

Ejercicios en clase

Carlos Santiago Martinez Torres

2025-05-24

Contents

| | |
|--|----------|
| Configuracion inicial | 1 |
| Carpetas | 1 |
| Ejercicio Tema 4. tidy iris. | 2 |
| Carga del conjunto | 2 |
| Numero de registros y tipos de variables | 2 |
| Variables perdidas. NA. | 2 |
| Transformaciones | 2 |
| Ejercicio Tema 4. Baby boom. | 3 |
| Lectura de fichero | 3 |
| Numero de registros y tipos de variables | 3 |
| Variables perdidas. NA. | 3 |
| Transformaciones | 3 |
| Representacion | 4 |

Configuracion inicial

Carpetas

```
# Creamos directorio data si no existe, para guardar los ficheros
if (!file.exists('data')) { dir.create('data') }

# Creamos directorio figure si no existe, para guardar las figuras
if (!file.exists('figure')) { dir.create('figure') }

# Creamos los accesos a cada carpeta
datos <- 'data/'
figuras <- 'figure/'
```

Ejercicio Tema 4. tidy iris.

Carga del conjunto

```
# Ya está en r. Lo trabajamos con otra variable  
dfiris <- iris
```

Numero de registros y tipos de variables

```
str(dfiris)  
  
## 'data.frame': 150 obs. of 5 variables:  
## $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...  
## $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...  
## $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...  
## $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...  
## $ Species : Factor w/ 3 levels "setosa","versicolor",...: 1 1 1 1 1 1 1 1 1 1 ...  
  
# glimpse(df)  
# dim(df)
```

Variables perdidas. NA.

```
sum(is.na(dfiris))
```

```
## [1] 0
```

Transformaciones

```
# Pasar a conjunto tidy  
# dfiris <- pivot_longer(data = dfiris, names_to = 'Parte', values_to = 'Valor', -Species)  
dfiris <- dfiris %>%  
  pivot_longer(data = ., names_to = 'Parte', values_to = 'Valor', -Species)  
  
# Separar las columnas y organizar  
# ?separate()  
dfiris <- dfiris %>%  
  separate(data = ., col = Parte, into = c('ParteFlor', 'TipoMedida'), sep = '[.]') %>%  
  select(ParteFlor, TipoMedida, Valor, Species)  
  
# Verificamos el conjunto  
head(dfiris)
```

```
## # A tibble: 6 x 4
##   ParteFlor TipoMedida Valor Species
##   <chr>      <chr>      <dbl> <fct>
## 1 Sepal     Length        5.1 setosa
## 2 Sepal     Width         3.5 setosa
## 3 Petal     Length        1.4 setosa
## 4 Petal     Width         0.2 setosa
## 5 Sepal     Length        4.9 setosa
## 6 Sepal     Width          3   setosa
```

Ejercicio Tema 4. Baby boom.

Lectura de fichero

```
f <- paste0(datos, 'children_per_woman_total_fertility.csv')
df <- read_csv(file = f, show_col_types = FALSE)
```

Numero de registros y tipos de variables

```
# str(df)
# glimpse(df)
dim(df)
```

```
## [1] 197 302
```

Variables perdidas. NA.

```
sum(is.na(df))
```

```
## [1] 0
```

Transformaciones

```
# Convertir a un conjunto tidy
df <- pivot_longer(data = df, names_to = 'Year', values_to = 'Value', cols = -country) # Hacerlo una vez
```

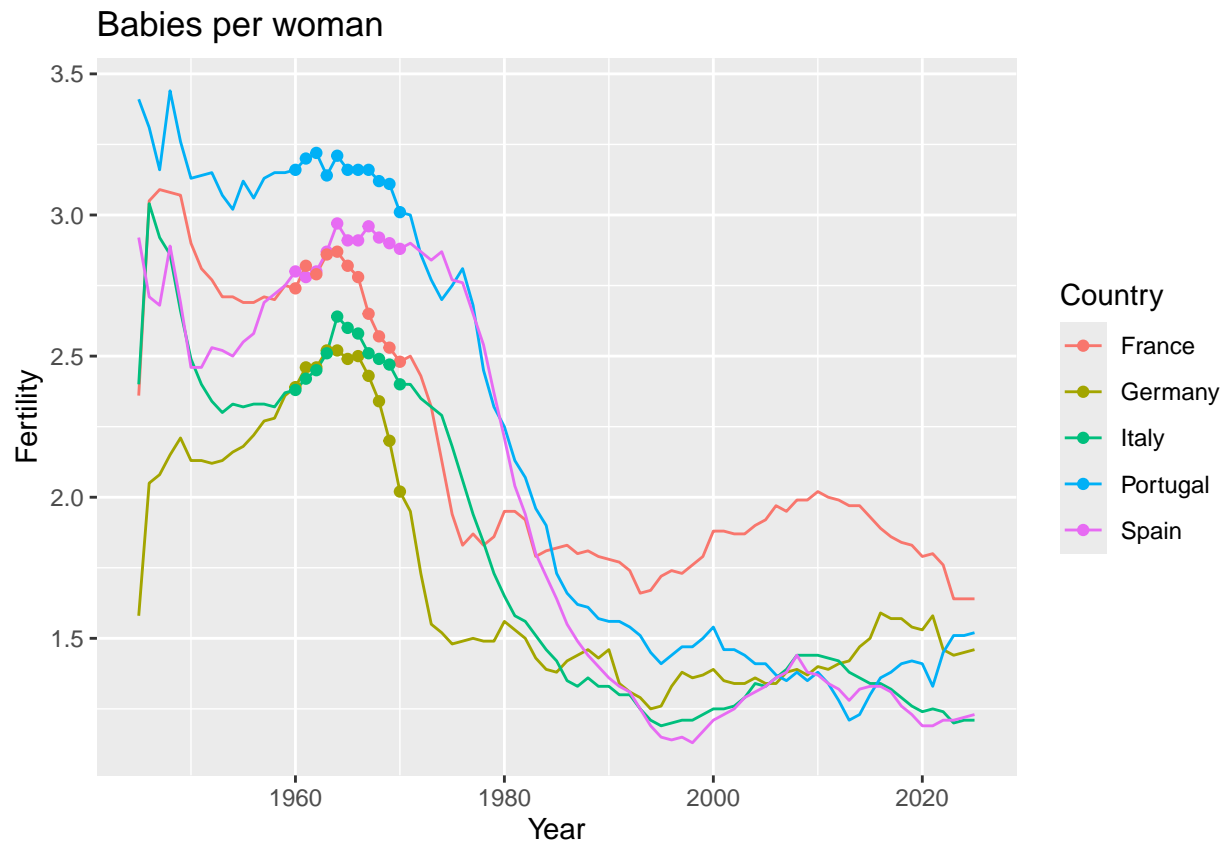
```
# Verificacion de valores
str(df) # El año no es necesario cambiarlo porque no operaremos aritmeticamente con el
```

```
## tibble [59,297 x 3] (S3: tbl_df/tbl/data.frame)
## $ country: chr [1:59297] "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
## $ Year : chr [1:59297] "1800" "1801" "1802" "1803" ...
## $ Value : num [1:59297] 7 7 7 7 7 7 7 7 7 ...
```

```
# Actualizamos para trabajar mejor los filtros
df <- df %>%
  mutate(Year = as.integer(Year)) %>% # Cambiamos el tipo de dato
  rename(Country = country) %>% # Cambiamos 'country' a 'Country'
  select(Country, Year, Value) # Seleccionamos en orden para organizar
```

Representacion

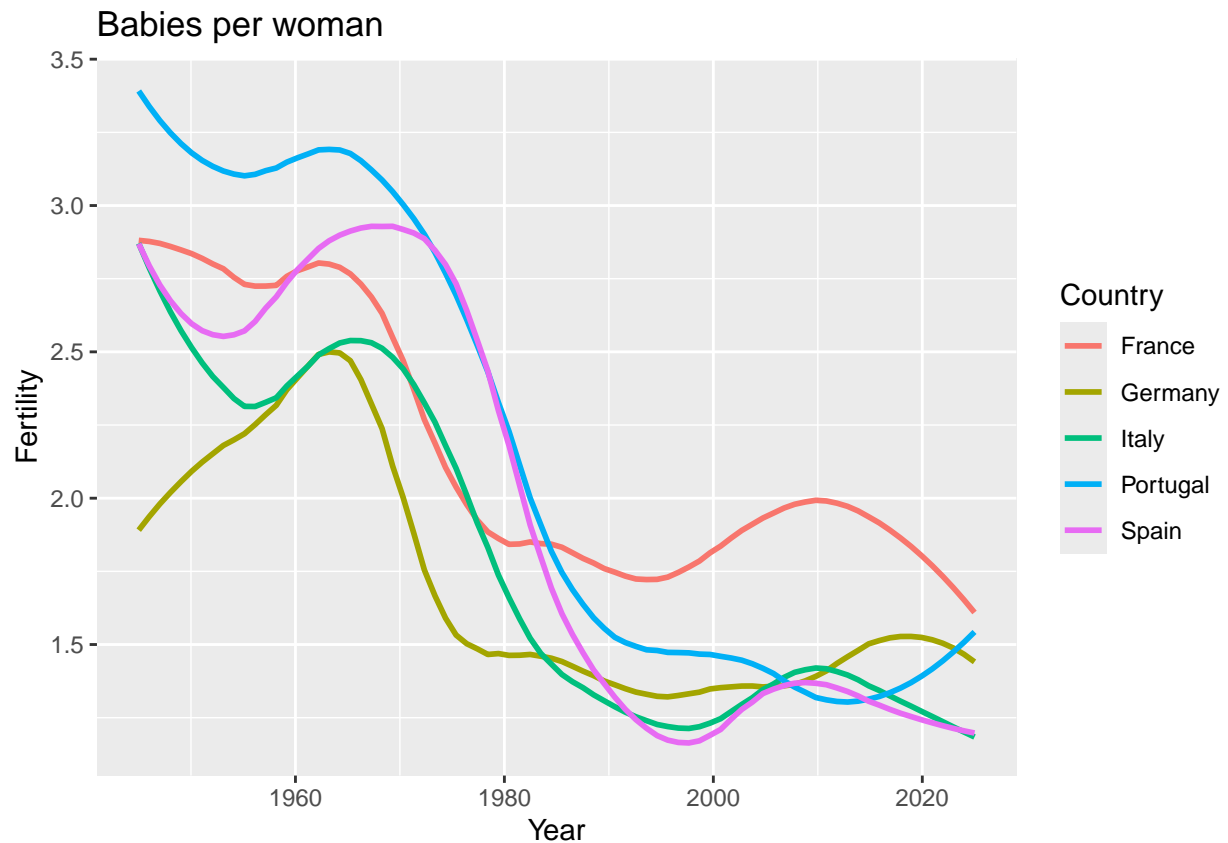
```
# Representacion 1
df %>%
  filter(Year >= 1945 & Year <= 2025) %>%
  filter(Country %in% c('Spain', 'France', 'Germany', 'Portugal', 'Italy')) %>%
  ggplot(data = ., mapping = aes(x = Year, y = Value, group = Country, color = Country)) +
  geom_line() +
  geom_point(data = ~filter(., Year >= 1960 & Year <= 1970)) + # Para añadir puntos de 1960 a 1980
  labs(title = 'Babies per woman',
       x = 'Year',
       y = 'Fertility')
```



```
# Representacion 2
df %>%
  filter(Year >= 1945 & Year <= 2025) %>%
  filter(Country %in% c('Spain', 'France', 'Germany', 'Portugal', 'Italy')) %>%
```

```
ggplot(data = ., mapping = aes(x = Year, y = Value, group = Country, color = Country)) +
  # geom_line() +
  # geom_point(data = ~filter(., Year >= 1960 & Year <= 1970)) + # Para añadir puntos de 1960 a 1980
  labs(title = 'Babies per woman',
        x = 'Year',
        y = 'Fertility') +
  geom_smooth(se = FALSE, span = 0.3) # Línea suavizada
```

'geom_smooth()' using method = 'loess' and formula = 'y ~ x'



```
# se, oculta la banda de error
# span, controla el suavizado. 0.3 mas suave, 1 mas rigido
```