

Series Temporales. Práctica 1

Departamento de Estadística e Investigación Operativa.

Grado en Estadística. Universidad de Sevilla

Uso de datos y gráficas

1. Obtención de datos en la web	2
2. Obtención de datos de la web del INE	3
3. Lectura y grabado de datos con R	4
4. Gráficos sencillos para series temporales	6

Uso de datos y gráficas

1 Obtención de datos en la web

El **primer paso** de todo análisis de series temporales es la obtención de los datos. Si este paso no se puede hacer o se hace mal, todo lo que se haga posteriormente no será útil.

Dado que actualmente la **web se configura como una fuente muy importante de datos publicados** (bien oficiales o puestos a disposición de los interesados por terceros), es necesario conocer las pautas habituales para acceder a tales datos.

Existen numerosas páginas web donde se pueden obtener datos sobre temas específicos. Muchos de estos datos son susceptibles de ser analizados como **series temporales**.

En la mayoría de estas páginas web se pueden descargar los datos en **formatos diversos**. Aunque en algunas web existen formatos propios para ser tratados también con software propio, es habitual disponer de la opción de descargar los datos en formato **csv** (datos separados por comas o por puntos y comas), en **excel** o en **formato plano**.

Algunas direcciones web donde podemos encontrar datos de este tipo son las siguientes:

- <https://ec.europa.eu/eurostat> Este es el portal de EUROSTAT.
- <https://www.juntadeandalucia.es/institutodeestadisticaycartografia> Esta es la página del IECA.
- <http://www.ine.es/> Esta es la página del INE.
- <http://serviciosedemineco.gob.es/Indeco/BDSICE/HomeBDSICE.aspx> Acceso a las bases de datos de series de Indicadores de Coyuntura Económica (BDSICE).
- <http://www.bde.es/bde/es/areas/estadis/> Banco de España.

2 Obtención de datos de la web del INE

Dado que la página web del I.N.E proporciona gran cantidad de datos, veremos un ejemplo de descarga de series de datos.

Ejemplo: Obtener los datos mensuales del I.P.C. general desde enero de 2002 hasta el último dato.

1. Ir a la dirección de la [web del INE \(www.ine.es\)](http://www.ine.es).
2. En la columna izquierda de la página, en [INEbase](#), seleccionar el grupo donde están los datos del I.P.C. En este caso, donde pone [Nivel y condiciones de vida \(IPC\)](#), subapartado [Índices de precios de consumo y vivienda](#).
3. En la nueva página, seleccionar [Índice de precios de consumo](#).
4. En la nueva página, pulsar donde pone [Resultados](#), seleccionar la pestaña [Mensuales](#), luego [Resultados nacionales](#) y luego [Índices nacionales](#).
5. En la nueva página, dentro de [Índices nacionales: general y de grupos ECOICOP](#) seleccionar [índice general](#), dentro de [Tipo de dato](#) seleccionar [Índice](#), en [Periodos](#) pulsar el icono con rayas azules para seleccionar todos los datos y el icono con la A y la Z para que los datos se presenten en el orden temporal desde 2002 hasta ahora.
6. Desplegar las opciones de presentación de la tabla pulsando en el signo -"junto a " [Elija forma de presentación de la tabla](#)" y cambiar las columnas por las filas.
7. Pulsar en [Consultar selección](#).
8. Aparecen los datos en una nueva pantalla. En el lado derecho pulsar en el icono de descarga (con la flecha hacia abajo) y seleccionar el formato deseado (por ejemplo excel xls).
9. También se pueden representar en la pestaña [Gráfica](#), al lado de la de [Tabla](#).
10. Ya disponemos de los datos deseados. Podemos guardarlos en el directorio que se quiera.

Nota: En esta página web se puede usar el buscador para encontrar resultados con más rapidez.

Ejercicio 1.

1. Obtener un fichero con los datos en formato de datos separados por puntos y comas sobre el índice de producción industrial entre 2002 y 2015.
2. Obtener un fichero con los datos en formato de datos separados por puntos y comas sobre los datos trimestrales de Paro entre 2003 y 2012. Guardarlo con el nombre `paro.csv` en el directorio raíz de C:

3 Lectura y grabado de datos con R

R es un entorno muy útil para el análisis estadístico. En esta y sucesivas prácticas estudiaremos el **uso de R para analizar series temporales**.

Existe mucha **documentación relativa al uso del R en general**. Puede consultarse las direcciones:

- <https://cran.r-project.org/other-docs.html>
- <https://cran.r-project.org/manuals.html>

en las que hay documentación en varios idiomas, entre ellos en español, sobre R y sus aplicaciones en el análisis estadístico.

Sobre el uso de **R para el análisis de series temporales** también hay material disponible en internet y en libros. Citamos **dos libros**:

- Andrew V. Metcalfe, Paul S.P. Cowpertwait, **Introductory Time Series with R**, New York, NY : Springer-Verlag New York, 2009.
- Robert H. Shumway, David S. Stoffer, **Time Series Analysis and Its Applications: With R Examples**. New York, NY. Springer Science+Business Media, LLC, 2011.

Para iniciar el análisis de una serie temporal supondremos que tenemos los datos en **formato electrónico**.

Para un tratamiento extenso sobre la **importación y exportación de datos**, puede consultarse el siguiente manuales de la página oficial del **proyecto R**:

- <http://cran.r-project.org/doc/manuals/r-release/R-data.pdf>

Existen varias funciones para la lectura de ficheros de datos:

- `read.table()` y variantes como `read.csv()`
- otros: `read.fwf()`, `scan()`, etc.

Instrucción `read.table()` y `read.csv()`

Para leer un fichero simple, con los datos separados por espacios en blanco, tabuladores o saltos de línea, se utiliza la instrucción `read.table` en la forma:

```
fichero.df <- read.table("c:/dir/mi.fichero",  
+ header = TRUE, sep = " ", comment.char = " ")
```

Si el carácter decimal no es un punto sino, por ej., una coma, usar: `dec = ","`.

Se pueden saltar líneas (`skip`) o leer un número fijo de líneas (`nrows`).

Hay funciones especializadas para otros archivos (por ejemplo `read.csv`), pero son casos específicos de `read.table`.

```
fichero <- read.csv("C:/datosipc.csv", header=T, dec=".", sep=";")
```

Ejercicio 2.

Leer los datos del fichero `paro.csv` puesto en la documentación de la práctica:

La instrucción a usar es:

```
paro <- read.csv("paro.csv", header=F, dec=".", sep=";")
```

Nota: La especificación del fichero debe llevar la ruta completa, por ejemplo [C:/paro.csv](#), si el fichero no está en el directorio de trabajo de R.

Si son pocos, podemos **teclear los datos directamente** en un vector o dataframe. Por ejemplo, así:

```
a <- c(1,2,3,4)
```

```
b <- matrix(a,2,2,byrow=FALSE)
```

Para **escribir a un fichero CSV**, importable desde Excel

```
write.table(datos, file = "datos.csv", sep = ";", col.names = NA)
```

Para **escribir una matriz $n \times r$ a un fichero**,

```
write(t(datos), file = "datos.txt", ncol=r)
```

4 Gráficos sencillos para series temporales

Existen diversas funciones en **R** para realizar gráficas. Podemos destacar las siguientes en relación al análisis de series temporales:

- **plot(x)**: Hace la gráfica de los valores de x (en el eje y), ordenados en el eje x.
- **plot(x, y)**: gráfico bivariado de x (en el eje x) e y (en el eje y).
- **plot.ts(x)**: si x es un objeto de clase **ts** (se verá a continuación), hace la gráfica de x con respecto al tiempo. x puede ser multivariada pero las series deben tener la misma frecuencia y fechas.
- **ts.plot(x)**: igual a la anterior pero si x es multivariado, las series pueden tener diferentes fechas pero la misma frecuencia.

Se pueden usar varios parámetros para ajustar el gráfico (ver *help(par)* y *help(plot)*.)

Algunas de estas opciones son idénticas para varias funciones gráficas. Listamos las principales (con sus valores por defecto):

- **type="p"**. Especifica el tipo de gráfico; "p": puntos, "l": líneas, "b": puntos conectados por líneas, ".": igual al anterior, pero las líneas están sobre los puntos, "h": líneas verticales, "s": escaleras, los datos se representan como la parte superior de las líneas verticales, "S": igual al anterior pero los datos se representan como la parte inferior de las líneas verticales.

- **xlim=, ylim=**. Especifica los límites inferiores y superiores de los ejes; por ejemplo con `xlim=c(1, 10)` o `xlim=range(x)`.
- **xlab=, ylab=**. Títulos en los ejes; deben ser variables de tipo carácter.
- **main=**. Título principal; debe ser de tipo carácter.
- **sub=**. Sub-título (escrito en una letra más pequeña).

Un ejemplo de gráfica con `plot` es la siguiente:

```
x<- c(-10:10)
y<- x^2
plot(x, y, main = "Titulo principal", sub = "Subtitulo",
     xlab= "Eje x", ylab = "Eje y",
     xlim = c(-10,10),ylim = c(-5,100),
     type="l", col="red", lwd="3")
```

Ejercicio 3.

Representar los datos de paro registrado entre 2003 y 2012.

```
plot(paro[,2],type="l")
```

```
plot(paro[,2], type="l", lwd=3, col=" red")
```