Programación científica en R

Relational data

Marcos Ehekatzin García Guzmán

Agosto de 2024

Introducción

- Típicamente, para hacer un análisis cuantitativo necesitaremos varias tablas de datos combinadas entre sí.
- Colectivamente, a estas tablas se les llama “relational data”.
- Para trabajar con relational data utilizaremos tres familias de verbos:
  1. Mutating joins: Agrega nuevas variables de un data frame a otro, haciendo match entre sus observaciones.
  2. Filtering joins: Filtramos observaciones de un data frame con base en si hacen match o no con alguna observación de otra tabla.
  3. Set operations: Trata a las observaciones como si fueran elementos de un conjunto.

Ejemplo de relational data: ENOE

- La enoe se compone de 5 tablas que se relacionan entre sí

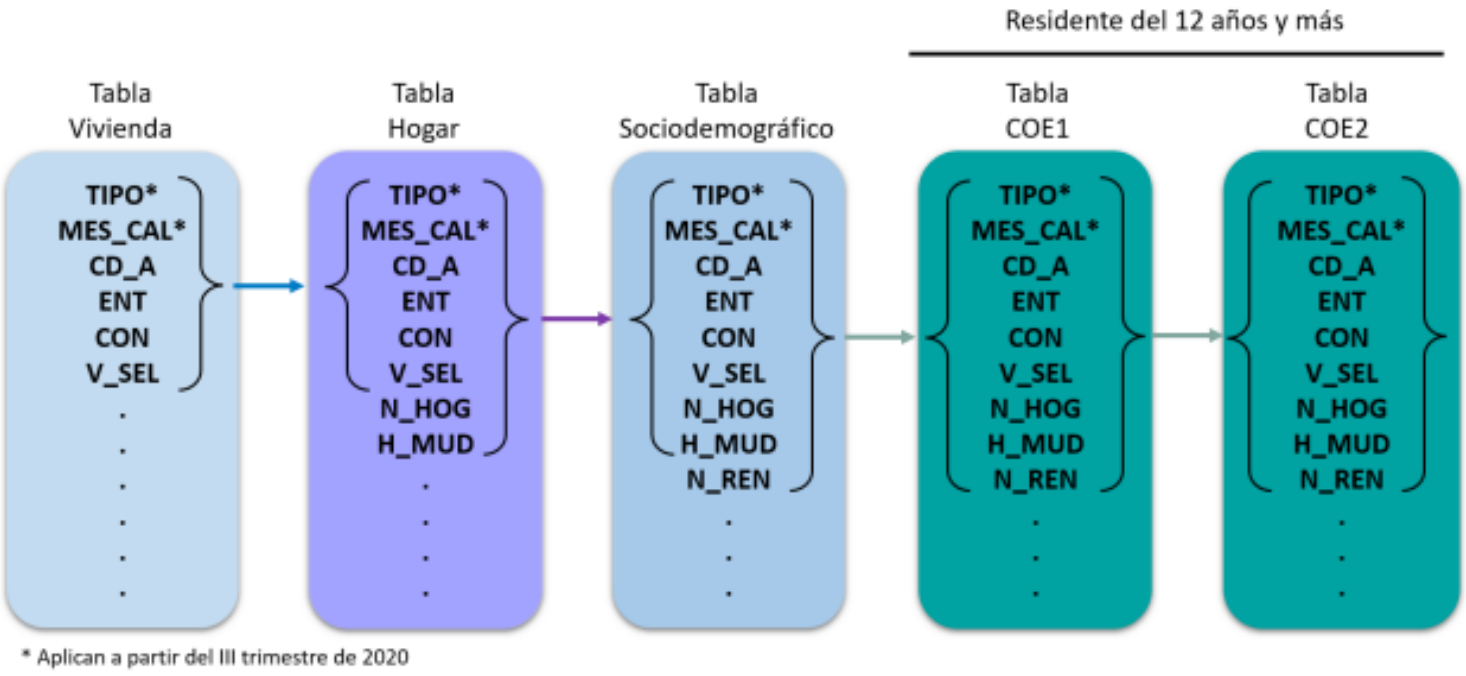

Figura 1: ENOE: Tablas y relaciones

Ejemplo de relational data: ENIGH

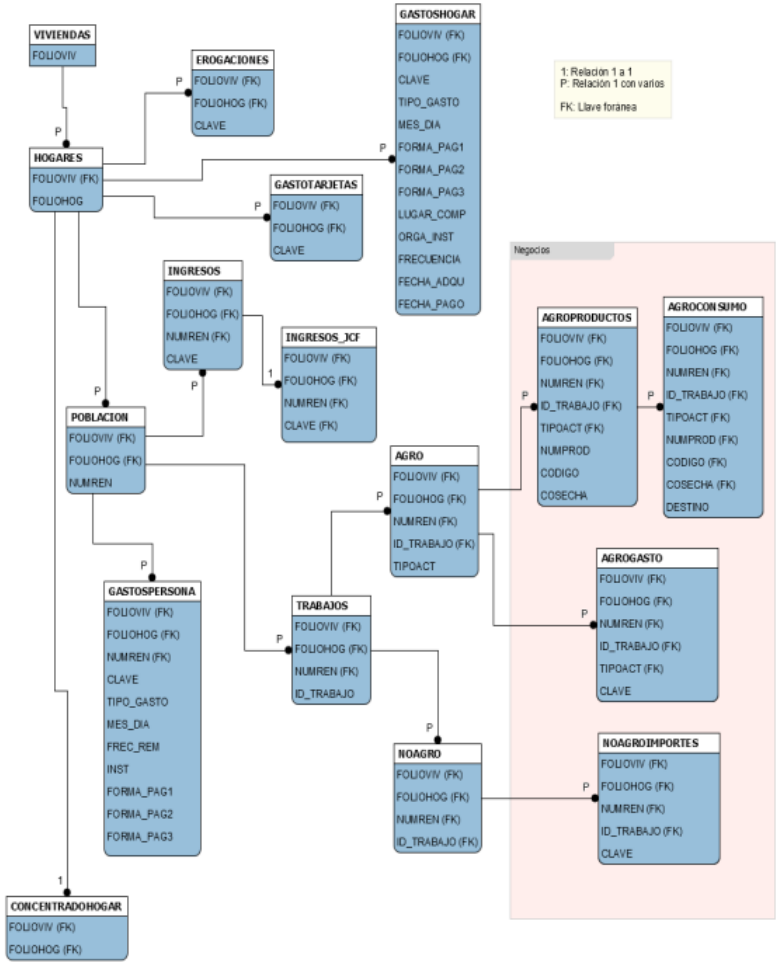

Figura 2: ENIGH: Tablas y relaciones

Variables llave (*keys*)

- Para conectar observaciones de diferentes data frames utilizaremos *keys*.
- Las *keys* pueden ser una sola variable o un conjunto de variables.
- Hay dos tipos de *keys*
  1. Primarias: Que identifican a una observación dentro de su propia tabla.
  2. Externas: Identifica a una observación en otros data frames.

- Utilizaremos la base de datos `nycflights13`, incluida en la librería del mismo nombre.
- Esta base contiene cinco tablas que se encuentran relacionadas.

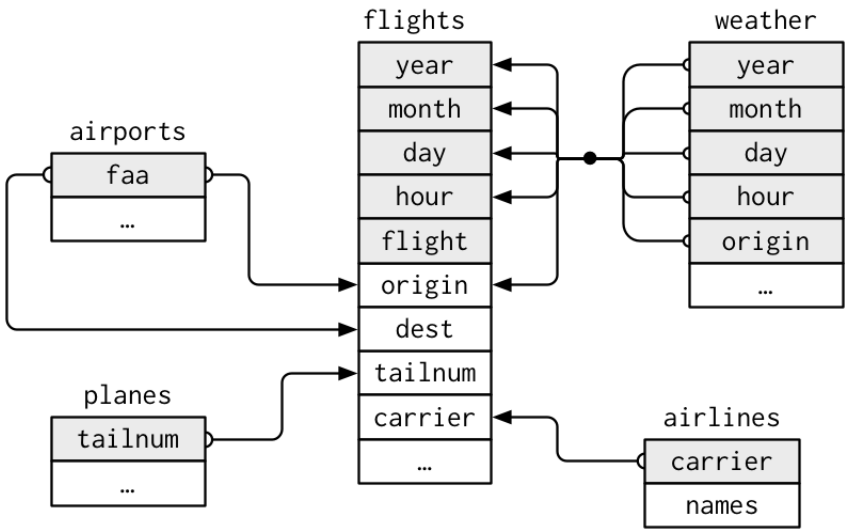

Figura 3: nycflights13

Mutating joins

- Con un mutating join podemos combinar dos data frames con base en los keys de cada observación
- Se llaman mutating joins porque, como el verbo `mutate()`, se agregan variables nuevas a un data frame.
- Hay diferentes clases de mutating joint. Cada uno lo utilizaremos según lo vayamos requiriendo:
  1. left_join(x,y): Conserva todas las observaciones en x
  2. right_join(x,y): Conserva todas las observaciones en y
  3. full_join(x,y): Conserva todas las observaciones en x y y.
  4. inner_join(x,y): Conserva solo las observaciones que aparezcan tanto en x como en y.

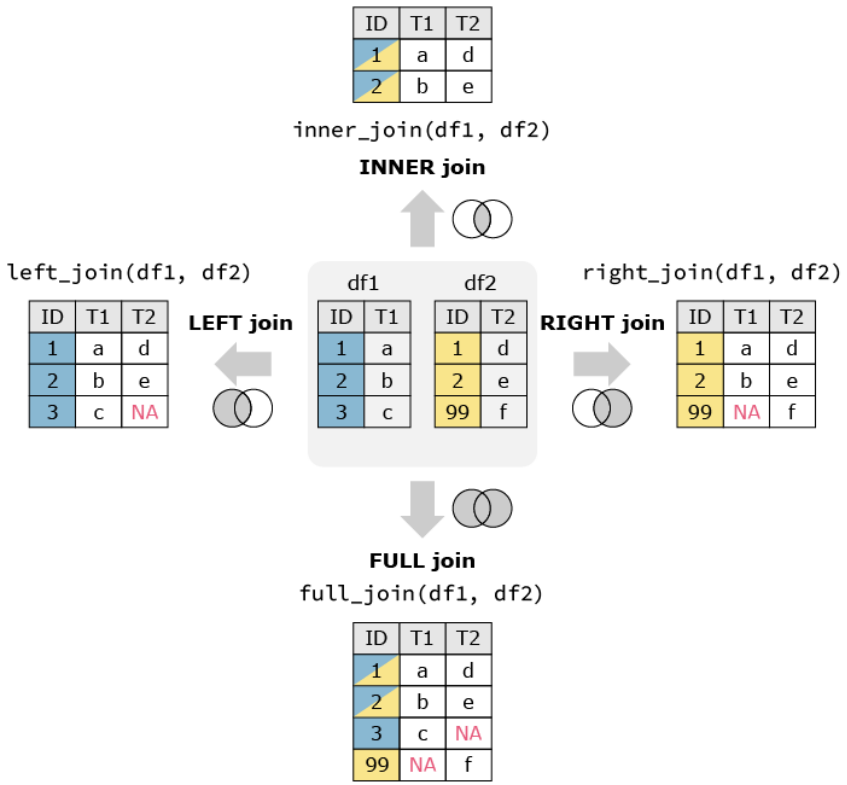

Figura 4: Mutating joins

Mutating joins

- Imaginemos que queremos agregar el nombre de la aerolínea a la tabla de vuelos. Podemos hacerlo utilizando el verbo `left_join()`:

```
flights2 <- flights %>%
  select(year:day, hour, tailnum, carrier)

flights2 <-  left_join(flights2,airlines, by = "carrier")
```

year	month	day	hour	tailnum	carrier	name
2013	1	1	5	N14228	UA	United Air Lines Inc.
2013	1	1	5	N24211	UA	United Air Lines Inc.
2013	1	1	5	N619AA	AA	American Airlines Inc.
2013	1	1	5	N804JB	B6	JetBlue Airways
2013	1	1	6	N668DN	DL	Delta Air Lines Inc.
2013	1	1	5	N39463	UA	United Air Lines Inc.

Mutating joins: Duplicate keys

- Nos encontraremos con keys duplicadas cuando querramos relacionar datos con relación *one-to-may*.

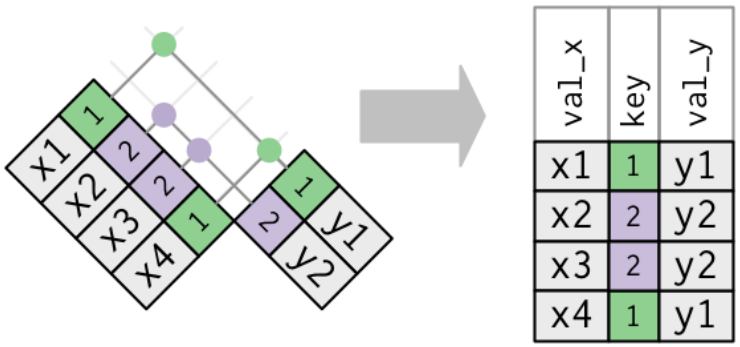

Figura 5: One-to-many join

Mutating joins: Duplicate keys

- Solamente una de las tablas debe tener duplicados, si las dos tienen duplicados será un error.

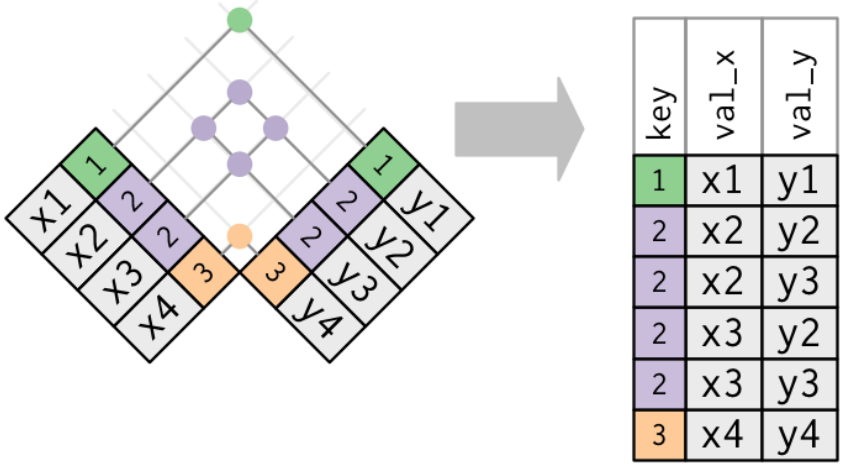

Figura 6: many-to-many join

Filtering joins

- Estos joins funcionan como el verbo `filter()`. Por tanto, van a afectar a las observaciones y no a las variables.

1. semi_join(x,y): Conservará a todas las observaciones de x que hagan match con las observaciones de y.
2. anti_join(x,y): Conservará todas las observaciones de x que **NO** hagan match con alguna observación de y.

- No hay problema en relaciones many-to-many

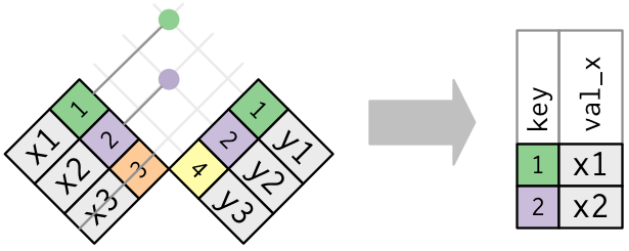

Figura 7: Semi join

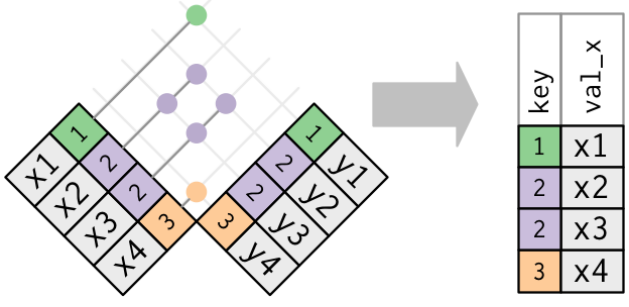

Figura 8: Semi join many-to-many

Ejercicio

1. Con la ENOE:
   a. Calcule la tasa de informalidad por estado. Hint: Utilice la variable emp_ppal.
   b. El porcentaje de trabajadores que ganan menos de un salario mínimo por estado. Hint: Utilice la variable salario (salario mínimo mensual)
   c. Guarde ambos estadísticos en un solo df.
2. Con la ENIG:
   a. Por estado, calcule qué porcentaje de hogares que viven en condiciones de pobreza y pobreza extrema por ingresos. Hint: La línea de pobreza extrema es de 2124.70 por persona y la línea de pobreza es de 4246.06 por persona. Utilice la variable tot_integ (Total de integrantes del hogar)
3. Combine los data frame y calcule la correlación entre el porcentaje de hogares que viven en pobreza y la tasa de informalidad/porcentaje de trabajadores que ganan menos que el mínimo.