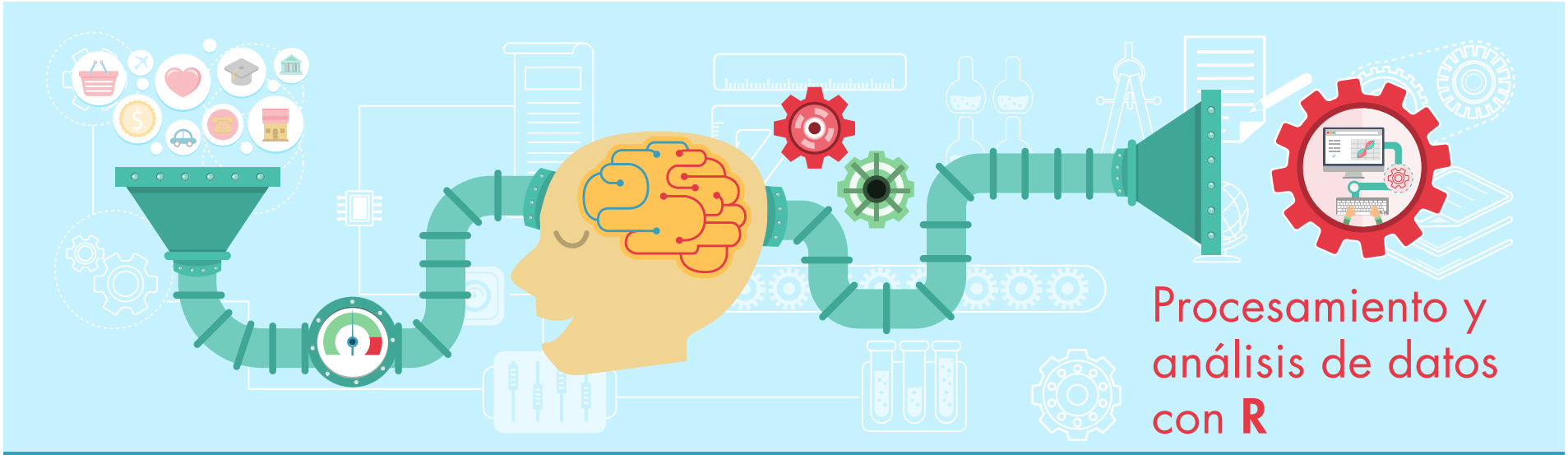

Añadir variables (*merge/join*): pegado de bases por el lado

Para realizar este procedimiento lo primero que se requiere son dos tablas: la izquierda (x) y la derecha (y). La tabla que se combina contendrá además de las variables comunes, las que hacen parte decada una de ellas.

Existen al menos 5 formas de realizar el pegado de bases por el lado: producto cartesiano (no es muy usado en el contexto de manejo de bases de datos), *inner_join, outer_join, left_join, right_join* entre otras. A continuación, se explican las principales.

Para pegar dos tablas es necesario que exista una o varias variables comunes a ambas y que se indique cuáles de los registros se van a combinar. Al realizar el procedimiento (*merging*) se conforma un nuevo *Dataframe* con las variables de las dos bases de datos. Aquellas que son comunes, permiten emparejar los registros entre los conjuntos y aparecerán una vez en la nueva tabla.

Añadir variables (*merge/join*): pegado de bases por el lado

El siguiente diagrama ilustra de manera sencilla los tipos de pegados:

a

x1	x2
A	1
B	2
C	3

+ b

x1	x2
A	T
B	F
D	T

=

Mutating Joirs

x1	x2	x3
A	1	T
B	2	F
C	3	NA

dplyr::**left_join(a, b, by = "x1")**
Join matching rows from b to a.

x1	x2	x3
A	T	1
B	F	2
D	T	NA

dplyr::**rigth_join(a, b, by = "x1")**
Join matching rows from a to b.

x1	x2	x3
A	1	T
B	2	F

dplyr::**inner_join(a, b, by = "x1")**
Join data. Retain only rows in both sets.

x1	x2	x3
A	1	T
B	2	F
C	3	NA
D	NA	T

dplyr::**full_join(a, b, by = "x1")**
Join data. Retain all values, all rows.

Suponga que se tienen las siguientes dos tablas y luego observe los diferentes tipos de pegados:

```
datos1 <- data.frame(IdCliente = paste0("Cli
ente", c(1:6)),
     Producto = c(rep("TV", 3), rep("Radio", 3)))
datos1
```

```
  IdCliente      Producto
1 Cliente1         TV
2 Cliente2         TV
3 Cliente3         TV
4 Cliente4       Radio
5 Cliente5       Radio
6 Cliente6       Radio
```

```
> datos2 <- data.frame(IdCliente = paste0("Cli
ente", c(2, 4, 6, 7, 8)),
+ Departamento = c(rep("Valle", 2),
+ rep("Antioquia", 1),
+ "Atlántico", "Bolivar"))
> datos2
```

Añadir variables (*merge/join*): pegado de bases por el lado

```
  IdCliente    Departamento
1 Cliente1       Valle
2 Cliente2       Valle
3 Cliente3     Antioquia
4 Cliente4     Atlántico
5 Cliente5      Bolivar
```

1. Inner joint: busca los registros comunes a ambas tablas y conforma una común.

```
> df_inner

  IdCliente      Producto     Departamento
1 Cliente1         TV            Vale
2 Cliente2       Radio          Valle
3 Cliente3       Radio        Antioquia
```

2. Outer joint: los datos que aparezcan en al menos una de las dos tablas se incluyen en la resultante. Observe que se debe especificar el argumento **all = T.** No es necesario utilizar el argumento **by** si los dos conjuntos tienen una variable única común; sin embargo, es una buena práctica hacerlo.

```
df_outer <- merge(x = datos1, y = datos2,
                  by = "IdCliente", all = T)

df_outer

  IdCliente      Producto     Departamento
1 Cliente1         TV            <NA>
2 Cliente2         TV            Valle
3 Cliente3         TV            <NA>
4 Cliente4       Radio           Valle
5 Cliente5       Radio           <NA>
6 Cliente6       Radio         Antioquia
7 Cliente7        <NA>         Atlántico
8 Cliente8        <NA>          Bolivar
```

3. Left joint: los registros (individuos) del lado izquierdo de la tabla son los únicos que aparecerán en la resultante. Por medio de este pegado se traerán las variables de la tabla derecha de dichos individuos. Es necesario especificar el argumento **all.x = T.** Observe que solamente quedan los registros que se encontraban en la tabla de la izquierda (datos1).

Añadir variables (*merge/join*): pegado de bases por el lado

```
df_left <- merge(x = datos1, y = datos2, by =
"IdCliente",
+ all.x = TRUE)
df_left
```

	IdCliente	Producto	Departamento
1	Cliente1	TV	<NA>
2	Cliente2	TV	Valle
3	Cliente3	TV	<NA>
4	Cliente4	Radio	Valle
5	Cliente5	Radio	<NA>
6	Cliente6	Radio	Antioquia

4. Right joint: es análogo al pegado por la izquierda. En este caso solo se conservan los registros de la tabla derecha y la resultante estará conformada por las variables de las dos tablas. Se debe utilizar el argumento **all.y = T.**

```
df_left <- merge(x = datos1, y = datos2, by =
"IdCliente",
+ all.x = TRUE)
df_left
```

	IdCliente	Producto	Departamento
1	Cliente2	TV	Valle
2	Cliente4	Radio	Valle
3	Cliente6	Radio	Antioquia
4	Cliente7	<NA>	Atlántico
5	Cliente8	<NA>	Bolivar

5. Producto cartesiano: en el producto cartesiano se realiza una integración de las dos tablas y se conservan todas las posibles combinaciones entre los individuos de la primera tabla y los individuos de la segunda tabla.

```
df_cross <- merge(x = datos1, y = datos2, by =
NULL)
df_cross
```

	IdCliente.x	Producto	IdCliente.y	Departamento
1	Cliente1	TV	Cliente2	Valle
2	Cliente2	TV	Cliente2	Valle
3	Cliente3	TV	Cliente2	Valle
4	Cliente4	Radio	Cliente2	Valle

Añadir variables (*merge/join*): pegado de bases por el lado

```
 5 Cliente5    Radio      Cliente2       Valle
 6 Cliente6    Radio      Cliente2       Valle
 7 Cliente1       TV      Cliente4       Valle
 8 Cliente2       TV      Cliente4       Valle
 9 Cliente3       TV      Cliente4       Valle
10 Cliente4    Radio      Cliente4       Valle
11 Cliente5    Radio      Cliente4       Valle
12 Cliente6    Radio      Cliente4       Valle
13 Cliente1       TV      Cliente6    Antioquia
14 Cliente2       TV      Cliente6    Antioquia
15 Cliente3       TV      Cliente6    Antioquia
16 Cliente4    Radio      Cliente6    Antioquia
17 Cliente5    Radio      Cliente6    Antioquia
18 Cliente6    Radio      Cliente6    Antioquia
19 Cliente1       TV      Cliente7     Atlántico
20 Cliente2       TV      Cliente7     Atlántico
21 Cliente3       TV      Cliente7     Atlántico
22 Cliente4    Radio      Cliente7     Atlántico
23 Cliente5    Radio      Cliente7     Atlántico
24 Cliente6    Radio      Cliente7     Atlántico
25 Cliente1       TV      Cliente8      Bolivar
26 Cliente2       TV      Cliente8      Bolivar
27 Cliente3       TV      Cliente8      Bolivar
28 Cliente4    Radio      Cliente8      Bolivar
29 Cliente5    Radio      Cliente8      Bolivar
30 Cliente6    Radio      Cliente8      Bolivar
```

Además de las formas de hacer el pegado que se acaban de explicar, el paquete **dplyr** dispone de funciones que son de manera computacional más rápidas y trabajan muy bien con grandes bases de datos. Conózcalas a continuación. Ejecútelas en R para comprender mejor su manejo:

```
library(dplyr)
inner_join(datos1, datos2)
left_join(datos1, datos2)
right_join(datos1, datos2)
outer_join(datos1, datos2)
full_join(datos1, datos2)
```

Integrar variables comunes con nombres diferentes

Es muy frecuente cuando se quieren integrar dos *Dataframes* que estos engan una variable común con nombres diferentes. Note que en el siguiente ejemplo la primera columna se denomina **idCliente** y la segunda **Id.**

Añadir variables (*merge/join*): pegado de bases por el lado

```
datos1 <- data.frame(IdCliente = paste0("Cli
ente", c(1:6)),
          Producto = c(rep("TV", 3), rep("Radio",
3)))
> datos1

   IdCliente    Producto
  1 Cliente1       TV
  2 Cliente2       TV
  3 Cliente3       TV
  4 Cliente4     Radio
  5 Cliente5     Radio
  6 Cliente6     Radio

datos2 <- data.frame(Id = paste0("Cliente", c(2,
4, 6, 7, 8)),
Departamento = c(rep("Valle", 2), rep("Antio
quia", 1),"Atlántico", "Bolivar"))
datos2

   IdCliente   Departamento
  1 Cliente2       Valle
  2 Cliente4       Valle
  3 Cliente6    Antioquia
  4 Cliente7    Atlántico
  5 Cliente8     Bolivar
```

```
merge(datos1, datos2, by.x = "IdCliente", by.y =
"Id")
   IdCliente    Producto Departamento
  1 Cliente2       TV        Valle
  2 Cliente4     Radio       Valle
  3 Cliente6     Radio     Antioquia
```

Lo anterior también es posible cuando los nombres de las bases son diferentes y se quieren integrar múltiples variables. Esto es usual cuando se pretenden pegar conjuntos por datos como departamento, municipio, sector, sección y manzana. Note que en este caso las tablas son **xy**, **x2** *y* **y2**.

```
d1 <- data_frame( x = letters[1:3],  y = LET
TERS[1:3],   a = rnorm(3)  )
d2 <- data_frame(x2 = letters[3:1],  y2 = LET
TERS[3:1], b = rnorm(3)  )
left_join(d1, d2, by = c("x" = "x2", "y" = "y2"))
Source: local data frame [3 x 4]
     x y          a          b
    1 a A  0.3924069  1.3073880
    2 b B  0.3397483 -0.3811417
    3 c C -0.2541935  0.4381216

merge(d1, d2, by.x = c("x", "y"), by.y = c("x2",
"y2"), all.x = T)
```