CLASE 05 - MANIPULAR Y TRANSFORMAR DATOS.

Diplomado en Análisis de datos con R para la Acuicultura.

Dr. José Gallardo Matus

Pontificia Universidad Católica de Valparaíso.

16 April 2022

PLAN DE LA CLASE

1.- Introducción

- ¿Para qué manipular datos?
- Diferencia entre Tidy and messy data.
- Paquete tidyr.
- Operador pipe (Tuberías).
- Paquete dplyr.

2). Práctica con R y Rstudio cloud.

- Realizar manipulación de datos con tidyr y dplyr.
- Realizar gráficas avanzadas con ggplot2.

MANIPULACIÓN DE DATOS

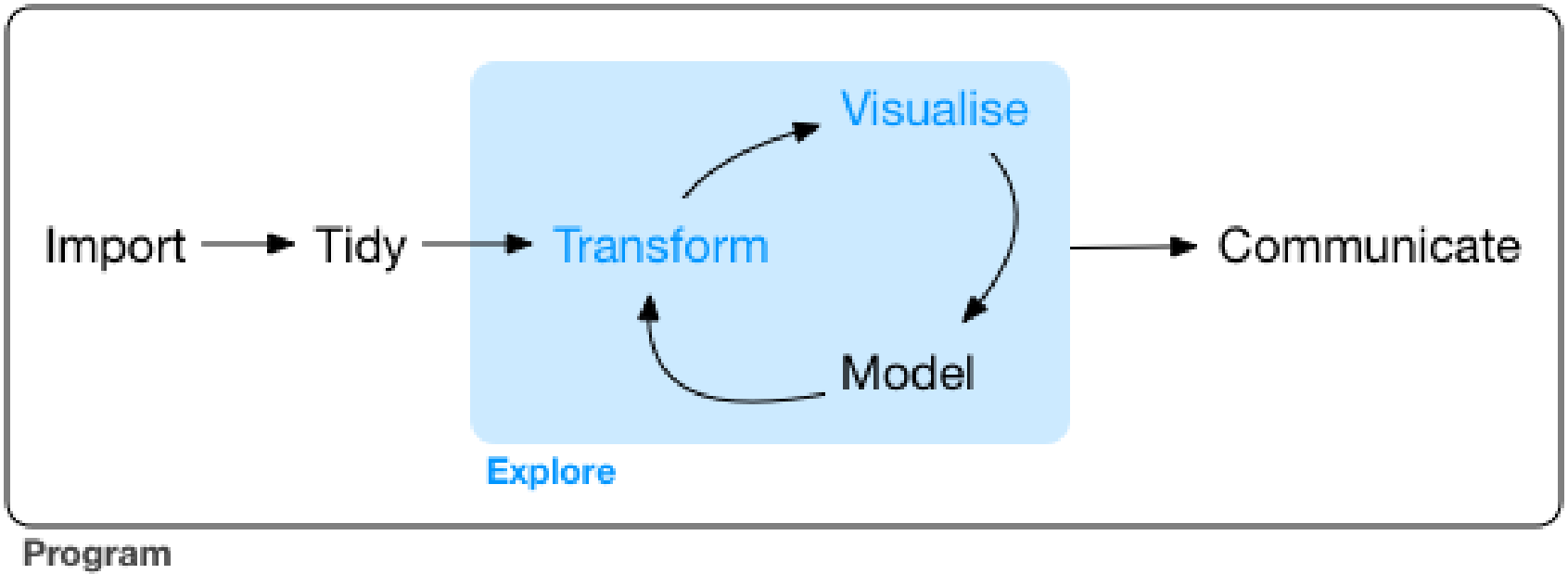
¿Para qué manipular datos?

- Para hacer datos más legibles y organizados.
- Para dar formato adecuado previo a visualización y análisis estadístico.

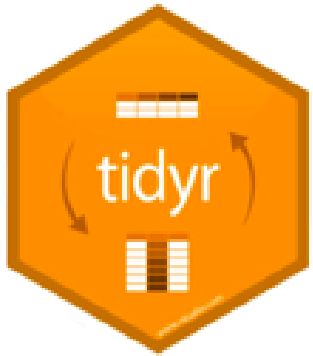
Ejemplos de tareas comunes durante esta etapa:

- Filtrar datos por categorías.
- Remover o imputar datos faltantes.
- Agrupar datos por algún criterio.
- Seleccionar y calcular estadísticos.
- Generar variables derivadas a partir de variables existentes.
- Transformar variables.

ETAPAS DEL ANÁLISIS DE DATOS



PAQUETES CLAVE



DATOS TIDY - MESSY

Tidy data (datos ordenados)

- Cada columna es una variable.
- Cada fila es una observación.
- Cada celda es un simple dato o valor.

Messy data (desordenados)

- Cualquier conjunto de datos que no cumple alguno de estos criterios.

EJEMPLO DATOS MESSY

¿Por qué son messy?

Variable	Replica	Especie A	Especie B	Especie C
peso	1	174	NA	135
peso	2	155	103	138
peso	3	131	138	135
parásitos	1	25	8	5
parásitos	2	12	3	8
parásitos	3	4	11	NA

EJEMPLO DATOS TIDY

¿Por qué son tidy?

Pez	Especie	Sexo	Peso	Parásitos
1	A	Hembra	174	25
2	A	Hembra	155	12
3	A	Hembra	131	4
4	B	Macho	NA	8
5	B	Macho	103	3
6	B	Hembra	138	11
7	C	Hembra	135	5
8	C	Macho	138	8
9	C	Hembra	135	NA

PAQUETE TIDYR: FUNCIONES CLAVE

gather(): Colapsa múltiples columnas para crear tidy data.
spread(): Separa una columna en múltiples columnas.

Bahía	2019	2020	2021
Valparaíso	12	13	14
Concepción	10	11	12

gather(...)

spread(...)

Bahía	Año	TSM
Valparaíso	2019	12
Concepción	2019	10
Valparaíso	2020	13
Concepción	2020	11
Valparaíso	2021	14
Concepción	2021	12

gather("Año","TSM",2:4)

spread("Año","TSM")

EL OPERADOR PIPE: %>%.

En programación **pipe** es una técnica que permite pasar información de un proceso o programa a otro por etapas.

Evita pipe cuando: a) Deseas manipular varios objetos a la vez. b) Un paso intermedio genera un objeto que luego deseas analizar separadamente.

PAQUETE DPLYR: FUNCIONES BÁSICAS

select(): Permite extraer o seleccionar variables/columnas específicas de un data.frame.

filter(): Para filtrar desde una tabla de datos un subconjunto de filas. Ej. solo un nivel de un factor, observaciones que cumplen algún criterio (ej. > 20).

mutate(): Permite calcular/generar nuevas variables "derivadas". Útil para calcular proporciones, tasas.

arrange(): Permite ordenar la base de datos según una variable de forma ascendente o descendente.

PAQUETE DPLYR: SELECT()

Datos %>% select(Año, TSM)

Bahía	Año	TSM
Valparaíso	2019	12
Concepción	2019	10
Valparaíso	2020	13
Concepción	2020	11
Valparaíso	2021	14
Concepción	2021	12

Año	TSM
2019	12
2019	10
2020	13
2020	11
2021	14
2021	12

PAQUETE DPLYR: FILTER()

Datos %>% filter(Bahía="Valparaíso")

Bahía	Año	TSM
Valparaíso	2019	12
Concepción	2019	10
Valparaíso	2020	13
Concepción	2020	11
Valparaíso	2021	14
Concepción	2021	12

Bahía	Año	TSM
Valparaíso	2019	12
Valparaíso	2020	13
Valparaíso	2021	14

PAQUETE DPLYR: MUTATE()

```
Datos %>% mutate(Fahrenheit=(TSM*1.8)+32)
```

Bahía	Año	TSM
Valparaíso	2019	12
Concepción	2019	10
Valparaíso	2020	13
Concepción	2020	11
Valparaíso	2021	14
Concepción	2021	12

Bahía	Año	TSM	Fahrenheit
Valparaíso	2019	12	53,6
Concepción	2019	10	50,0
Valparaíso	2020	13	55,4
Concepción	2020	11	51,8
Valparaíso	2021	14	57,2
Concepción	2021	12	53,6

PAQUETE DPLYR: GROUP_BY + SUMMARIZE()

```
Datos %>% grup_by(Bahía) %>%
          summarize(n=n(),
                    promedio=mean(TSM))
```

Bahía	Año	TSM
Valparaíso	2019	12
Concepción	2019	10
Valparaíso	2020	13
Concepción	2020	11
Valparaíso	2021	14
Concepción	2021	12

Bahía	Año	TSM
Valparaíso	2019	12
Valparaíso	2020	13
Valparaíso	2021	14
Concepción	2019	10
Concepción	2020	11
Concepción	2021	12

Bahía	n	Promedio
Valparaíso	3	13
Concepción	3	11

PAQUETE DPLYR: JOIN

tb1

Index	TSM
Valparaíso	12
Concepción	10
Puerto Montt	13

tb2

Index	O2
Valparaíso	8
Puerto Montt	9
Punta Arenas	10

left_join(tb1, tb2, by = "Index")

Index	TSM	O2
Valparaíso	12	8
Concepción	10	*NA*
Puerto Montt	13	9

full_join(tb1, tb2, by = "Index")

Index	TSM	O2
Valparaíso	12	8
Concepción	10	*NA*
Puerto Montt	13	9
Punta Arenas	*NA*	10

ERRORES COMUNES PARA IMPORTAR DATOS - 1

Tabla de datos experimentales

sample_id	Weight	sex
1	17,2	female
2	18,8	female
3	27,8	male
4	20,4	male
5	20,6	MALE
6	28,6	male
7	22,3	MALE
8	13,7	female
9	16,6	female
10	17,8	Female
11	26,1	male
12	21,8	male
13	22	male
14	20,6	male
15	17,2	female
16	28,9	male
17	22,5	macho
18	10,2	female
19	23,5	male
20	17,6	female
21	14,7	female
22	18,9	female
23	14,9	female
24	16,4	female

Promedio de Etiquetas	Total
female	15,8
macho	22,5
male	23,9
Total genera	**19,8**

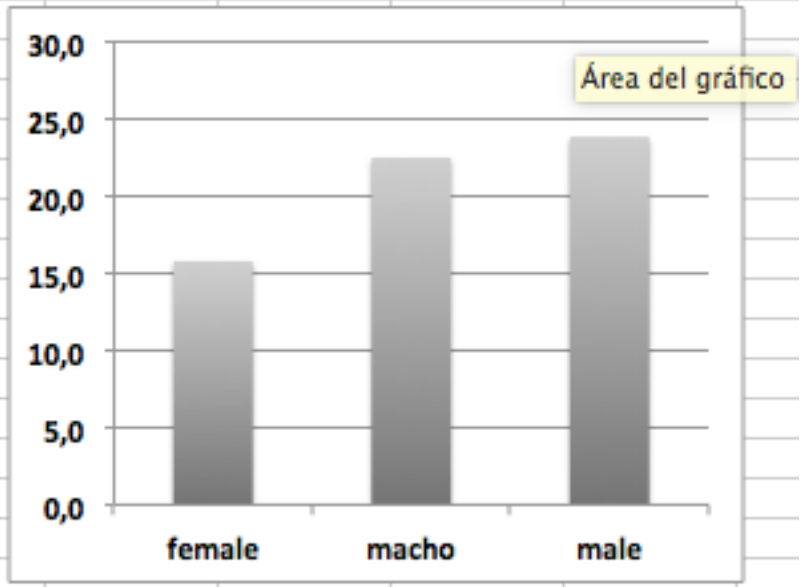


Figure 1: Errores comunes antes de importar a excel

ERRORES COMUNES PARA IMPORTAR DATOS - 2

sample_id	Weight	sex
1	17,2	female
2	18,8	female
3	27,8	male
4	20,4	male
5	20,6	male
6	28,6	male
7	sin registro	male
8	13,7	female
9	16,6	female
10	17,8	female
11	26,1	male
12	21,8	male
13	22	Indeterminado
14	20,6	male
15	17,2	female
16	28,9	male
17	22,5, cola deforme	male
18	10,2	female
19	23,5	male

sample_id	Weight	sex	Observaciones
1	17,2	female	
2	18,8	female	
3	27,8	male	
4	20,4	male	
5	20,6	male	
6	28,6	male	
7		male	
8	13,7	female	
9	16,6	female	
10	17,8	female	
11	26,1	male	
12	21,8	male	
13	22	NA	Sexo Indeterminado
14	20,6	male	
15	17,2	female	
16	28,9	male	
17	22,5	male	cola deforme
18	10,2	female	
19	23,5	male	

Figure 2: Errores comunes antes de importar a excel

Importante: No colocar comentarios en las celdas de datos. Dejar celdas vacias o usar el simbolo *NA* es preferido cuando hay datos faltantes.

RESUMEN DE LA CLASE

- Diferenciamos datos ordenados (Tidy) y desordenados (Messy).
- Manipulamos datos con tidyr y dplyr.
- Utilizamos tuberías o pipe **%>%**.
- Hicimos gráfico ggplot2 usando datos transformados y variables derivadas.