Módulo 2 — Tidyverse I

Sección 1: Introducción al Tidyverse

CEPAL - Unidad de Estadísticas Sociales

2025-10-30

¿Qué es el Tidyverse?

El **Tidyverse** es una colección de paquetes de R diseñados para el trabajo con datos de forma coherente y eficiente.
Se centra en la idea de *datos ordenados* (*tidy data*), donde:

- Cada variable es una columna
- Cada observación es una fila
- Cada tipo de unidad observacional forma una tabla

Filosofía del Tidyverse

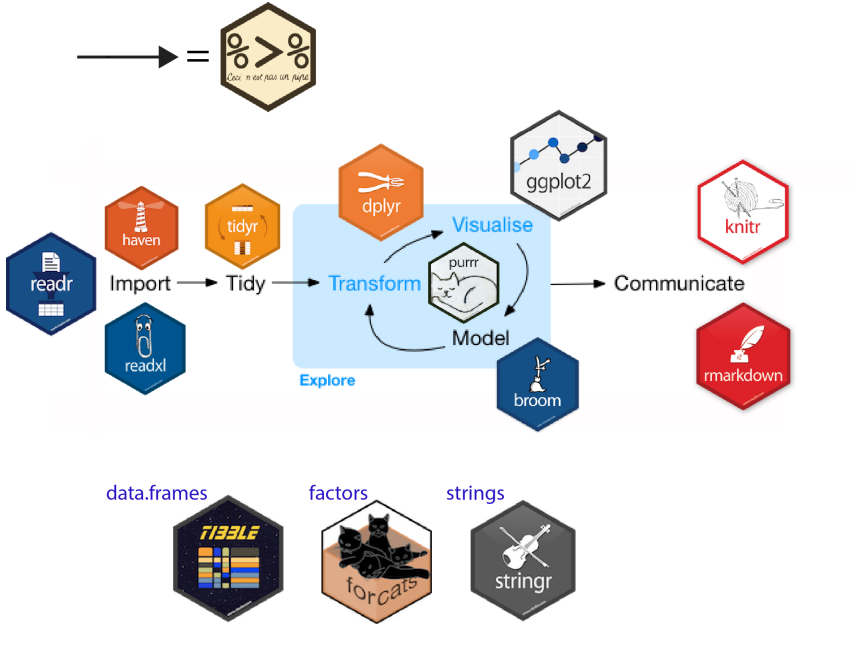
- Coherencia en la sintaxis y nombres de funciones
- Operaciones expresivas y legibles
- Compatibilidad entre paquetes
- Uso del operador %>% (*pipe*) para encadenar acciones

```
library(dplyr)
library(ggplot2)
```

Paquetes principales

Paquete	Objetivo general	Funciones o ejemplos clave
ggplot2	Creación de gráficos avanzados y personalizables mediante la "gramática de los gráficos".	`ggplot()`, `geom_point()`, `geom_bar()`, `facet_wrap()`
dplyr	Manipulación eficiente de datos: filtrar, seleccionar, ordenar, resumir y crear nuevas variables.	`filter()`, `select()`, `mutate()`, `summarise()`, `arrange()`
readr	Lectura y escritura rápida de archivos de texto (CSV, TXT).	`read_csv()`, `write_csv()`, `read_delim()`
tidyr	Estructura y limpieza de datos, asegurando que cada columna sea una variable y cada fila una observación.	`pivot_longer()`, `pivot_wider()`, `separate()`, `unite()`

Ecosistema Tidyverse



Paquetes del Ecosistema Tidyverse (I)

A continuación, se describen los principales paquetes y su función dentro del ecosistema.

Paquete	Objetivo general	Funciones o ejemplos clave
purrr	Programación funcional: facilita la aplicación de funciones a listas o marcos de datos.	`map()`, `map_df()`, `map2()`, `pmap()`
haven	Importa y exporta datos de software estadístico como SPSS, Stata o SAS.	`read_sav()`, `read_dta()`, `write_sav()`
readxl	Importa datos desde hojas de cálculo de Excel (`.xls` y `.xlsx`).	`read_excel()`, `excel_sheets()`

Paquetes del Ecosistema Tidyverse (II)

Paquete	Objetivo general	Funciones o ejemplos clave
tibble	Versión moderna de los data frames: impresión legible y manejo seguro de tipos de datos.	`tibble()`, `as_tibble()`
stringr	Manipulación de texto y expresiones regulares de manera consistente.	`str_detect()`, `str_replace()`, `str_split()`, `str_c()`
forcats	Herramientas para manejar factores (variables categóricas).	`fct_reorder()`, `fct_lump()`, `fct_recode()`
lubridate	Simplifica el trabajo con fechas y horas, desde su creación hasta cálculos temporales.	`ymd()`, `dmy()`, `now()`, `interval()`

Paquetes del Ecosistema Tidyverse (II)

Paquete	Objetivo general	Funciones o ejemplos clave
modelr	Integra la modelación dentro del flujo de trabajo tidyverse.	`model_matrix()`, `add_predictions()`, `add_residuals()`
broom	Convierte resultados de modelos estadísticos en data frames ordenados para análisis y visualización.	`tidy()`, `augment()`, `glance()`
dbplyr	Permite trabajar con bases de datos usando sintaxis `dplyr`; traduce el código R a SQL.	`tbl()`, `collect()`, `show_query()`
hms	Maneja datos de tiempo (horas, minutos, segundos) sin fechas.	`hms()`, `as_hms()`
rlang	Proporciona herramientas para programación avanzada y manipulación del lenguaje R.	`sym()`, `quo()`, `eval_tidy()`

¿Qué es un `data.frame` y qué es un `tibble`?

- **`data.frame`**
  Es la estructura de datos base de R para manejar tablas.
  Permite almacenar columnas de diferentes tipos (numéricas, texto, factores, etc.).
  Sin embargo, puede tener comportamientos poco predecibles —por ejemplo, convertir texto en factores o devolver vectores al filtrar columnas.

- **`tibble`**
  Es una versión moderna de `data.frame`, desarrollada dentro del *tidyverse*.

Data Frame vs. Tibble

Característica	`data.frame`	`tibble`
Impresión en consola	Muestra *todo* el contenido, incluso grandes volúmenes de datos.	Muestra solo las primeras filas y ajusta el ancho automáticamente.
Conversión de tipos	Convierte automáticamente las cadenas en factores (si no se especifica `stringsAsFactors = FALSE`).	Nunca convierte tipos de forma automática.
Nombres de variables	Permite nombres no sintácticos (pueden causar errores).	Requiere nombres válidos, aunque puede mantener no sintácticos con comillas invertidas.
Compatibilidad con el tidyverse	Limitada: requiere coerción o conversión a tibble.	Totalmente integrada: se comporta de forma coherente con `dplyr`, `ggplot2`, etc.

Instalación

```
install.packages("tidyverse")
```

El comando carga los paquetes más usados del ecosistema.

```
library(tidyverse)
```

Datos ordenados (*Tidy Data*)

Ejemplo de datos *no ordenados*:

id	ingreso_2023	ingreso_2024
1	500	700

Forma *ordenada*:

id	año	ingreso
1	2023	500
1	2024	700

Ventajas de trabajar con Tidyverse

- Código más claro y reproducible
- Flujo de análisis más directo
- Integración entre importación, manipulación y visualización
- Resultados consistentes y fácilmente documentables

Operador pipe %>%

Permite encadenar funciones de manera secuencial:

```
data %>%
  filter(edad > 25) %>%
  summarise(media = mean(ingreso))
```

Equivale a:

```
summarise(filter(data, edad > 25), media = mean(ingreso))
```

Ejemplo básico

```r
library(dplyr)

mtcars %>%                    # Es una base de datos incluida en R
  select(mpg, cyl, hp) %>%  # Selecciona solo las columnas
  filter(cyl == 6) %>%      # Filtra las filas
  summarise(media_hp = mean(hp)) # Resumen de la variable
```

Uso de `glimpse()`

Muestra la estructura compacta de un data frame:

```
dplyr::glimpse(mtcars)
```

```
Rows: 32
Columns: 11
$ mpg  <dbl> 21.0, 21.0, 22.8, 21.4, 18.7, 18.1, 14.3, 24.4, 22.8, 19.2, 1
$ cyl  <dbl> 6, 6, 4, 6, 8, 6, 8, 4, 4, 6, 6, 8, 8, 8, 8, 8, 8, 4, 4, 4, 4
$ disp <dbl> 160.0, 160.0, 108.0, 258.0, 360.0, 225.0, 360.0, 146.7, 140.8
$ hp   <dbl> 110, 110, 93, 110, 175, 105, 245, 62, 95, 123, 123, 180, 180,
$ drat <dbl> 3.90, 3.90, 3.85, 3.08, 3.15, 2.76, 3.21, 3.69, 3.92, 3.92, 3
$ wt   <dbl> 2.620, 2.875, 2.320, 3.215, 3.440, 3.460, 3.570, 3.190, 3.150
$ qsec <dbl> 16.46, 17.02, 18.61, 19.44, 17.02, 20.22, 15.84, 20.00, 22.90
$ vs   <dbl> 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1
$ am   <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0
$ gear <dbl> 4, 4, 4, 3, 3, 3, 3, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 4, 4, 4, 3
$ carb <dbl> 4, 4, 1, 1, 2, 1, 4, 2, 2, 4, 4, 3, 3, 3, 4, 4, 4, 1, 2, 1, 1
```

Recursos recomendados

- R for Data Science
- Cheat Sheet oficial del Tidyverse
- Documentación: `?dplyr`, `?tidyr`

Conclusión

El **Tidyverse** es más que un conjunto de paquetes: es una forma de pensar el análisis de datos con coherencia, claridad y reproducibilidad.

"El código legible hoy es el análisis replicable mañana." — Hadley Wickham