

Uniendo Dataframes

Contenidos

- 01.** Paquete dplyr
- 02.** ¿Cómo unimos tablas de datos en R?
- 03.** Funciones join: `left_join` y otros join
- 04.** Funciones ensamblado o pegado



01.

Paquete dplyr



Antes de comenzar: Paquete dplyr

- **Dplyr** corresponde a un paquete correspondiente al universo **Tidyverse**, que permite la manipulación y operaciones con *dataframes*.
- Para utilizarlo, primero debemos instalar el paquete:

```
install.packages('dplyr')
```

- Posteriormente debemos "llamar" a la librería correspondiente, lo que permitirá su uso:

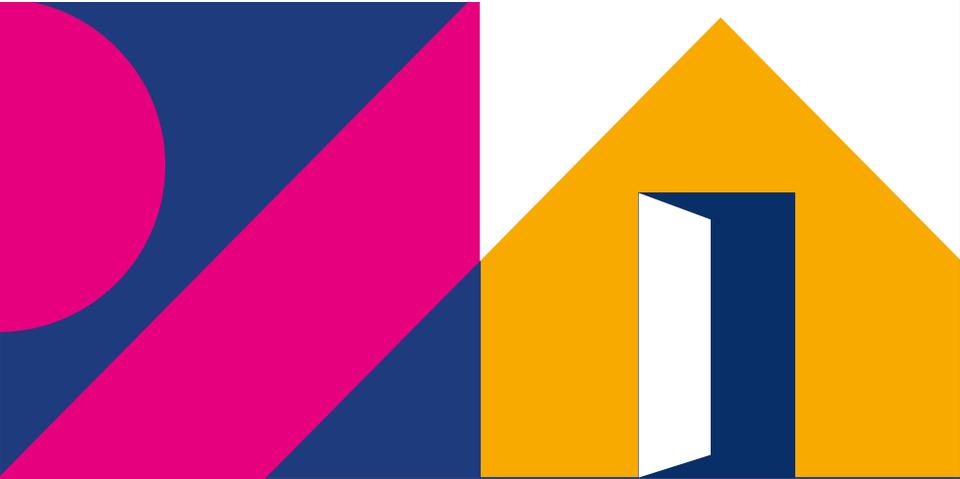
```
library(dplyr)
```





02.

**¿Cómo unimos tablas
de datos en R?**





Uniando dataframes: funciones

- Básicamente, existen dos maneras de unir tablas de datos, también llamadas dataframes, en dplyr, ya sea con las **funciones join y las funciones de ensamblaje (denominada "bind")**.
- En primer lugar, nos enfocaremos en las **funciones join**.



03.

**Funciones join: left join
y otros join**





Funciones join

Las funciones join agregan las columnas de un *dataframe* (**x**) con uno (**y**), siempre respetando el identificador único de cada observación.

Creamos dos dataframes y ejecutamos códigos:

```
x <- data.frame(id=c(1,2,3,5),  
                x=c("x1", "x2", "x3", "x5"))  
y <- data.frame(id= c(1,2,3,4),  
                y= c("y1", "y2", "y3", "y4"))
```



Visualización
consola:

```
x  
##      id  x  
## 1    1 x1  
## 2    2 x2  
## 3    3 x3  
## 4    5 x5
```

```
y  
##      id  y  
## 1    1 y1  
## 2    2 y2  
## 3    3 y3  
## 4    4 y4
```



Función `left_join()`

La función `left_join()`, permite pegar la columna y del dataframe `y`, en el dataframe `x`.

Introducimos y ejecutamos código:

```
left_join(x, y, by = "id")
```

Visualizamos consola:

```
##   id  x   y
## 1  1 x1  y1
## 2  2 x2  y2
## 3  3 x3  y3
## 4  5 x5 <NA>
```

En este caso podemos ver como `left_join()`, excluye el cuarto valor del dataframe "y"

- Los joins mantienen el orden de las filas
- El identificador utilizado en el parametro "by", puede ser más de una columna

`left_join(x, y)`

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

Otros joins

- **Mutating joins:** agregan columnas

`right_join(x, y)`

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

`inner_join(x, y)`

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

`inner_join(x, y)`

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

Otros Joins

- **Filtering joins:** permiten filtrar observaciones

`semi_join(x, y)`

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

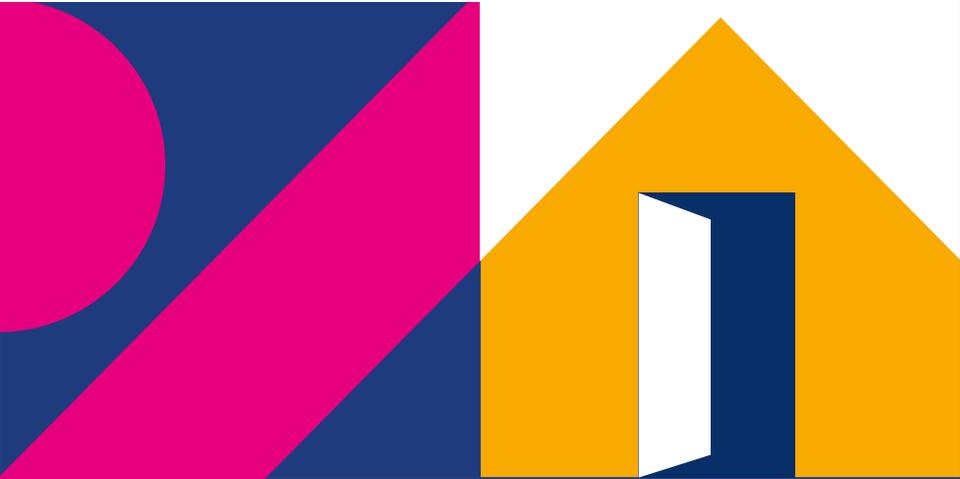
`anti_join(x, y)`

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4



04.

**Funciones ensamblado
o pegado**



¿Para qué nos sirven estas funciones?

Estas funciones permiten "pegar" dos *dataframes*, independiente de su identificador único, tanto, sumando nuevas observaciones **bind_rows()** o sumando nuevas columnas con **bind_cols()**.

Escribimos y ejecutamos códigos:

Visualización consola:

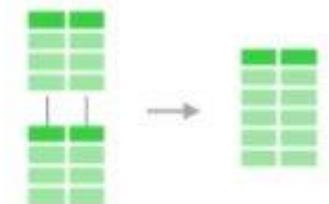
bind_rows(x,y)

```
##   id   x   y
## 1  1  x1 <NA>
## 2  2  x2 <NA>
## 3  3  x3 <NA>
## 4  5  x5 <NA>
## 5  1 <NA> y1
## 6  2 <NA> y2
## 7  3 <NA> y3
## 8  4 <NA> y4
```

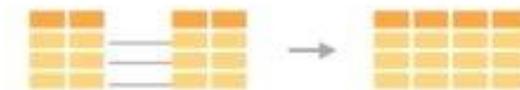
bind_cols(x,y)

```
## id...1 x id...3 y
## 1 1 x1 1 y1
## 2 2 x2 2 y2
## 3 3 x3 3 y3
## 4 5 x5 4 y4
```

bind_rows()



bind_cols()



¡Muchas gracias!





www.ine.gob.cl