

# Notas de cómo hacer algunas cosas con R

Marcos Bujosa

July 27, 2024

Algunas notas de cómo hacer cosas en R.

## Carga de algunas librerías de R que vamos a usar aquí

---

```
library(readr)      # para leer ficheros CSV
library(zoo)         # para leer datos con índices temporales
library(ggplot2)
library(jtools)      # para representación resultados estimación (summ)
```

---

y además fijamos los parámetros por defecto para las figuras en png del notebook

---

```
# fijamos el tamaño de las figuras que se generan en el notebook
options(repr.plot.width = 12, repr.plot.height = 4, repr.plot.res = 200)
```

---

## 1 De CSV a data\_frame

---

```
datos_df <- read_csv('datos/GNPvsMelanoma.csv', show_col_types = FALSE)
head(datos_df, 3)
```

---

obs	GNP	Melanoma
<dbl>	<dbl>	<dbl>
1936	193.0	1.0
1937	203.2	0.8
1938	192.9	0.8

## 2 De data\_frame a ts

Este método permite volver al data\_frame

---

```
datos_ts <- ts(data = datos_df,
               start = 1936,
               end = 1972,
               frequency = 1)
head(datos_ts, 3)
```

---

obs	GNP	Melanoma
1936	193.0	1.0
1937	203.2	0.8
1938	192.9	0.8

## 3 De CSV a ts (vía zoo)

### 3.1 Primero zoo

---

```
data <- read.zoo('datos/GNPvsMelanoma.csv', header=TRUE, index.column = 1, sep=";", FUN = as.yearmon)
class(data)
head(data, 3)
plot(data)
plot(data[, 'Melanoma'])
```

---

'zoo'

	GNP	Melanoma
ene 1936	193.0	1.0
ene 1937	203.2	0.8
ene 1938	192.9	0.8



### 3.2 Luego a ts

---

```
data_ts = as.ts( data )
class(data_ts)
head(data_ts, 3)
plot(data_ts[, 'GNP'])
```

---

1. 'mts'
2. 'ts'

### 3. 'matrix'

GNP	Melanoma
193.0	1.0
203.2	0.8
192.9	0.8

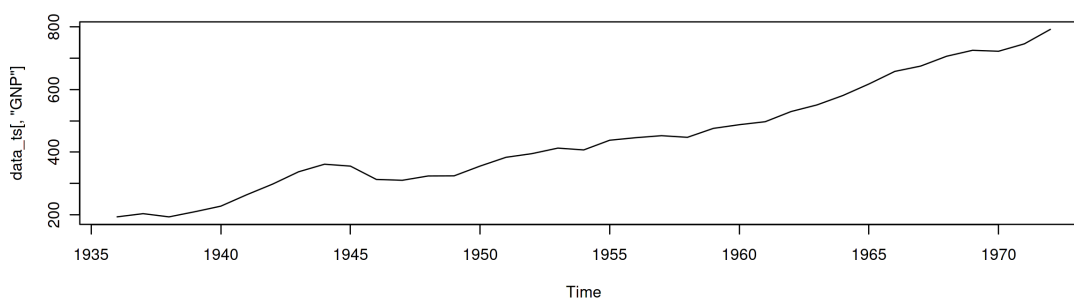


Figure 1: A matrix: 3 (E 2 of type dbl

## 4 De dataframe a ts

---

```

# +BEGIN_SRC jupyter-R
tseries <- as.ts( read.zoo( datos_df ) )
head(tseries)
plot(tseries)

```

---

GNP	Melanoma
193.0	1.0
203.2	0.8
192.9	0.8
209.4	1.4
227.2	1.2
263.7	1.0

## 5 De ts a data\_frame

---

```

DF = data.frame(date = zoo::as.Date(time(tseries[, "GNP"])),
                GNP = as.matrix(tseries[, "GNP"]),
                Melanoma = as.matrix(tseries[, "Melanoma"]))
head(DF, 3)

```

---

	date	GNP	Melanoma
	<date>	<dbl>	<dbl>
1	1936-01-01	193.0	1.0
2	1937-01-01	203.2	0.8
3	1938-01-01	192.9	0.8

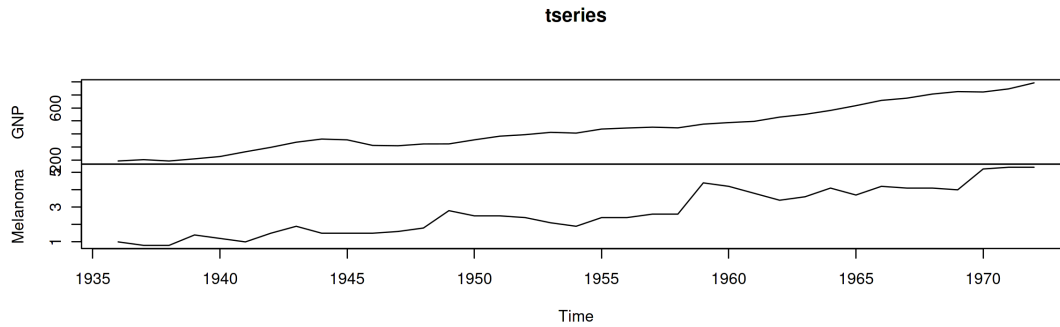


Figure 2: A matrix: 6 (E 2 of type dbl

## 6 Resumen: dos formas de pasar de datos anuales en CSV a ts

Una forma

---

```
time_series <- ts(data = read_csv('datos/GNPvsMelanoma.csv', show_col_types = FALSE),
  start = 1936,
  end = 1972,
  frequency = 1)
head(time_series, 2)
```

---

	obs	GNP	Melanoma
	1936	193.0	1.0
	1937	203.2	0.8

donde `read_csv('datos/GNPvsMelanoma.csv', show_col_types = FALSE)` es un `data_frame`  
 Otra forma (vía zoo)

---

```
# library(zoo)
data_ts = as.ts( read.zoo('datos/GNPvsMelanoma.csv', header=TRUE, index.column = 1, sep=",", FUN = as.yearmon) )
head(data_ts, 3)
```

---

	GNP	Melanoma
	193.0	1.0
	203.2	0.8
	192.9	0.8

## 7 Gráfico de dos series temporales con sendos ejes verticales

---

```
# Mostrando la serie GNP
p <- autoplot(as.zoo(data_ts[, 'GNP']))
p <- p + geom_line(aes(y = as.zoo(data_ts[, 'GNP'])), colour="blue")

# como tienen escalas distintas se requiere ajustar los datos
sf <- max(data_ts[, 'GNP'])/max(data_ts[, 'Melanoma'])

# Se agrega Melanoma a Y multiplicada por el factor
p <- p + geom_line(aes(y = as.zoo(data_ts[, 'Melanoma'])*sf), colour="red")
p <- p + scale_y_continuous(sec.axis = sec_axis(~./sf, name = "Incidencia casos de melanoma"))
```

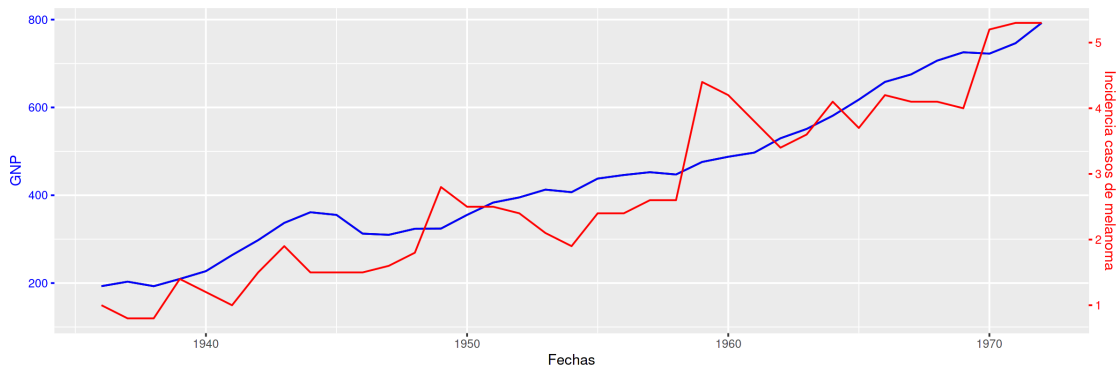
```

p <- p + labs(y = "GNP",
              x = "Fechas")

# Se modifican los colores de los ejes
p <- p + theme(
  axis.title.y.left=element_text(color="blue"),
  axis.text.y.left=element_text(color="blue"),
  axis.ticks.y.left = element_line(color = "blue"),
  axis.title.y.right=element_text(color="red"),
  axis.text.y.right=element_text(color="red"),
  axis.ticks.y.right = element_line(color = "red")
)
p

```

---



## 7.1 Desde el dataframe

```

# Mostrando la serie GNP
p <- ggplot(DF, aes(x = date))
p <- p + geom_line(aes(y = GNP), colour="blue")

# como tienen escalas distintas se requiere ajustar los datos
sf<-max(DF['GNP'])/max(DF['Melanoma'])

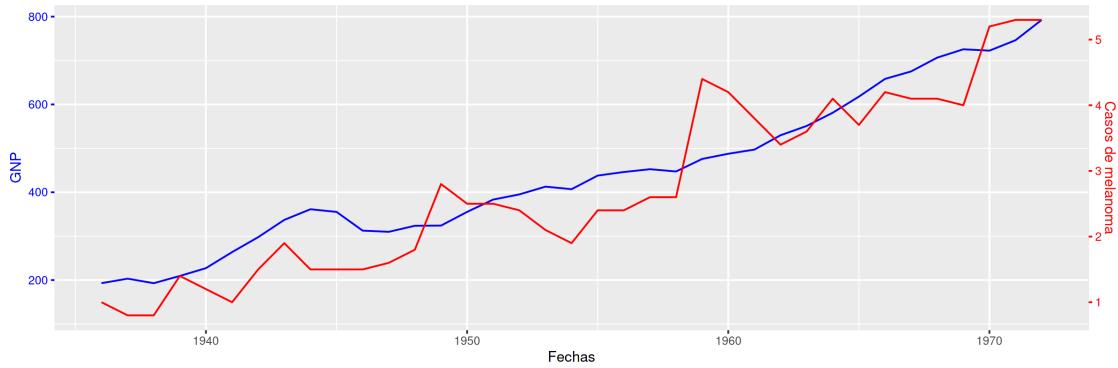
# Se agrega Melanoma a Y multiplicada por el factor
p <- p + geom_line(aes(y = Melanoma*sf), colour="red")

p <- p + scale_y_continuous(sec.axis = sec_axis(~./sf, name = "Casos de melanoma"))
p <- p + labs(y = "GNP",
              x = "Fechas")

# Se modifican los colores de los ejes
p <- p + theme(
  axis.title.y.left=element_text(color="blue"),
  axis.text.y.left=element_text(color="blue"),
  axis.ticks.y.left = element_line(color = "blue"),
  axis.title.y.right=element_text(color="red"),
  axis.text.y.right=element_text(color="red"),
  axis.ticks.y.right = element_line(color = "red")
)
p

```

---



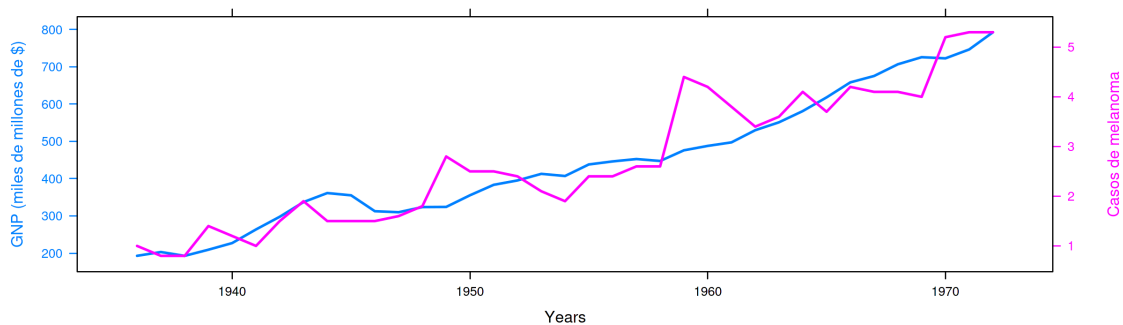
### 7.1.1 Y otra manera con xyplot (de latticeExtra)

Y ahora generamos el gráfico

---

```
library(latticeExtra) # alternativa para gráficos con doble eje vertical (doubleYScale)
kk <- xyplot(GNP + Melanoma ~ date, DF, type="l")
# se agrega dos ejes Y se construye cada serie por separado
obj1 <- xyplot(GNP ~ date, DF, type = "l", lwd=2, ylab="GNP (miles de millones de $)", xlab="Years")
obj2 <- xyplot(Melanoma ~ date, DF, type = "l", lwd=2, ylab="Casos de melanoma")
# --> se realiza la grafica con el segundo eje Y
doubleYScale(obj1, obj2, add.ylab2 = TRUE)
```

---



## 8 Añadir nueva columna a un ts

---

```
d_GNP      = diff(datos_ts[, "GNP"])
d_Melanoma = diff(datos_ts[, "Melanoma"])
```

---

5f061e65-c149-4dee-bc7d-a5338e092fb4

este método cambia el nombre de las primeras columnas

---

```
serie = ts.union(data_ts, d_GNP, d_Melanoma)
head(serie, 3)
```

---

76659920-41c9-4bd6-a654-09e26c8b2a20

Este método es mejor, pero es pesado

---

```
serie = ts.union(GNP = data_ts[, 'GNP'], Melanoma = data_ts[, 'Melanoma'], d_GNP, d_Melanoma)
head(serie, 3)
```

---

db11562d-7198-478f-953f-40fdfaf12282

---

```
DF.diferencias = data.frame(date = zoo::as.Date(time(d_GNP)),
                             d_GNP = as.matrix(d_GNP),
                             d_Melanoma = as.matrix(d_Melanoma))
head(DF.diferencias, 2)
```

---

cd42271e-0f3a-4cd9-a1c4-1cdf0b4cb9c9

## 9 Otros

---

```
mal_modelo <- lm(d_GNP ~ d_Melanoma)
summ( mal_modelo )
```

---

1ceafe33-19a8-41f8-885a-cec3987c19fa

---

```
plot(as.ts(resid(mal_modelo)))
abline(0,0)
```

---

9fb7d7f2-2948-46c7-8240-ca2e9384cc2d