Introducción al Análisis Exploratorio de Datos (EDA) en R

Módulo 2

2024-10-22

Paquetes en R

Paquetes en R

Los paquetes en R son colecciones de funciones, datos y documentación cuyo objetivo es extender las capacidades básicas de R. **CRAN** (The Comprehensive R Archive Network) es una red de servidores que almacenan versiones de R, así como librerías en R que cumplen las políticas del repositorio (CRAN, 2022).

Para instalar paquetes del repositorio **CRAN**:

```
install.packages("dplyr")
```

Después de instalar el paquete, se debe cargar la librería:

```
library(dplyr)
```

Para encontrar la documentación del paquete:

```
help(dplyr)
```

Tidyverse

Tidyverse es un conjunto de librerías en R diseñadas para el análisis de datos (importar, transforma, visualizar y modelar con datos) (Wickham, 2019).

Nos concentraremos en las siguientes librerías:

- dplyr
- ggplot2
- forcats*

Figure 1: Librerías en Tidyverse

Importar datos

Importar datos

El primer paso es definir el directorio de trabajo:

```
setwd("path")
```

Nos concentraremos en funciones para importar los siguientes formatos de datos

Formato	Formato específico	Función	Paquete
Texto o tabulares	CSV	read_csv()	readr
	Otros formatos de texto	read_delim()	readr
Formatos de otros programas	Excel	read_excel()	readxl
	SPSS	read_sav()	haven
	STATA	read_dta()	haven
	SAS	read_sas()	haven
Formatos propios de R	.rda	load()	base
	.rds	readRDS()	base

Base de datos

La base de datos usada es extraída de los microdatos de la **Gran Encuesta Integrada de Hogares (GEIH)** para diciembre de 2023. El análisis considera las siguientes 13 ciudades y áreas metropolitanas:

- Medellín A.M.
- Barranquilla A.M.
- Bogotá
- Cartagena
- Manizales A.M.
- Monteria
- Villavicencio
- Pasto
- Cucuta A.M.
- Pereira A.M.
- Pereira A.M.
- Bucaramanga A.M.
- Ibague
- Cali A.M

La información es extraída de dos módulos de la GEIH:

- **Ocupados** (horas trabajadas , ingreso laboral, actividad económica, etc.)
- **Características generales, seguridad social en salud y educación** (edad, sexo, nivel de educación, etc.)

Base de datos (cont.)

Para importar la base de datos (.xlsx),

```
library(readxl)
dataset <- readxl::read_excel("Datos/Formatos/geih_dataset.xlsx")
```

La siguiente tabla muestra un resumen de la base de datos:

Variable	Clase	Descripción
area	Factor	Área metropolitana
dpto	Factor	Departamento
sexo	Factor	Sexo al nacer
parent	Factor	Parentesco con el jefe o jefa del hogar
edad	Numérica	Años cumplidos
edu	Factor	Mayor nivel educativo alcanzado
ingreso	Numérica	Ingreso laboral
horas_semana	Numérica	Horas trabajadas normalmente a la semana
cotiza	Factor	¿Cotiza a un fondo de pensiones?
lugar	Factor	Lugar principal de trabajo
meses	Numérica	¿Cuántos meses trabajó en los últimos 12 meses?
rama_4	Cadena	Rama de actividad CIIU REV 4 (4 dígitos)
rama_2	Cadena	Rama de actividad CIIU REV 4 (4 dígitos)
posic	Factor	Posición laboral
fondo	Factor	¿A cuál fondo cotiza?
cambiar	Factor	¿Desea cambiar su trabajo?
estable	Factor	¿Considera que su empleo es estable?
t_actual	Numérica	¿Cuánto tiempo lleva en su empleo actual?
t_viaje	Numérica	Tiempo de desplazamiento al trabajo
mas_h	Factor	¿Quiere trabajar más horas?
medio	Factor	Medio de transporte
sintrab	Factor	¿Si no tuviera trabajo, de dónde obtendría sus recursos?
n_comp	Factor	¿Cuántas personas tiene la empresa donde trabajo?
srl	Factor	¿Afiliación a ARL?
caja	Factor	¿Afiliación a caja de compensación familiar?
actividad	Factor	Actividad económica recodificada
cotiza_fondo	Factor	Fondo de pensiones recodificado
factor_exp	Numérica	Factor de expansión

Herramientas para la manipulación de datos

Pipe (%>%)

La tubería de comando o *pipeline* (%>%) es una herramienta utilizada para el encadenamiento de funciones. El operador nos permite escribir una secuencia de operaciones

Una secuencia en su **forma estándar** sigue la forma

```
dataset_2 <- dplyr::filter(dataset, attend > 15 & attend != 20)
```

En **forma encadenada**:

```
dataset_2 <- dataset %>% dplyr::filter(attend > 15 & attend != 20)
```

El siguiente atajo es útil: 

Dplyr

El paquete **dplyr** proporciona una sintaxis para la manipulación de datos. (El operador %>% pertenece a la sintaxis de dplyr). Nos concentraremos en las siguientes funciones:

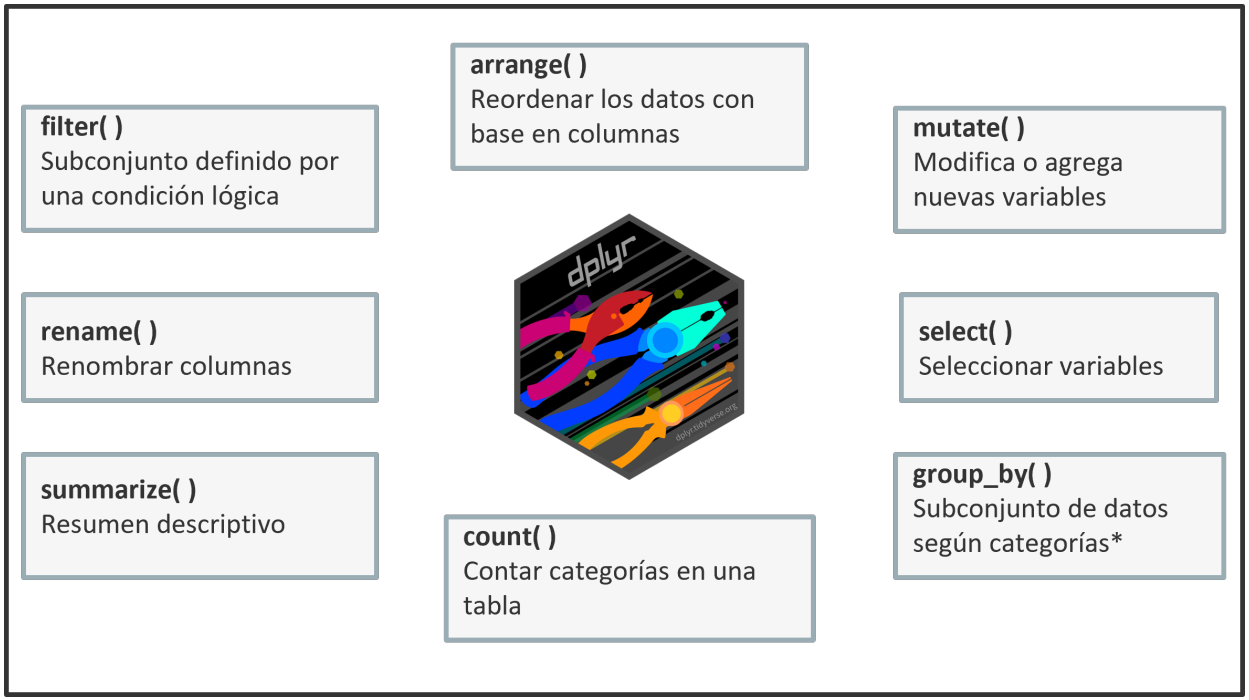


Figure 2: Algunas funciones en el paquete dplyr

Resumen por grupo

Usando las funciones `summarize()` y `group_by()`, obtenemos un resumen descriptivo de la base de datos diferenciado según una o más variables de control. Por ejemplo:

```
# Resumen general
table_1 <- new_dataset %>% filter(Int_attend == "Group 4")
%>%   summarize(MeanAttend = mean(attend),SdAttend = sd(attend))
```

```
# Resumen diferenciado
table_2 <- new_dataset %>% group_by(Int_attend) %>%
  summarize(MeanAttend = mean(attend), SdAttend = sd(attend))
```

La **Figura 10** muestra el funcionamiento de `summarize()` y `group_by()`.

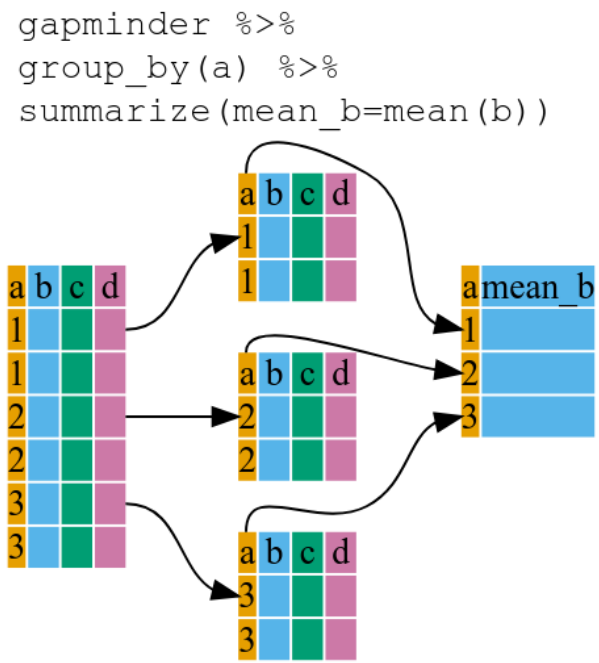
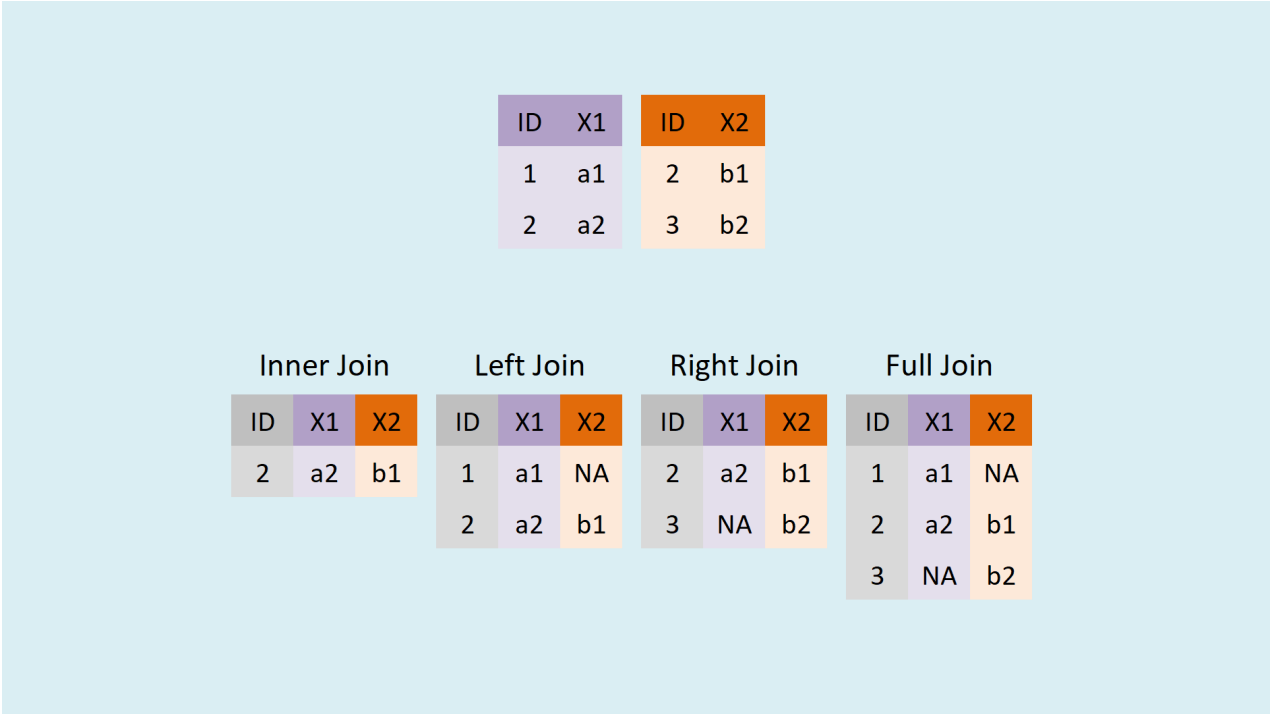


Figure 3: Caption for the picture.

Uniones de bases de datos

Funciones en `dplyr`:

- `left_join(x, y)`
- `right_join(x, y)`
- `inner_joind(x,y)`
- `full_join(x,y)`



ggplot2

El paquete **ggplot2** proporciona un sistema coherente para visualizar datos y crear gráficos. La versatilidad de **ggplot2** radica en el uso de la Gramática de Gráficos (*Grammar of Graphics*).

```
ggplot(dataset, aes()) + geometría + faceta + opciones
```

donde:

1. *dataset* es un data frame
2. Las características del mapa **aes()** describe los ejes (x, y), el color exterior (**color** o **colour**), el color interior (**fill**), la forma de los puntos (**shape**), el tipo de línea (**linetype**) y el tamaño (**size**)
3. Los objetos geométricos (**geometría**) determinan el tipo de gráfico:
   - Puntos (*geom_point*)
   - Líneas (*geom_lines*)
   - Histogramas (*geom_histogram*)
   - Boxplot (*geom_boxplot*)
4. La **faceta** permite dividir un gráfico en múltiples gráficos de acuerdo con grupos

Recursos alternativos

Recursos alternativos

- Data wrangling with dplyr and tidyr (Cheat Sheet): Recurso 1.2
- Visualización de datos usando ggplot2 (Guía Rápida): Recurso 1.3
- Factors with forcats (Cheat Sheet): Recurso 1.4

Bibliografía de consulta

Bibliografía de consulta

- Wickham, H. (2016) GGplot2. Elegant Graphics for Data Analysis. Springer
- Grolemund, G. (2014). Hands-On Programming with R. O'Reilly Media: Sebastopol, CA.
- Schutt, R. & O'Neil, C. (2014). Doing Data Science. O'Reilly Media: Sebastopol, CA.
- Wickham & Grolemund, G. (2016). R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. O'Reilly Media: Sebastopol, CA.
- Aldás J. & Uriel, E. (2017). Análisis multivariante aplicado con R. Madrid: Ediciones Paraninfo