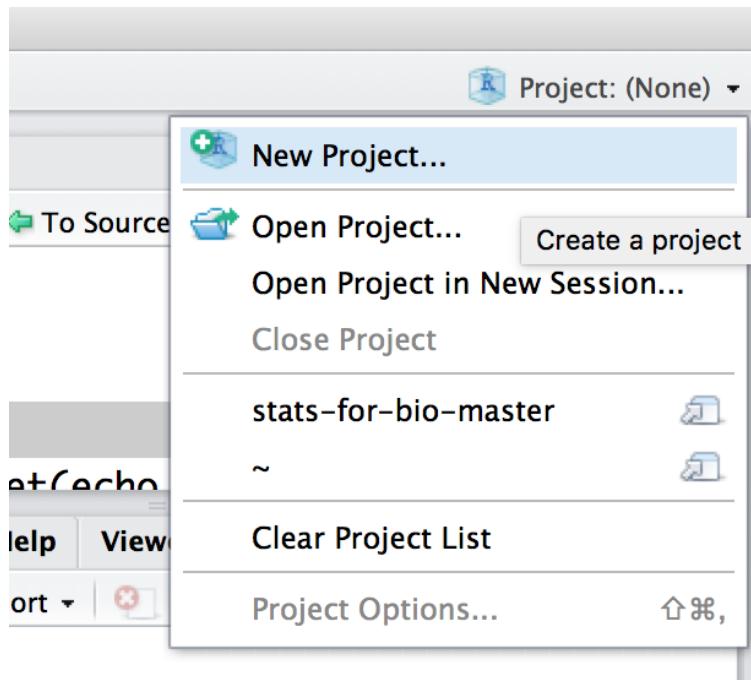


Figure 13: Proyectos en RStudio