

Fundamentos de ciencia de datos con R - Módulo 4

Clase 1: Estructura básica de ggplot (grammar of graphics)

CEPAL - Unidad de Estadísticas Sociales

2025-11-03

Introducción

Esta clase introduce la lógica detrás del paquete **ggplot2**, basado en la *Gramática de los Gráficos* propuesta por Leland Wilkinson (2005).

Comprender esta estructura te permitirá crear gráficos complejos a partir de una sintaxis coherente y modular.

¿Qué es *ggplot2*?

- ▶ Es un paquete del ecosistema **tidyverse**.
- ▶ Permite construir gráficos de manera **declarativa**, es decir, describiendo *qué* se quiere mostrar más que *cómo*.
- ▶ Se basa en la **Gramática de los Gráficos (Grammar of Graphics)**:
 - ▶ **Datos** → qué se visualiza
 - ▶ **Estética (aes)** → cómo se asignan las variables a los ejes o atributos
 - ▶ **Geometrías (geom)** → qué tipo de gráfico se usa (puntos, barras, líneas)
 - ▶ **Facetas, escalas, temas** → refinamientos visuales

¿Qué es *ggplot2*?

i Nota

El nombre `ggplot` proviene de “*Grammar of Graphics plot*”.

Cada gráfico se construye por capas (+), combinando datos, estéticas y geometrías.

Estructura básica de un gráfico

```
library(ggplot2)

ggplot(data = mpg, aes(x = displ, y = hwy)) +  
  geom_point()
```

i Nota

Este es el formato mínimo:

1. `ggplot(data, aes(...))` define los datos y las variables.
2. `geom_point()` añade la geometría del gráfico (puntos).
3. El operador `+` agrega capas adicionales.

Componentes de ggplot2

Componente	Función principal
data	Fuente de datos a graficar
aes()	Mapea variables a atributos visuales
geom_	Define el tipo de gráfico (puntos, barras, líneas, etc.)
facet_	Divide los datos en paneles
scale_	Controla colores, tamaños, ejes
theme()	Ajusta elementos visuales del gráfico

Componentes de ggplot2



Tip

Cada gráfico de *ggplot2* es la combinación de estos elementos mediante el operador `+`.

Esa suma no es estética: representa **una adición lógica de significado**.

Bloque 1: Datos (data)

- ▶ Es el **punto de partida** de cualquier gráfico.
- ▶ Deben ser un **data frame** o tibble: filas (observaciones), columnas (variables).
- ▶ ggplot no crea datos: **solo los representa**.

Bloque 2: Estética (aes())

- ▶ Define el **mapeo** entre los datos y los atributos visuales del gráfico.
- ▶ Cada variable se asigna a una propiedad visual (eje, color, tamaño, forma, transparencia).

Atributo visual	Significado habitual
x, y	Posición horizontal y vertical
color, fill	Diferenciación de grupos
size	Magnitud o peso
shape	Tipo de marcador
alpha	Transparencia

Bloque 2: Estética (aes())



Advertencia

No confundir **mapeo** con **asignación fija**:

- `aes(color = grupo)` → el color depende de los datos.
- `color = "blue"` → color fijo, no depende de los datos.

Bloque 3: Geometría (geom_*)

Determina **la forma visual** con la que se mostrarán los datos.



Tip

Antes de elegir la geometría, pregúntate:
“¿Qué tipo de relación quiero mostrar?”

Bloque 3: Geometría (geom_*)

Geometría	Función	Tipo de gráfico
geom_point()	puntos	Dispersión
geom_line()	líneas	Series de tiempo
geom_bar()	barras	Frecuencia o conteo
geom_boxplot()	cajas	Distribución estadística
geom_histogram()	histogramas	Distribución continua
geom_density()	curvas de densidad	Probabilidad
geom_text()	etiquetas de texto	Anotaciones

Bloque 4: Facetas (facet_*)

Permiten **dividir el gráfico** en múltiples paneles según una o más variables.
Ejemplos:

- ▶ `facet_wrap(~sexo)`
- ▶ `facet_grid(edad ~ zona)`

Nota

Facetar ayuda a comparar grupos sin sobrecargar un solo gráfico.

Bloque 5: Escalas (scale_*)

Traducen valores numéricos o categóricos a **representaciones visuales** (colores, tamaños, posiciones).

- Ejemplos:

▶ `scale_x_continuous()`, `scale_y_log10()`

▶ `scale_color_brewer()`, `scale_fill_manual()`

Nota

Nota:

Las escalas también controlan etiquetas, límites y unidades de medida en ejes y leyendas.

Bloque 6: Tema (`theme()`)

Controla **la apariencia general del gráfico**, sin afectar los datos. Elementos ajustables: - Tipografía y tamaño de texto.

- Colores de fondo.

- Cuadrículas, márgenes y leyendas.

Existen temas prediseñados: `theme_minimal()`, `theme_light()`, `theme_bw()`, etc.

La lógica de construcción: capas (+)

Cada gráfico se forma al **superponer capas**:

```
ggplot(data = ...) +  
  aes(...) +  
  geom_...() +  
  scale_...() +  
  theme_...()
```