

Introducción a R

M. A. Ibáñez

14/09/2023

Índice

1. ¿Qué es R?	1
2. Un poco de historia	1
3. Algunas características de R	1
4. ¿Cómo Instalar R?	2
5. La consola de R	2
5.1. Área de trabajo y carpeta de trabajo:	3
6. Funciones y Paquetes	4
6.1. Cargar paquetes en memoria. (<code>library()</code>)	4
6.2. Instalar paquetes	5
7. Formas de trabajar en R	5

1. ¿Qué es R?

R es un entorno de programación **enfocado al análisis estadístico**.

R proporciona una amplia variedad de técnicas estadísticas (modelos lineales y no lineales, análisis de series temporales, clasificación, . . .), multitud de posibilidades gráficas y además, al ser un lenguaje de programación, es altamente extensible permitiendo que los usuarios construyan sus propias funciones.

R tiene una pagina oficial en <http://www.R-project.org>. Este es el primer lugar al que debes ir para todo lo que esté relacionado con R.

2. Un poco de historia

El lenguaje de programación R lo comenzaron a crear Ross Ihaka y Robert Gentleman en el Departamento de Estadística de la universidad de Auckland, Nueva Zelanda en el año 1991. Es un dialecto de S que fue diseñado en la década de los 80 del siglo pasado en los laboratorios de AT&T Bell por Rick Becker, John Chambers y Allan Wilks.

Uno de las limitaciones de S es que solo podía usarse bajo el paquete comercial S-PLUS y por eso Ross y Robert crearon R. El nombre deriva de las iniciales de sus creadores.

En 1997, R forma parte del proyecto [GNU General Public License](#) de la Free Software Foundation. A partir de este momento, el código fuente del sistema R fue asequible para todo aquel que quisiera trabajar con el.

Dada la acogida que tiene R, se crean dos listas de correos para dar soporte a los usuarios: R-help y R-dev. La primera es una lista general para preguntas y la segunda más específicas para gente que participa en el desarrollo de R.

Además, se crea el grupo núcleo de R ([R Deveploment Core Team](#)), con la finalidad de administrar y controlar el código fuente de R. Cualquier modificación en el código fuente de R debe ser aprobada por este grupo.

3. Algunas características de R

R funciona en cualquier plataforma o sistema operativo: Mac, Windows, Linux, etc.

Hay lanzamientos de versiones frecuentes. Anualmente hay una o dos lanzamiento de versiones principales, y en ocasiones hay parches para solucionar algún problema detectado.

El sistema base de R es muy ligero. Su funcionalidad se divide en paquetes modulares, de forma que no es necesario descargar e instalar un software de gran tamaño.

El sistema base contiene:

- El paquete base que tiene todas las funciones fundamentales de bajo nivel que se necesitan para ejecutar R.
- Otros paquetes con utilidades básicas, tales como **stats** (funciones estadísticas), **datasets** (juego de datos), **graphics** (funciones para gráficos) y otros paquetes fundamentales que la mayoría de gente utiliza.
- Paquetes recomendados. No son críticos pero son usados frecuentemente.

R se puede ampliar fácilmente utilizando otros paquetes que se pueden obtener a través de la pagina oficial del proyecto R y cubren un amplio rango de modernas técnicas estadísticas.

El [CRAN](#)(Red integrada de archivos de R, en inglés Comprehensive R Archive Network) es una red de servidores ftp y web de todo el mundo que almacenan versiones idénticas y actualizadas de código y documentación para R.

En este momento, existen en el CRAN 18603 paquetes que han sido desarrollados por usuarios y programadores de todo el mundo ([packages](#)).

Estos paquetes son contribuciones de los usuarios. No son controlados por el 'R core'. Ahora bien, para poder estar en el CRAN tienen que tener cierto nivel de calidad. Así, tienen que tener documentación para las funciones que vienen en el paquete, y asegurarse de que pasan un cierto número de pruebas.

4. ¿Cómo Instalar R?

Para obtener el archivo de instalación hay que seleccionar en el [CRAN](#) un servidor desde donde bajar el ejecutable. Se recomienda utilizar uno próximo a nuestra localización para no saturar la red. Seleccionando para que sistema deseamos instalar R, accedemos al programa de instalación que debemos bajar a nuestro ordenador. Una vez guardado, se ejecuta y se siguen las instrucciones durante la instalación.

Para cualquier duda durante la instalación es posible consultar [R-FAQ](#). En concreto para Windows [R](#)

for [Windows FAQ](#) y para mac [R for MAC OS X FAQ](#).

5. La consola de R

Una vez instalado R aparecerá en el escritorio un acceso directo, si se ha seleccionado dicha opción al hacer la instalación. En el caso de que no aparezca, podemos ir a Inicio de Windows donde encontraremos el ejecutable para comenzar una sesión en R.

De esta forma se ejecutará la aplicación `Rgui.exe` que inicia una consola de windows (las siglas gui corresponden a Graphics User Interface) que será el entorno donde trabajaremos con R. En la figura 1 se muestra la consola.

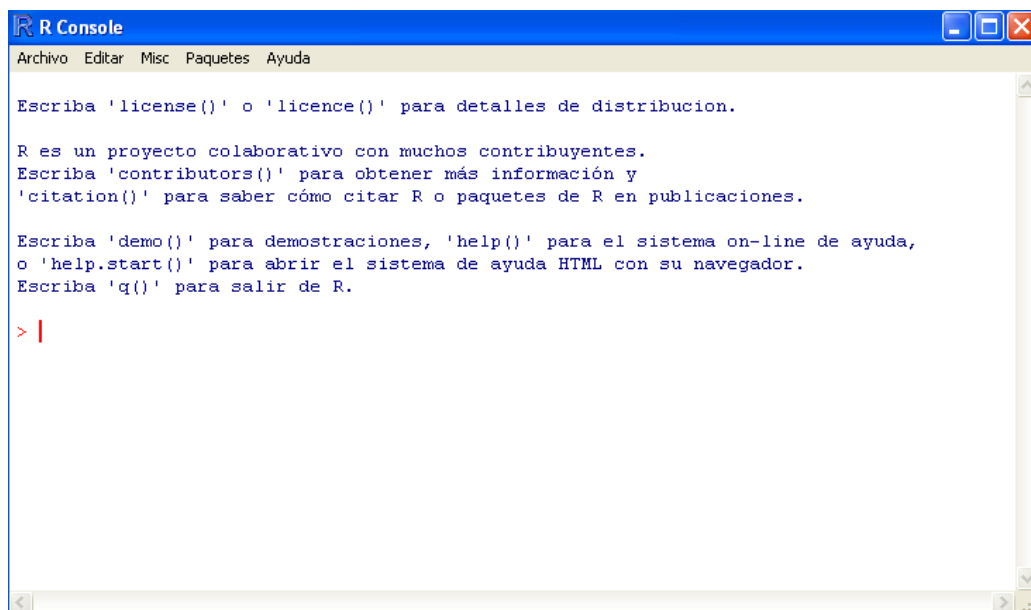


Figura 1: Consola en windows de R

La consola está diseñada para trabajar de forma interactiva, es decir mediante instrucciones o comandos que son tecleadas en la línea donde aparece el símbolo `>` y son ejecutadas al pulsar la tecla **Intro** o **Enter**. Al ejecutarse la instrucción se muestra en pantalla el resultado, en el caso de que exista, y vuelve a aparecer la línea con el símbolo `>`, indicando que es posible introducir una nueva instrucción.

Por ejemplo, tecleamos la siguiente instrucción.

```
> 3+5  
[1] 8
```

5.1. Área de trabajo y carpeta de trabajo:

Todas las cosas que creamos en una sesión de R se denominan objetos y están almacenados temporalmente en el **área de trabajo**.

El área de trabajo es parte de la memoria RAM que la aplicación ha reservado para almacenar los objetos que creamos durante una sesión de R (cuando lanzamos la aplicación).

Los objetos son nombres que almacenan distinto tipo de información (datos numéricos, caracteres,

matrices, resultados de un análisis estadístico, etc).

Para terminar la sesión en R ejecutamos la función `q()`, o utilizamos el menú `|Archivo/Salir|` o pinchamos con el ratón en el aspa que aparece en la esquina superior derecha de la consola.

La aplicación nos preguntará si deseamos guardar una imagen del área de trabajo (crear un archivo) con los objetos creados.

Si seleccionamos **no**, perderemos todos los objetos almacenado en el área de trabajo.

En el caso de seleccionar la opción **si**, lo guardará en un **directorio o carpeta de trabajo** con el nombre **.RData**^a.

Por defecto R, y salvo que se le diga lo contrario, asume que el directorio o carpeta de trabajo es la carpeta del disco duro en la que está instalado. Debemos cambiar la carpeta de trabajo utilizando el menú de la consola `|Archivo/Cambiar dir...|` e indicar que carpeta queremos utilizar. La carpeta de trabajo es donde R va a buscar o guardar cualquier fichero que necesitemos.

Es recomendable que cada tipo de trabajo que vamos a realizar con R este en una carpeta distinta del ordenador.

En dicha carpeta estarán los archivos de datos y los archivos con las instrucciones de R (scripts) que vamos a utilizar.

De tal forma, que al iniciar la sesión con R (lanzar la aplicación) le indiquemos que dicha carpeta va a ser la carpeta de trabajo.

6. Funciones y Paquetes

La mayoría del código que utilizamos en R son funciones.

Las funciones son conjuntos de instrucciones destinadas a realizar una determinada tarea computacional (generalmente un análisis estadístico).

Normalmente la función tiene unos argumentos de entrada y al ejecutarse se obtiene una salida que normalmente es una lista (más adelante veremos estudiará que es una lista)

Las funciones que manejamos en R vienen empaquetadas en **paquetes** (packages, en inglés).

En R, un paquete, es un conjunto de funciones diseñadas con un objetivo determinado.

Cuando instalamos R en nuestro ordenador, automáticamente se instalan un conjunto de paquetes. Los paquetes se instalan en la carpeta **library** que se encuentra dentro de la carpeta donde se ha instalado R. Dentro de **library**, cada paquete instalado tiene su propia carpeta.

Cuando iniciamos una sesión en R (lanzamos la aplicación R y se abre la consola de R), sólo se cargan en memoria un determinado número de paquetes. Es decir, sólo podemos utilizar las funciones de los paquetes que están cargados en memoria.

Así, por ejemplo, la función `mean()` es una función del paquete **stats** que se carga en memoria cuando iniciamos una sesión en R.

Otro de los paquetes que se cargan, es el paquete **base()** que contiene las funciones básicas de R para manejar objetos. Funciones como `c()`, `list()`, `data.frame()`, `colnames()` que utilizaremos en

^aLo normal es no guardar las áreas de trabajo. Ya veremos más adelante que la forma de trabajar es crear archivos de instrucciones (script) que cargamos y ejecutamos para obtener los resultados de nuestro análisis.

los temas posteriores son de este paquete.

6.1. Cargar paquetes en memoria. (`library()`)

En realidad, en memoria se cargan muy poquitos paquetes. La siguiente lista muestra los paquetes que se cargan al iniciar una sesión en R.

```
[1] "package:stats"      "package:graphics"  "package:grDevices"
[4] "package:utils"      "package:datasets"  "package:methods"
[7] "Autoloads"          "package:base"
```

Son los paquetes básicos para manipular objetos, hacer gráficos, importar y exportar datos y realizar análisis estadísticos.

El resto de paquetes instalados no se cargan en memoria y por tanto sus funciones no se pueden utilizar. Para poder utilizar las funciones de un paquete determinado que tenemos instalado en el ordenador, debemos cargarlo en memoria. Esto se hace con la función `library()`. El argumento de la función es el nombre del paquete que queremos cargar entre comillas.

Por ejemplo, entre los paquetes instalados está el paquete **Matrix** que es un paquete con funciones para el manejo de matrices que amplía las funciones que existen en el paquete **base** para manejar matrices.

Para cargar dicho paquete tecleamos la siguiente instrucción:

```
> library("Matrix")
```

De esta forma podemos, por ejemplo, utilizar la función `nnzero()` que nos devuelve el número de elementos distinto de 0 de un vector o una matriz.

```
> nnzero(c(0,2,3,4,5,0,0)) ## número de elementos distinto de cero en el vector
[1] 4
```

Una forma alternativa, para cargar paquetes, es utilizar el menú de la consola. En el menú vamos a **|paquetes/cargar paquetes|** y en la ventana que se despliega seleccionamos el paquete que deseamos cargar.

Cada vez que iniciamos una sesión en R y queremos utilizar las funciones de un paquete que no se carga por defecto, debemos cargarlo utilizando la función `library()` o usando el menú de la consola.

6.2. Instalar paquetes

Pero además de los paquetes que se instalan con R, en el CRAN de R existen multitud de paquetes que puede interesarnos utilizar. En este caso lo primero que hay que hacer es instalar el paquete en nuestro ordenador. Para ello podemos utilizar el menú de la consola de R e ir a **|paquetes/instalar paquetes|**. Nos abrirá un ventana para que seleccionemos el servidor desde donde queremos instalar el paquete. Normalmente se selecciona el más cercano a nuestra ubicación. En nuestro caso seleccionamos **Spain (Madrid)**. A continuación se despliega una ventana con todos los paquetes que hay en el CRAN, ordenados por orden alfabético. Nos desplazamos hasta encontrar el paquete que deseamos instalar.

Puede ocurrir que el paquete que queremos instalar utilice funciones de otros paquetes. En este caso R comprueba la dependencia e instala todos los paquetes de los que depende el paquete que queremos instalar. Por ahora, nosotros no necesitamos instalar ningún paquete por lo que no seleccionamos ningún paquete.

Por defecto (aunque esto se puede cambiar) el paquete se instala dentro de la carpeta `library` creando una nueva carpeta con el nombre del paquete instalado.

Un paquete solo es necesario instalarlo una vez.

Una vez instalado, para utilizar sus funciones es necesario cargarlo en memoria, tal y como hemos visto en el apartado anterior.

Recordar que en <https://cran.r-project.org/web/packages/> se encuentra la lista de los paquetes que hay en el CRAN y una descripción breve de cada uno de ellos.

7. Formas de trabajar en R

Cuando comenzamos una sesión en R es importante indicarle cual es nuestra carpeta de trabajo. Como hemos indicado anteriormente, la carpeta de trabajo es la carpeta de nuestro ordenador donde tenemos almacenados los archivos (script, archivos de datos, etc) que vamos a utilizar. Normalmente se utiliza una carpeta de trabajo distinta para cada uno de nuestros proyectos o análisis.

En R podemos trabajar de distintas formas.

Una posibilidad es trabajar directamente en la consola e ir ejecutando instrucciones.

Esta forma no es muy aconsejable ya que si en el futuro queremos reproducir el análisis realizado, tendríamos que volver a teclear todas las instrucciones.

Otra posibilidad es trabajar con los script. Los script son archivos en formato plano formado por líneas con instrucciones de R.

Normalmente, las instrucciones de un script tienen un orden de ejecución, comenzando en la línea una, luego la dos, etc. Si no se sigue ese orden, es posible que obtengamos un error de ejecución, al intentar utilizar un objeto que no ha sido creado previamente.

Para crear un nuevo script debemos ir al menú `|Archivo/Nuevo script|`. Se abre una ventana nueva donde podemos ir tecleando las instrucciones.

Para ejecutar una instrucción desde un script, nos colocamos en la línea de la instrucción y pulsamos conjuntamente las teclas `Ctrl+R`.

Al hacer esto, podemos observar que la instrucción es enviada a la consola de R y se ejecuta obteniendo el resultado correspondiente.

También es posible seleccionar con el ratón un conjunto de instrucciones y ejecutarlas con las teclas `Ctrl+R`.

Cuando hemos terminado el análisis, guardamos el script utilizando el menú `|Archivo/Guardar Como|` que se encuentra en la ventana del script.

Nos dará la posibilidad de ponerle un nombre. Podemos ponerle cualquier nombre pero es conveniente acabar el nombre con la extensión `.r`

El archivo se guarda en la carpeta de trabajo que le hayamos indicado previamente a R.

Podemos recuperar un script antiguo utilizando el menú `|Archivo/Abrir script|` de la consola.

Existen distintos entornos integrados de desarrollo, en inglés *Integrated Development Environment* (IDE) en R. Son aplicaciones que integran la consola de R, el editor de script y distintas utilidades para facilitar el trabajo en R.

Entre ellas, la más popular es [Rstudio](#).

En el siguiente apartado veremos que es Rstudio, como se instala y como se utiliza.