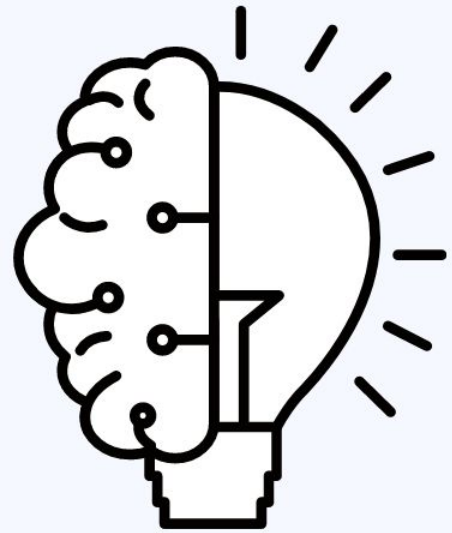


Ciencia de datos

Para el sector público de salud



Estructura de datos y operadores

En colaboración con...



@RLadiesConce



MÓDULO 1: Nivelación y conceptos básicos

Actividades sincrónicas (2 hrs cada uno)

Fecha	Hora	Tema
09/12/2020	18.45h	Aspectos generales del curso
15/12/2020	18.45h	Introducción a R y RStudio
17/12/2020	18.45h	Estructura de datos y operadores
22/12/2020	18.45h	Importación de datos
29/12/2020	18.45h	Análisis prefactibilidad y valor público

Estructura y tipos de datos

Los tipos de datos de uso más común en R son los siguientes.

Tipo	Ejemplo	Nombre en inglés
Entero	1	integer
Numérico	1.3	numeric
Cadena de texto	“uno”	character
Factor	uno	factor
Lógico	TRUE	logical
Perdido	NA	NA
Vacio	NULL	null

Fechas, datos espaciales (raster), imágenes...

Coerción

En R, los datos pueden ser coercionados, es decir, forzados, para transformarlos de un tipo a otro.

Lo anterior ocurre porque no todos los tipos de datos pueden ser transformados a los demás, para ello se sigue una regla general : **La coerción de tipos se realiza de los tipos de datos más restrictivos a los más flexibles.**

```
lógico -> entero -> numérico -> cadena de texto ( logical -> integer -> numeric -> character )
```

Coerción explícita

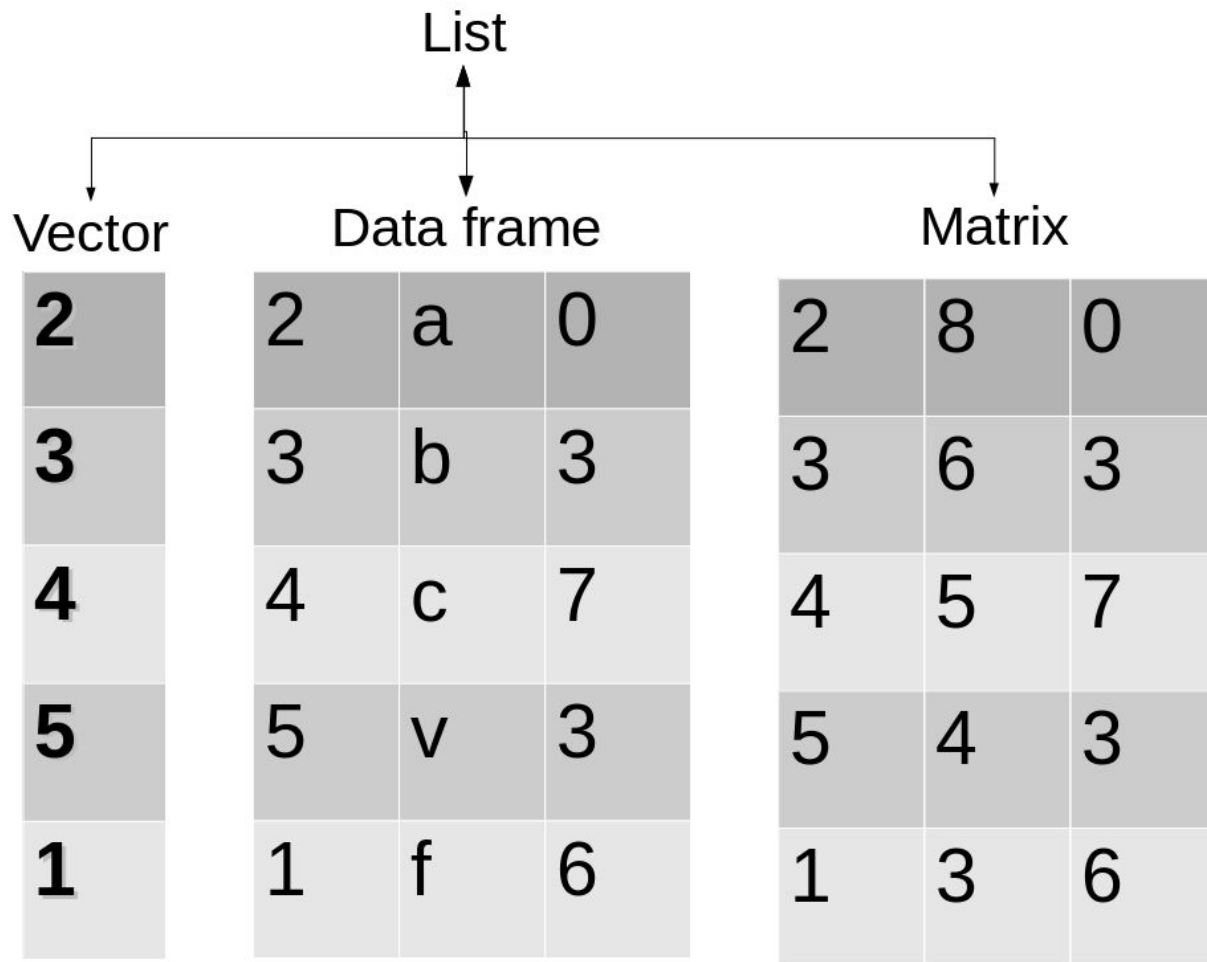
También podemos hacer coerciones **explícitas** usando la familia de funciones `as()` .

Función	Tipo al que hace coerción
<code>as.integer()</code>	Entero
<code>as.numeric()</code>	Numerico
<code>as.character()</code>	Cadena de texto
<code>as.factor()</code>	Factor
<code>as.logical()</code>	Lógico
<code>as.null()</code>	NULL

Los factores son un caso particular para la coerción. Dado que son **valores numéricos con etiquetas**, pueden ser coercionados a tipo numérico y cadena de texto; y los datos numéricos y cadena de texto pueden ser coercionados a factor. Sin embargo, al coercionar un factor tipo numérico, perdemos sus niveles.

Estructura de datos

- Vector: Un conjunto lineal de datos (secuencia de datos, serie de tiempo, lógicos, nombres...)
- Matrix: Una tabla con solo números
- Data Frame: Una tabla donde cada columna tiene un tipo de datos (estándar dorado)
- List: Aquí podemos meter lo que queramos



Tidy Data

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20095360
Brazil	1999	37737	172006362
Brazil	2000	80488	174004898
China	1999	212258	1272015272
China	2000	216766	128042583

variables

country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20095360
Brazil	1999	37737	172006362
Brazil	2000	80488	174004898
China	1999	212258	1272015272
China	2000	216766	128042583

observations

country	year	cases	population
Afghanistan	999	745	19987071
Afghanistan	000	2666	20095360
Brazil	999	37737	172006362
Brazil	000	80488	174004898
China	999	212258	1272015272
China	000	216766	128042583

values

Cada columna una variable

Cada fila una observación

messy

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

tidy

name	trt	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Operadores

Operadores principales

Aritméticos

Relacionales

Lógicos

De asignación

Operadores aritméticos

Operador	Operación	Ejemplo	Resultado
+	Suma	<code>5 + 3</code>	8
-	Resta	<code>5 - 3</code>	2
*	Multiplicación	<code>5 * 3</code>	18
/	División	<code>5 / 3</code>	1.666667
^	Potencia	<code>5 ^ 3</code>	125
%%	División entera	<code>5 %% 3</code>	2

Operadores relacionales

Operador	Comparación	Ejemplo	Resultado
<	Menor que	5 < 3	FALSE
<=	Menor o igual que	5 <= 3	FALSE
>	Mayor que	5 > 3	TRUE
>=	Mayor o igual que	5 >= 3	TRUE
==	Exactamente igual que	5 == 3	FALSE
!=	No es igual que	5 != 3	TRUE

Operadores lógicos

Operador	Comparación	Ejemplo	Resultado
<code>x y</code>	x Ó y es verdadero	<code>TRUE FALSE</code>	<code>TRUE</code>
<code>x & y</code>	x Y y son verdaderos	<code>TRUE & FALSE</code>	<code>FALSE</code>
<code>!x</code>	x no es verdadero (negación)	<code>!TRUE</code>	<code>FALSE</code>
<code>isTRUE(x)</code>	x es verdadero (afirmación)	<code>isTRUE(TRUE)</code>	<code>TRUE</code>

Los operadores `|` y `&` siguen estas reglas:

- `|` devuelve `TRUE` si alguno de los datos es `TRUE`
- `&` solo devuelve `TRUE` si ambos datos es `TRUE`
- `|` solo devuelve `FALSE` si ambos datos son `FALSE`
- `&` devuelve `FALSE` si alguno de los datos es `FALSE`

Operadores de asignación

Operador	Operación
<code>< -</code>	Asigna un valor a una variable
<code>=</code>	Asigna un valor a una variable

Orden de los operadores

Orden	Operadores
1	<code>^</code>
2	<code>*</code> <code>/</code>
3	<code>+</code> <code>-</code>
4	<code><</code> <code>></code> <code><=</code> <code>>=</code> <code>==</code> <code>!=</code>
5	<code>!</code>
6	<code>&</code>
7	<code> </code>
8	<code><-</code>