

# RMarkdown

## Investigación Reproducible

Documentos dinámicos con R

# ¿Qué es RMarkdown?

Un formato que permite combinar código R, resultados y texto narrativo en un único documento



## Reproducible

Todo el análisis es ejecutable y verificable



## Dinámico

Los resultados se actualizan con los datos



## Flexible

Múltiples formatos de salida

# Casos de Uso

- **Reportes de análisis:** Informes automáticos con datos actualizados
- **Documentación de código:** Explicar análisis paso a paso
- **Artículos académicos:** Papers con análisis integrado
- **Presentaciones:** Slides interactivas con código
- **Dashboards:** Reportes interactivos en HTML

 **Ventaja clave:** El análisis y el documento son uno solo, sin copiar y pegar

# Anatomía de un Documento RMarkdown

## 1 Encabezado YAML

Metadatos y configuración del documento

## 2 Texto Markdown

Narrativa, explicaciones, contexto

## 3 Chunks de Código R

Ánalisis, cálculos, visualizaciones

**Estos tres elementos se combinan para crear un documento completo**

# Encabezado YAML

Define metadatos y opciones de renderizado del documento

```
---
```

```
title: "Mi Análisis Económico"
author: "Tu Nombre"
date: "2025-11-10"
output: html_document
---
```

- **title, author, date:** Información básica del documento
- **output:** Formato(s) de salida (HTML, PDF, Word)

# Chunks de Código R

Bloques de código ejecutable que se integran en el documento

```
```{r nombre-chunk, opciones}
# Código R aquí
library(tidyverse)

datos <- read_csv("datos.csv")
summary(datos)
````
```

**Atajo:** Ctrl+Alt+I (Win/Linux) o Cmd+Option+I (Mac)

# Opciones de Chunks: Control de Ejecución

eval = TRUE

Ejecutar:

TRUE = ejecuta

FALSE = no ejecuta

echo = TRUE

Mostrar código:

TRUE = visible

FALSE = oculto

include = TRUE

Incluir salida:

FALSE = ejecuta

pero no muestra

```
```{r, echo=FALSE}
# Código oculto pero ejecutado
resultado <- mean(datos$variable)
```

```

# Opciones de Chunks: Mensajes y Figuras

warning = FALSE

Oculta advertencias

message = FALSE

Oculta mensajes

## Control de Figuras

```
```{r, fig.width=8, fig.height=5}
ggplot(datos, aes(x=var1, y=var2)) +
geom_point()
```
```

**fig.width, fig.height:** Dimensiones en pulgadas

# Configuración Global

Usa `knitr::opts_chunk$set()` para configurar opciones una vez para todo el documento

```
```{r setup, include=FALSE}
knitr::opts_chunk$set(
  echo = FALSE,
  warning = FALSE,
  message = FALSE,
  fig.width = 8,
  fig.height = 5
)
````
```



**Tip:** Coloca este chunk al inicio, después del YAML

# Sintaxis Markdown Básica

## Encabezados

```
# Título 1  
## Título 2  
### Título 3
```

## Énfasis

```
*cursiva*  
**negrita**  
***ambas***
```

## LaTeX

```
$\beta_0$ o $$ formula larga $$
```

## Listas

1. Item 1
2. Item 2
- a. Subitem

## Enlaces

```
[texto](url)
```

## Imágenes

```
![alt](ruta.png)
```

# Tablas con kable()

El paquete **knitr** incluye kable() para tablas formateadas

```
```{r}
library(knitr)

kable(head(mtcars))
```

```
library(kableExtra)
kable(head(mtcars)) |>
  kable_styling()
``
```

**kableExtra** agrega estilos y opciones avanzadas

# Figuras y Gráficos

Los gráficos de ggplot2 (y base R) se insertan automáticamente

```
```{r grafico, fig.cap="Relación"}  
library(ggplot2)  
  
ggplot(datos, aes(x = ingreso,  
y = consumo)) +  
geom_point() +  
theme_minimal()  
```
```

# Formatos de Salida

## HTML

- Interactivo
- Portable
- Gráficos dinámicos
- Fácil distribución

## PDF

- Vía LaTeX
- Aspecto profesional
- Para impresión
- Artículos académicos

## Word

- Edición colaborativa
- Comentarios
- Track changes
- Compatible

# Reportes Automáticos: ¿Por qué?



## Escenario típico:

Necesitas generar el mismo análisis para múltiples unidades: provincias, años, empresas, etc.

### Sin automatizar

- Copiar y pegar manual
- Propenso a errores
- Difícil actualizar
- Tedioso y lento

### Con automatización

- Un solo template
- Loop genera todos
- Actualización fácil
- Rápido y confiable

# Reportes en Loop: Código

```
library(rmarkdown)

provincias <- c("Buenos Aires", "Córdoba")

for (prov in provincias) {
  render(
    input = "template.Rmd",
    output_file = paste0("reporte_", prov,
    ".html"),
    params = list(provincia = prov)
  )
}
```

# Parametrización de Reportes

## En el YAML:

```
---
```

```
title: "Reporte `r params$provincia`"
```

```
output: html_document
```

```
params:
```

```
  provincia: "Buenos Aires"
```

```
  año: 2024
```

```
---
```

Valores por defecto

## En el código:

```
```{r}
```

```
prov <- params$provincia
```

```
año <- params$año
```

  

```
datos_filtrados <- datos |>
```

```
filter(provincia == prov,
```

```
año == año)
```

```
```
```

# Mejores Prácticas

## Organización

- Usa proyectos de RStudio
- Estructura clara de carpetas
- Nombres descriptivos

## Control de Versiones

- Usa Git desde el inicio
- Commits frecuentes
- Mensajes claros

## Nombres de Chunks

- Siempre nombra tus chunks
- Usa guiones, no espacios
- Descriptivos y concisos

## Reproducibilidad

- Documenta dependencias
- Usa rutas relativas
- Comparte el entorno

# Recursos y Cierre

## Recursos Esenciales

- **RMarkdown Cheatsheet:** Guía rápida en 2 páginas
- **RMarkdown: The Definitive Guide** - Yihui Xie et al.
- **R Markdown Cookbook** - Soluciones prácticas
- **bookdown.org:** Para libros y documentos largos
- **Comunidad R:** StackOverflow, RStudio Community

¡Gracias!

¿Preguntas?