

# Introducción al software r

## Software Estadístico

Dra. Eva Romero Ramos

Dpto. Estadística e Investigación Operativa

# ¿Qué es r?

- El software r es un entorno y un lenguaje de programación enfocado a la computación matemática y estadística.
- Fue desarrollado por Robert Gentleman y Ross Ihaka en la Universidad de Auckland en 1993.
- Es un proyecto de software libre y se distribuye bajo la licencia GNU GPL.
- La licencia GNU GPL permite el uso, estudio y modificación del software, dando también la posibilidad de compartir desarrollos con el resto de usuarios.

# ¿Cómo instalar r?

- 1 Cómo se trata de un software gratuito podemos descargarlo desde su página web:

<https://www.r-project.org>

- 2 En esta web podemos encontrar las últimas versiones del software para Windows, MacOS y Ubuntu entre otros sistemas operativos.

- Algunas herramientas de r están directamente integradas en el software base, otras se gestionan mediante **paquetes**.
- Los paquetes se pueden descargar e instalar directamente a través el software.
- Los usuarios también pueden desarrollar paquetes y ponerlos a disposici'ón del resto de usuarios para su descarga y uso.

# ¿Qué es Rstudio?

- Rstudio es un entorno de desarrollo integrado (IDE) para R, que incluye:
  - Un terminal o consola.
  - Un editor de código con ayudas al desarrollo.
  - Herramientas de visualización, historial, depuración y gestión de los espacios de trabajo entre otras funcionalidades.
- Rstudio está disponible en código abierto y en versiones comerciales.

# ¿Cómo instalar Rstudio?

- 1 Podemos descargar Rstudio desde su página web:  
<https://rstudio.com>
- 2 En esta web podemos encontrar las últimas versiones del software para Windows, MacOS y Ubuntu entre otros sistemas operativos.
- 3 Es importante tener en cuenta que para instalar Rstudio debemos tener instalado r, ya que Rstudio solo es un IDE para R.

# Interfaz de Rstudio

1. Editor de Scripts

```
1 summary(GastoMensual)
2 #Calculo de la media
3 mean(GastoMensual$`Gasto mensual e
4 #Calculo de la media geométrica
5 exp(mean(log(GastoMensual$`Gasto n
6 #Calculo de la media armonica
7
```

2. Consola

o 'help.start()' para abrir el sistema de ayuda HTML con su navegador. Escriba 'q()' para salir de R.

3. Entorno e Historial

4. Archivos, gráficos, paquetes

- **Editor de Scripts:** En esta ventana se **escriben y gestionan** los Scripts.
- **Consola:** En esta ventana **ejecutan** los Scripts y aparecen los resultados de las operaciones realizadas. Aquí **no** se guardan los comandos, por lo que se deben escribir y guardar en el editor de Scripts.
- **Entorno e historial:** El entorno incluye un listado con todas las variables creadas en la sesión y sus valores. El historial de todas las operaciones realizadas en la sesión. Este historial guarda únicamente las operaciones y no incluye sus resultados.
- **Archivos, gráficos, paquetes . . .:** En esta zona podemos consultar el directorio de archivos en el que estamos trabajando, los gráficos que se han ido generando, los paquetes instalados y la ayuda.



Al comenzar un nuevo proyecto en R, lo primero que debemos hacer es definir un directorio donde se almacenarán los diferentes ficheros de trabajo.

Para definir un nuevo directorio de trabajo debemos acceder desde el menú superior a:

**Session** → **Set Working Directory** → **Chose Directory**

Una vez seleccionada la ruta donde queremos trabajar, en la pestaña **Files** de la ventana inferior derecha de la interfaz, se mostrarán todos los archivos que se encuentran en el directorio de trabajo.

El directorio de trabajo también se puede definir directamente mediante el comando:

```
setwd("C:/Documentos/Carpeta Introduccion software")
```

Esta opción puede resultar más interesante cuando trabajamos con diferentes proyectos, ya que, añadiendo esta línea de código al principio de cada script fijaremos de forma automática el entorno de trabajo.

Cuando abrimos un script ya creado, RStudio determinará que el directorio de trabajo es el directorio en que se encuentra dicho fichero.

# Instalación y carga de paquetes

- Siempre que iniciemos una nueva sesión, R cargará únicamente la funcionalidades del paquete básicos.
- A medida que se realizan trabajos y proyectos más complejos, se hace necesario usar funciones que no vienen instaladas y cargadas por defecto.
- Podemos instalar paquetes desde el menú superior a través de las opciones:

**Tools → Install Packages**

- También podemos instalar los paquetes por línea de comandos mediante:

```
install.packages()
```

# Limpiar memoria y consola

- Si queremos eliminar las variables declaradas en la ejecución de los diferentes scripts dentro de una misma sesión podemos usar el código:

```
rm(list=ls(all=TRUE))
```

Es una buena práctica usar este código al principio de cada script para evitar que se ejecute usando variables declaradas en ejecuciones previas, pero no en el script.

- Si lo que queremos limpiar es la consola para ver con claridad las operaciones que nos interesen en cada caso podemos usar la ruta del menú superior:

**Edit → Clear Console**