Tutoría 2

Exploración de bases de datos y manipulación de tablas

Sofía Madariaga

Pontificia Universidad Católica de Chile

Aproximación a las Políticas Públicas desde los datos

Taller de Análisis de Datos I

7 de agosto de 2023

Materiales

En el portal del diplomado Materiales del Curso > Tutorías > Tutoría 2: https://ep.ingenieriauc.cl/mod/folder/view.php?id=87282
tutoria-2.zip

- `tutoria-2.r` ← aquí estaremos trabajando
- `tutoria-2.RProj`
- `ENS.xlsx`
- `Netflix_movies_and_TV_Shows_database.sav`
- `Netflix_movies_and_TV_Shows_database.csv`
- `tutoria-2_presentacion.pdf`

Comunicados iniciales

- **Reorrección** de la tarea 1: enviar la solicitud al profesor al profesor.
- Durante la semana, entregaré la **corrección de los scripts**. Plazo máximo: mediados de la próxima semana.
- Algunos comentarios generales sobre la tarea 1:
  - Interpretación de estadísticos descriptivos, **no narrar**.
  - Presentar las **tablas en el formato académico**, no imágenes.
  - Trabajen en proyectos y/o incorporen el archivo de proyecto en su carpeta comprimida.
  - Incorporen **todos los archivos**: especialmente la base de datos.

Outline

Proyectos

¿Por qué usarlos?

- Asegurar que mi código sea **reproducible**, debo trabajar con proyectos en Rstudio.
- En el curso exigimos la reproductibildiad del código.
- Además, es una buena prácitca a la hroa de presentar mi trabajo.

Proyectos

Cómo generar un proyecto

1. **Nuevo directorio.** Cuando no he creado una carpeta, Rstudio crea una carpeta
2. **Directorio existente.** Ya cree una carpeta, y Rstudio crea un archivo de proyecto en esa carpeta.

Carpeta = directorio.

Vamos a verlo en la práctica $\rightarrow$

Proyectos

Conceptos

- **Path (sendero):** es la dirección de mi carpeta. Literalmente, es el string (dato que va entre comillas), que indica la dirección de mi carpeta.
  - `"C:/Users/Sofia/Descargas/"`
- **Directorio:** es la carpeta a la que hace alusión el *path*.
  - *Fijar el directorio alduiendo al path:*
    `setwd("C:/Users/Sofia/Descargas/")`
- **Importante:** cuando trabajo en **proyectos**, debo considerar el directorio del proyecto como el **directorio raíz**.

Proyectos

Importar una base de datos

No recomendado

```
data <- read_sav("C:/Users/Sofía/Descargas/input/datos.csv")
```

Recomendado

```
data <- read_sav("datos.csv")
```

Recomendaciones

- Use una carpeta en donde guarde las bases de datos. (Opcional)
- Use las herramientas de importación de bases de datos. (Opcional)

Exploración básica de bases de datos

Para manejar bases de datos:

- Use $ para consultar variables.
- **Funciones básicas para la exploración**:
  - `head()`, `tail()`: ver primeras o últimas observaciones.
  - `str()`, `glimpse()`, `vtable()`: explorar el contenido (variables).

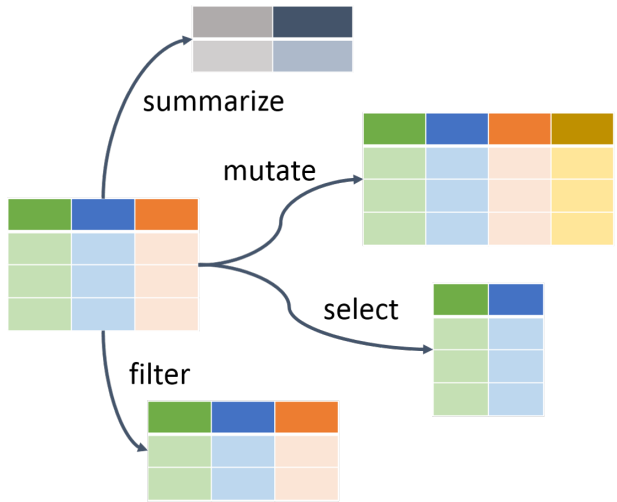
Vamos a R $\rightarrow$

Lubridate: para manejar fechas

- Es un paquete usual en R que nos ayuda a manejar fechas: es muy ineligente para idenitificar cualquier fecha!
- **Documentación:** lubridate.

Vamos a R $\rightarrow$

Paquete dplyr



Paquete dplyr

Repaso de funciones

1. **Select:** seleccionar variables.
   - data %>% **select**(variable_1, variable_2)
2. **Filter:** filtrar filas por condición.
   - data %>% **filter**(variable_1 > n) (ejemplo)
3. **Mutate:** modificar variables.
   - data %>% **mutate**(variable = operación)
4. **Arrange:** ordenar de manera descendente o ascendente una tabla de datos, según una variable numérica o categórica.
   - data %>% **arrange**(variable)

Paquete dplyr

Repaso de funciones

1. **Mutate y case_when:** agrupar variables en categorías generales.
   - Edad: $[-\infty - 25[\rightarrow$ *"15 a 24 años"*.
   - Edad: $[25 - 45[\rightarrow$ *"25 a 44 años"*.
   - Edad: $[45 - 65[\rightarrow$ *"45 a 64 años"*.
   - Edad: $[65 - \infty[\rightarrow$ *"65 años o más"*.

Paquete dplyr

Repaso de funciones

```
ens <- ens %>%
  mutate(categoria_edad = case_when(Edad < 25 ~ 1,
                                    Edad < 45 ~ 2,
                                    Edad < 65 ~ 3,
                                    Edad >= 65 ~ 4),
         categoria_edad = factor(categoria_edad, levels = c(1,2,3,4),
                                 labels = c("15 a 24 años",
                                            "25 a 44 años",
                                            "45 a 64 años",
                                            "65 años o más")))
```

Paquete dplyr

Repaso de funciones

1. **Group by y summarise:** agrupar y resumir esta agrupación.
2. **Otras funciones:** rename(), slice(), entre otras.

Tablas de Resumen con dplyr

Importante

Es importante que reporte su tabla en un formato académico.

1. Indique el título de la tabla con la enumeración correspondiente.
2. Indique la fuente de dónde provienen los datos.
3. No poner imágenes en su lugar.

Tablas de Resumen con dplyr

Tabla 1: películas y series según el año de estreno.

Año de estreno	Tipo de obra	N	%
2020	Movie	517	54 %
2020	TV Show	436	45 %
2021	Movie	277	46 %
2021	TV Show	315	53 %

Fuente: Elaborado a partir de los datos *Netflix Movies and TV Shows Database* (2023).

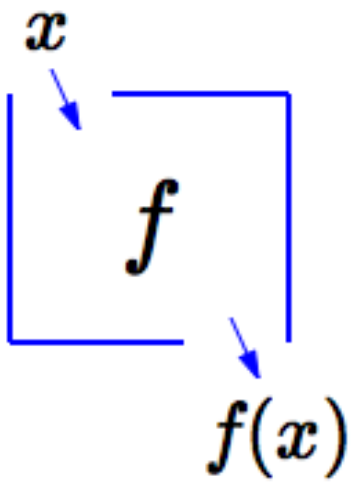
Tablas de Resumen con dplyr

Ejercicios

1. **Ejercicio 1:** Cuántas películas han sido añadidas por netflix por año e indique el director/a.
2. **Ejercicio 2:** Cálcule el promedio de cigarros por rango etario que fuman personas hipertensas. Organice por por sexo y solo incluya los rangos etarios 25-44, 45-65 y 65 o más.

Funciones

Funciones

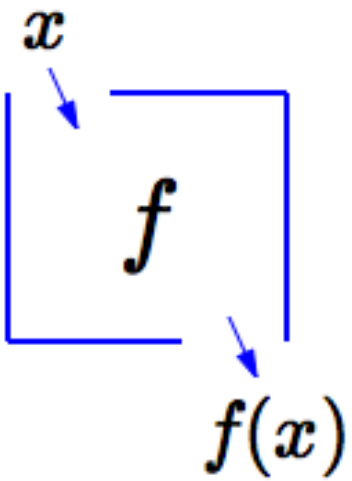


En R, podemos aplicar **funciones** a los objetos:

```
1 length(objeto)
2 mean(objeto)
3 sd(objeto)
```

Crear una función

Funciones




```
1 mi_funcion <- function(x){
2     (bloque de codigo)
3     ...
4     resultado <- codigo
5     return(resultado)
6 }
```