

Laboratorio 27-MD

Leislíe R. Manjarrez O.

2023-02-23

El Laboratorio 27 se puede observar después de haber ejecutado los Laboratorios 25 y 26

Pre-requisitos: instalar paquetería `install.packages("tidyverse")`

Instalar paquete de datos `install.packages("remotes") remotes::install_github("cienciadedatos/datos") install.packages("datos")`

Llamar la librería de tidyverse

```
library("tidyverse")
```

```
## — Attaching core tidyverse packages — tidyverse 2.0.0 —
## ✓ dplyr      1.1.0      ✓ readr      2.1.4
## ✓ forcats    1.0.0      ✓ stringr    1.5.0
## ✓ ggplot2     3.4.1      ✓ tibble      3.1.8
## ✓ lubridate  1.9.2      ✓ tidyr       1.3.0
## ✓ purrr       1.0.1
## — Conflicts — tidyverse_conflicts() —
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()     masks stats::lag()
## i Use the `library()` function to force all conflicts to become errors
```

Llamar librería de datos

```
library("datos")
```

Visualizar las tablas a utilizar (tabla1 a tabla4b). Ver datos como tibble

```
datos::tabla1
```

```
## # A tibble: 6 × 4
##   pais      anio  casos  poblacion
##   <chr>    <dbl> <dbl>    <dbl>
## 1 Afganistán 1999    745   19987071
## 2 Afganistán 2000   2666  20595360
## 3 Brasil     1999  37737  172006362
## 4 Brasil     2000  80488  174504898
## 5 China      1999 212258 1272915272
## 6 China      2000 213766 1280428583
```

```
datos::tabla2
```

```
## # A tibble: 12 × 4
##   pais      anio tipo      cuenta
##   <chr>    <dbl> <chr>    <dbl>
## 1 Afganistán 1999 casos      745
## 2 Afganistán 1999 población 19987071
## 3 Afganistán 2000 casos      2666
## 4 Afganistán 2000 población 20595360
## 5 Brasil     1999 casos      37737
## 6 Brasil     1999 población 172006362
## 7 Brasil     2000 casos      80488
## 8 Brasil     2000 población 174504898
## 9 China      1999 casos      212258
## 10 China     1999 población 1272915272
## 11 China     2000 casos      213766
## 12 China     2000 población 1280428583
```

```
datos::tabla3
```

```
## # A tibble: 6 × 3
##   pais      anio tasa
##   <chr>    <dbl> <chr>
## 1 Afganistán 1999 745/19987071
## 2 Afganistán 2000 2666/20595360
## 3 Brasil     1999 37737/172006362
## 4 Brasil     2000 80488/174504898
## 5 China      1999 212258/1272915272
## 6 China      2000 213766/1280428583
```

```
datos::tabla4a
```

```
## # A tibble: 3 × 3
##   pais      `1999` `2000`
##   <chr>    <dbl> <dbl>
## 1 Afganistán    745   2666
## 2 Brasil       37737 80488
## 3 China        212258 213766
```

```
datos::tabla4b
```

```
## # A tibble: 3 × 3
##   pais      `1999`      `2000`
##   <chr>    <dbl>    <dbl>
## 1 Afganistán 19987071 20595360
## 2 Brasil    172006362 174504898
## 3 China     1272915272 1280428583
```

Ver datos como dataframe

```
df1 <- data_frame(tabla1)
```

```
## Warning: `data_frame()` was deprecated in tibble 1.1.0.
## i Please use `tibble()` instead.
```

```
df2 <- data_frame(tabla2)
df3 <- data_frame(tabla3)
df4a <- data_frame(tabla4a)
df4b <- data_frame(tabla4b)
```

Visualizar encabezados dataframe

```
head(df1)
```

```
## # A tibble: 6 × 4
##   pais      anio casos poblacion
##   <chr>    <dbl> <dbl>    <dbl>
## 1 Afganistán 1999    745  19987071
## 2 Afganistán 2000   2666  20595360
## 3 Brasil     1999 37737 172006362
## 4 Brasil     2000 80488 174504898
## 5 China      1999 212258 1272915272
## 6 China      2000 213766 1280428583
```

```
head(df2)
```

```
## # A tibble: 6 × 4
##   pais      anio tipo      cuenta
##   <chr>    <dbl> <chr>    <dbl>
## 1 Afganistán 1999 casos      745
## 2 Afganistán 1999 población 19987071
## 3 Afganistán 2000 casos      2666
## 4 Afganistán 2000 población 20595360
## 5 Brasil     1999 casos      37737
## 6 Brasil     1999 población 172006362
```

```
head(df3)
```

```
## # A tibble: 6 × 3
##   pais      anio tasa
##   <chr>    <dbl> <chr>
## 1 Afganistán 1999 745/19987071
## 2 Afganistán 2000 2666/20595360
## 3 Brasil     1999 37737/172006362
## 4 Brasil     2000 80488/174504898
## 5 China      1999 212258/1272915272
## 6 China      2000 213766/1280428583
```

```
head(df4a)
```

```
## # A tibble: 3 × 3
##   pais      `1999` `2000`
##   <chr>    <dbl> <dbl>
## 1 Afganistán    745    2666
## 2 Brasil       37737  80488
## 3 China        212258 213766
```

```
head(df4b)
```

```
## # A tibble: 3 × 3
##   pais      `1999`      `2000`
##   <chr>    <dbl>    <dbl>
## 1 Afganistán 19987071 20595360
## 2 Brasil    172006362 174504898
## 3 China     1272915272 1280428583
```

Exportar los dataframe originales

```
write.csv(df1, file = "df1.csv")
write.csv(df1, file = "df2.csv")
write.csv(df1, file = "df3.csv")
write.csv(df1, file = "df4a.csv")
write.csv(df1, file = "df4b.csv")
```

Explicación de tibble

```
vignette("tibble")
```

```
## starting httpd help server ... done
```

La mayoría de las funciones que usaras en este libro producen tibbles, ya que son una de las características transversales de tidyverse. Si ya estas familiarizado con data.frame(), es importante que tomes en cuenta que tibble () hace menos cosas como nunca cambia el tipo de los inputs (por ejemplo nunca convierte caracteres en factores), nunca cambia el nombre de las variables y nunca asigna nombres a las filas

Empezar a ordenar datos con la tabla4a (pivotar)

```
t4a_PIVOTANTE = tabla4a %>%
  pivot_longer(cols = c ("1999", "2000"), names_to = "anio", values_to = "casos")
```

Exportar resultado: tabla ordenada

```
write.csv(t4a_PIVOTANTE, file = "t4a_PIVOTANTE.csv")
```

Hecho con gusto por Leislle R. Manjarrez O.

Laboratorio 26- Tidy data- Pivotar y unir tablas- Parte 2 Ver tabla4b (df4b)

Ordenar los datos de la tabla4b (pivotar)

```
t4b_PIVOTANTE = tabla4b %>%
  pivot_longer(cols = c ("1999", "2000"), names_to = "anio", values_to = "poblacion")
```

Combinar las versiones ordenadas de la tabla4a y tabla4b (ocupando dplyr)

```
union_t4 = left_join(t4a_PIVOTANTE, t4b_PIVOTANTE)
```

```
## Joining with `by` = join_by(pais, anio)`
```

Ver encabezados tabla4b

```
head(tabla4b)
```

```
## # A tibble: 3 × 3
##   pais      `1999`      `2000`
##   <chr>      <dbl>      <dbl>
## 1 Afganistán 19987071 20595360
## 2 Brasil    172006362 174504898
## 3 China     1272915272 1280428583
```

Exportar resultado: tabla ordenada

```
write.csv(union_t4, file = "union_t4.csv")
```

Ejercicio 3: Datos anchos con la tabla 2 Pivotar la tabla2 a lo ancho

Ver encabezados tabla2

```
head(tabla2)
```

```
## # A tibble: 6 × 4
##   pais      anio tipo      cuenta
##   <chr>    <dbl> <chr>    <dbl>
## 1 Afganistán 1999 casos      745
## 2 Afganistán 1999 población 19987071
## 3 Afganistán 2000 casos      2666
## 4 Afganistán 2000 población 20595360
## 5 Brasil    1999 casos      37737
## 6 Brasil    1999 población 172006362
```

Ordenar datos con la tabla2 (pivotar a lo ancho)

```
t2_ancha = tabla2 %>%
  pivot_wider(names_from = tipo, values_from = cuenta)
```

Ver encabezados t2_ancha

```
head(t2_ancha)
```

```
## # A tibble: 6 × 4
##   pais      anio casos población
##   <chr>    <dbl> <dbl>    <dbl>
## 1 Afganistán 1999    745  19987071
## 2 Afganistán 2000   2666  20595360
## 3 Brasil    1999  37737  172006362
## 4 Brasil    2000  80488  174504898
## 5 China     1999 212258 1272915272
## 6 China     2000 213766 1280428583
```

Exportar resultado: tabla ordenada

```
write.csv(t2_ancha, file = "t2_ancha.csv")
```

Hecho con gusto por Leislle R. Manjarrez O.

Laboratorio 27- Tidy data- Separar datos - Parte 3

En este laboratorio utilizaremos la tabla3 (df3) que tiene un problema distinto pues en este caso la tasa es el resultado del número de casos y la poblacion por lo que se realizara la separacion de dicha tabla

La columna tasa contiene dos variables (casos y población)

Ver tabla3

```
head(df3)
```

```
## # A tibble: 6 × 3
##   pais      anio tasa
##   <chr>    <dbl> <chr>
## 1 Afganistán 1999 745/19987071
## 2 Afganistán 2000 2666/20595360
## 3 Brasil    1999 37737/172006362
## 4 Brasil    2000 80488/174504898
## 5 China     1999 212258/1272915272
## 6 China     2000 213766/1280428583
```

Separacion 1: separar casos y población por default y ver resultado

```
Separado_1 = tabla3 %>%
  separate(tasa, into = c("casos", "poblacion"))

head(Separado_1)
```

```
## # A tibble: 6 × 4
##   pais      anio casos poblacion
##   <chr>    <dbl> <chr>  <chr>
## 1 Afganistán 1999 745    19987071
## 2 Afganistán 2000 2666   20595360
## 3 Brasil     1999 37737  172006362
## 4 Brasil     2000 80488  174504898
## 5 China      1999 212258 1272915272
## 6 China      2000 213766 1280428583
```

Separacion 2: separar casos y población por caracter "/" "***" "-" y ver resultado

```
Separado_2 = tabla3 %>%
  separate(tasa, into = c("casos", "poblacion"), sep = "/")

head(Separado_2)
```

```
## # A tibble: 6 × 4
##   pais      anio casos poblacion
##   <chr>    <dbl> <chr>  <chr>
## 1 Afganistán 1999 745    19987071
## 2 Afganistán 2000 2666   20595360
## 3 Brasil     1999 37737  172006362
## 4 Brasil     2000 80488  174504898
## 5 China      1999 212258 1272915272
## 6 China      2000 213766 1280428583
```

Separacion 3: separar el año por siglo y año con 2 dígitos cada uno y ver resultado

```
Separado_3 = tabla3 %>%
  separate(anio, into = c("siglo", "anio"), sep = 2)

head(Separado_3)
```

```
## # A tibble: 6 × 4
##   pais      siglo anio  tasa
##   <chr>    <chr> <chr> <chr>
## 1 Afganistán 19    99    745/19987071
## 2 Afganistán 20    00    2666/20595360
## 3 Brasil     19    99    37737/172006362
## 4 Brasil     20    00    80488/174504898
## 5 China      19    99    212258/1272915272
## 6 China      20    00    213766/1280428583
```

Ahora vamos a unir la tabla generada con anterioridad Podemos usar unite() para unir las columnas siglo y anio creadas en el ejemplo anterior y ver el resultado posteriormente

```
Union_1 = Separado_3 %>%
  unite(nueva, siglo, anio)

head(Union_1)
```

```
## # A tibble: 6 × 3
##   pais      nueva tasa
##   <chr>    <chr> <chr>
## 1 Afganistán 19_99 745/19987071
## 2 Afganistán 20_00 2666/20595360
## 3 Brasil     19_99 37737/172006362
## 4 Brasil     20_00 80488/174504898
## 5 China      19_99 212258/1272915272
## 6 China      20_00 213766/1280428583
```

Para quitar el guion bajo (_) entre los valores de la union generada previamente usamos sep y "", igual ver resultado

```
Union_2 = Separado_3 %>%  
  unite(nueva, siglo, anio, sep = "")
```

```
head(Union_2)
```

```
## # A tibble: 6 × 3  
##   pais      nueva tasa  
##   <chr>    <chr> <chr>  
## 1 Afganistán 1999  745/19987071  
## 2 Afganistán 2000 2666/20595360  
## 3 Brasil     1999 37737/172006362  
## 4 Brasil     2000 80488/174504898  
## 5 China      1999 212258/1272915272  
## 6 China      2000 213766/1280428583
```