Introducción a la Limpieza de Datos y Tidyverse

Darwin Del Castillo, MD

01/02/2025

Limpieza de Datos en R

¿Por qué es importante la limpieza de datos?

- Los datos raramente vienen en el formato que necesitamos
- Datos sucios = análisis incorrectos
- La limpieza puede tomar hasta 80% del tiempo de análisis
- Una buena limpieza facilita el análisis posterior

Problemas comunes en datos

- Valores faltantes (NA)
- Datos en formato incorrecto
- Nombres de variables inconsistentes
- Datos duplicados
- Errores de digitación

R Base: Limpieza básica

```r
# Detectar valores NA
is.na(datos)

# Eliminar NA
na.omit(datos)

# Cambiar nombres de columnas
names(datos) <- c("nombre1", "nombre2")

# Convertir tipos de datos
as.numeric(datos$columna)
as.character(datos$columna)
as.factor(datos$columna)
```

Introducción a Tidyverse

¿Qué es Tidyverse?

- Colección de paquetes para ciencia de datos
- Diseñado para trabajar en conjunto
- Filosofía común de datos ordenados
- Sintaxis consistente y moderna

Instalación de Tidyverse

```r
# Instalar tidyverse
install.packages("tidyverse")

# Cargar tidyverse
library(tidyverse)
```

Ejemplo práctico

```r
# Crear datos de ejemplo
set.seed(123)
datos_ejemplo <- data.frame(
  id = 1:5,
  edad = c(25, 30, NA, 45, 50),
  sexo = c("M", "F", "f", "M", "M"),
  peso = c(70, 65, 68, 80, 75)
)
```

```
  id edad sexo peso
1  1   25    M   70
2  2   30    F   65
3  3   NA    f   68
4  4   45    M   80
5  5   50    M   75
```

Limpieza con tidyverse

```
datos_limpios <- datos_ejemplo |>
  # Estandarizar sexo
  mutate(
    sexo = toupper(sexo),
    # Crear IMC
    altura = c(170, 165,
               168, 180,
               175),
    imc = peso / ((altura/100)^2)
  ) |>
  # Remover NA
  drop_na()
```

	id	edad	sexo	peso	altura	imc
1	1	25	M	70	170	24.22145
2	2	30	F	65	165	23.87511
3	4	45	M	80	180	24.69136
4	5	50	M	75	175	24.48980

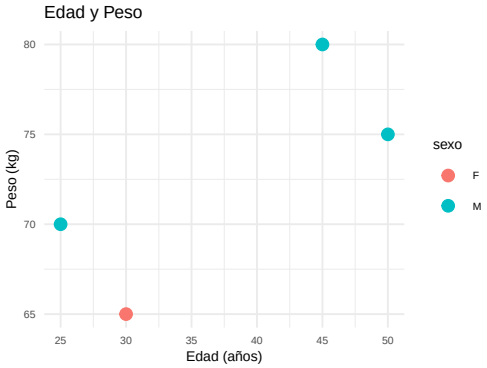
Análisis básico

```r
# Resumen por sexo
tabla_1 <- datos_limpios |>
  group_by(sexo) |>
  summarise(
    n = n(),
    edad_promedio = mean(edad),
    peso_promedio = mean(peso)
  )
```

```
# A tibble: 2 x 4
  sexo      n edad_promedio
  <chr> <int>         <dbl>
1 F         1            30
2 M         3            40
# i 1 more variable:
#   peso_promedio <dbl>
```

Visualización con ggplot2

```
# Crear el gráfico
graph_1 <- ggplot(datos_limpios,
       aes(x = edad,
           y = peso,
           color = sexo)) +
  geom_point(size = 3) +
  labs(
    title = "Edad y Peso",
    x = "Edad (años)",
    y = "Peso (kg)"
  ) +
  theme_minimal() +
  theme(
    text = element_text(size = 8),
    plot.title = element_text(size = 10)
  )
```



dplyr: Funciones principales

```r
# Select: seleccionar columnas
select(datos, columna1, columna2)

# Filter: filtrar filas
filter(datos, columna > 10)

# Mutate: crear/modificar columnas
mutate(datos, nueva = columna1 + columna2)

# Group_by + summarise
datos |>
  group_by(grupo) |>
  summarise(
    promedio = mean(valor),
    total = sum(valor)
  )
```

tidyr: Datos ordenados

```
# De ancho a largo
pivot_longer(
  datos,
  cols = c(col1, col2),
  names_to = "variable",
  values_to = "valor"
)

# De largo a ancho
pivot_wider(
  datos,
  names_from = "variable",
  values_from = "valor"
)
```

Recursos adicionales

- R for Data Science (r4ds.had.co.nz)
- Tidyverse website (tidyverse.org)
- RStudio cheat sheets
- Stack Overflow
- GitHub