

UNIVERSIDAD DE
GUANAJUATO



Programación en R

Reconocimiento de patrones

Dr. Mario Alberto Ibarra Manzano

ibarram@ugto.mx

<https://github.com/ibarram/ReconocimientoPatrones>

Agenda

Qué es R y qué no es

RStudio: interfaz y flujo de trabajo

Objetos, tipos y estructuras de datos

Importar datos, paquetes y el "tidyverse"

Transformación con dplyr y visualización con ggplot2

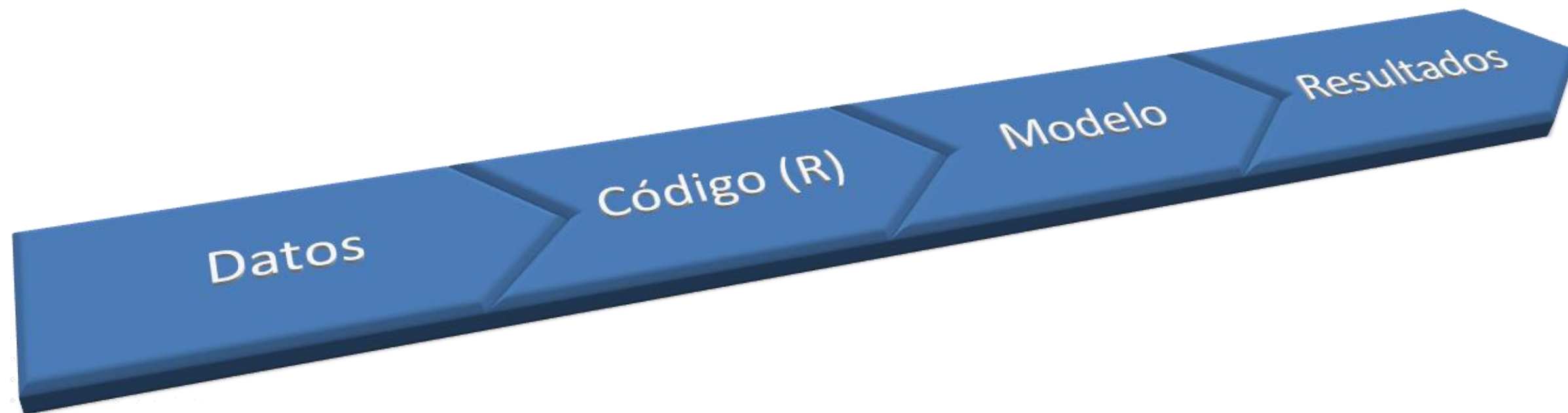
Errores típicos, ayuda y buenas prácticas

¿Qué es R?

Piensa en R como:

- Un lenguaje para manipular datos, programar y modelar
- Un ecosistema de paquetes (CRAN, Bioconductor)
- Una forma reproducible de trabajar: script → resultados → reporte

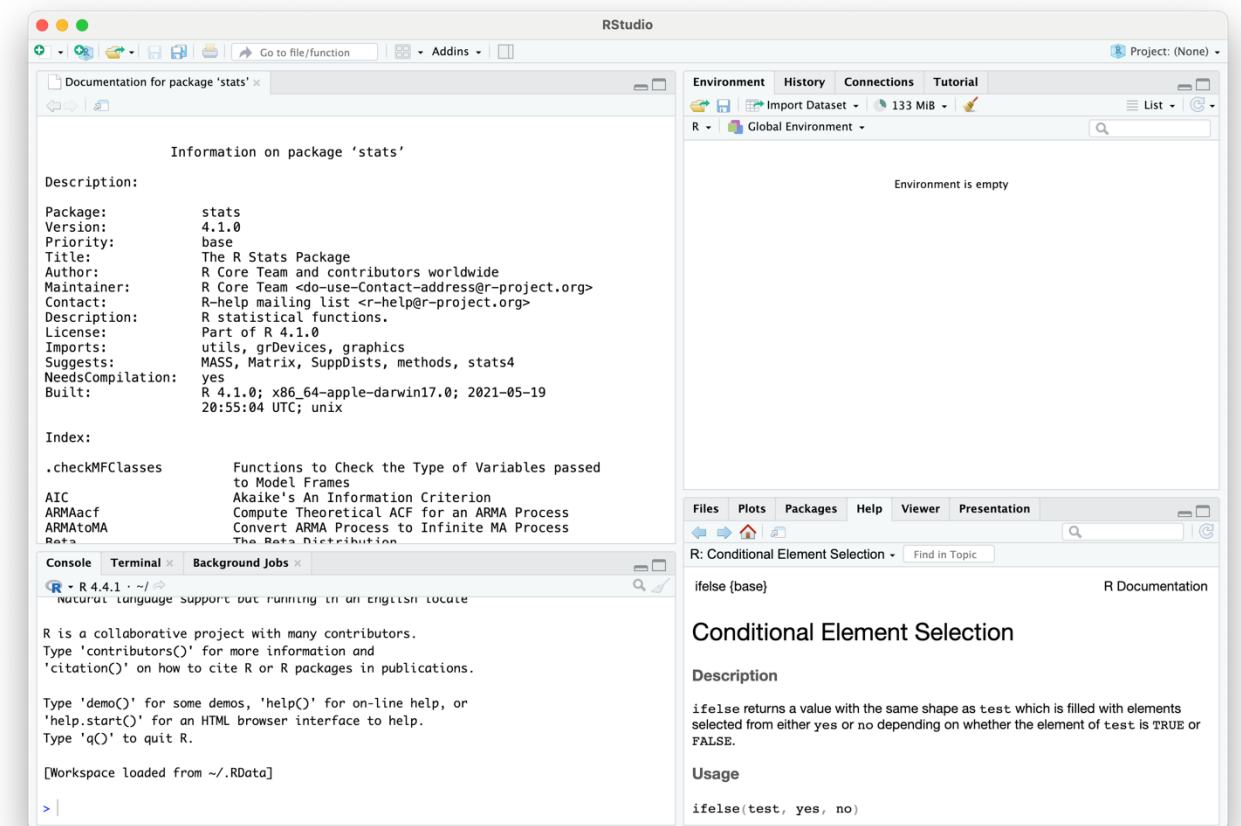
Idea clave: todo lo queda en código → se puede repetir y auditar



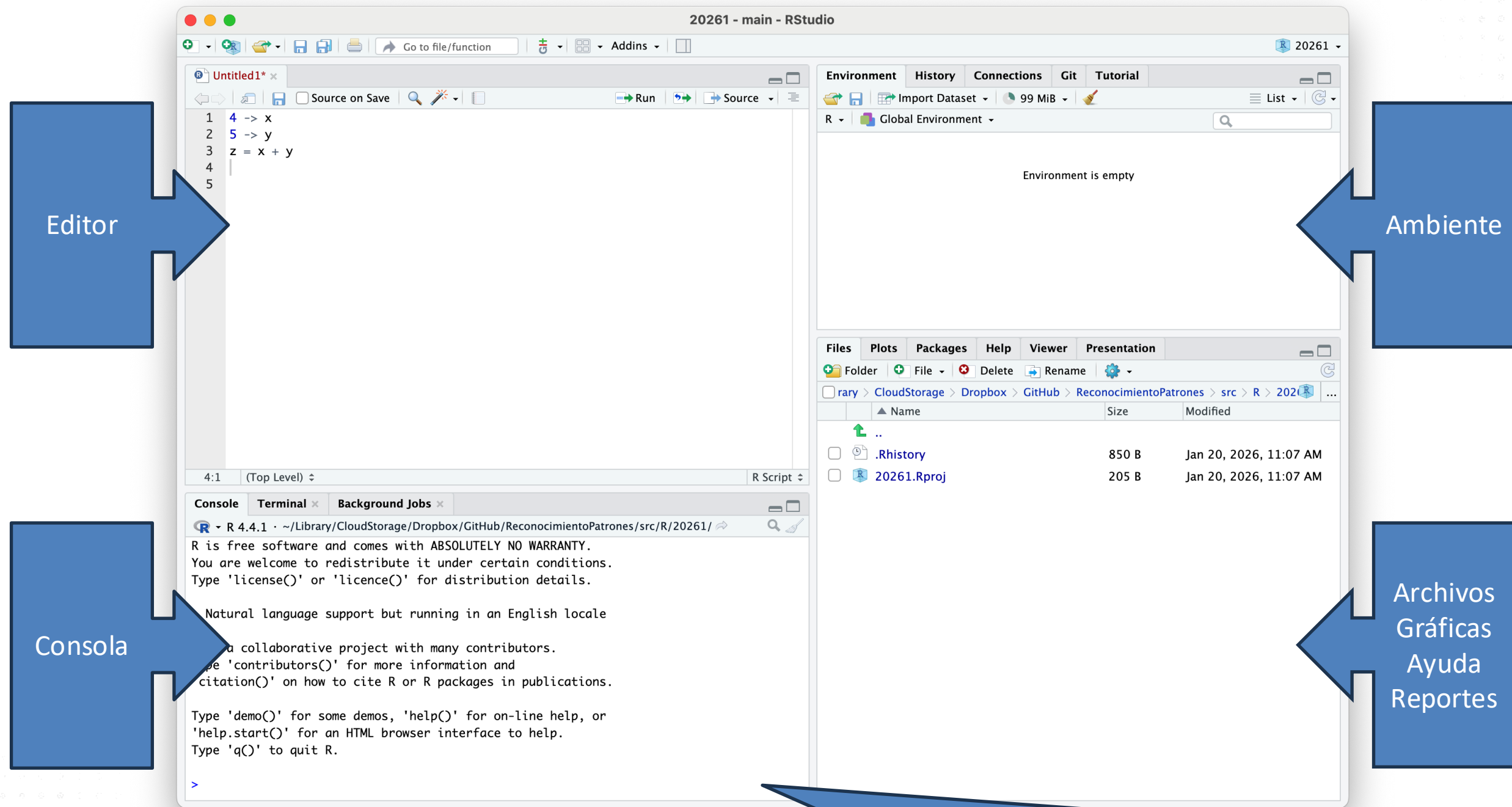
Instalación y herramientas

Proceso de instalación:

1. Instalar R (motor) desde CRAN
<https://www.r-project.org/>
2. Instalar RStudio Desktop (IDE)
<https://posit.co/downloads/>
3. Abrir RStudio Desktop → Crear un "Project" nuevo
Menu → File → New Project → New Directory → Directory name
4. Instalar paquetes: **tidyverse** y **tidymodels**
Tools → Install Packages ... → Packages → tidyverse, tidymodels
5. Crear un nuevo código fuente (R)
File → New File → R Script (⌘+⇧+N)



Interfaz de RStudio (4 paneles básicos)



Todo lo importante va en Source; Console, solo para probar; Environment, para verificar objetos; y el panel inferior derecho, para ver archivos, gráficas y ayuda.

Interfaz de RStudio (4 paneles básicos)

En RStudio (vista por defecto) la interfaz se organiza en cuatro paneles, cada uno con un propósito claro:

- **Editor / Source (arriba–izquierda):**

¿Qué es?: El espacio donde escribes y editas código en archivos (.R, .qmd, .Rmd, .Rproj, etc.).

¿Para qué se usa?: Preparar scripts, funciones, análisis reproducibles (Quarto/RMarkdown), ejecutar líneas o bloques seleccionados y guardar tu trabajo.

- **Console (abajo–izquierda):**

¿Qué es?: La línea de comandos interactiva de R (REPL).

¿Para qué se usa?: Ejecutar instrucciones "al vuelo", probar funciones rápidamente, ver mensajes/errores inmediatos y correr comandos que no necesariamente quieres guardar en un script.

- **Environment / History (arriba–derecha):**

¿Qué es?: Un panel de "estado" de tu sesión de R.

¿Para qué se usa?:

Environment: Ver los objetos creados (dataframes, modelos, variables), su tipo y tamaño; cargar/guardar objetos (.RData, .Rds) de forma guiada.

History: Revisar el historial de comandos ejecutados para reutilizarlos.

- **Files / Plots / Packages / Help / Viewer (abajo–derecha):**

¿Qué es?: Un panel multipestaña para trabajar con archivos y resultados.

¿Para qué se usa?:

Files: Navegar por carpetas del proyecto, abrir y renombrar archivos.

Plots: Visualizar y exportar gráficas.

Packages: Instalar o cargar paquetes y consultar su documentación.

Help: Consultar la ayuda sobre funciones y paquetes.

Viewer: Mostrar contenido renderizado (p. ej., HTML de Quarto, Shiny, widgets).

Tipos de datos (atómicos)

Tipo (R)	Ejemplo (R)	¿Qué representa?	Alcance / uso típico	Operaciones comunes	Notas importantes
logical	c(TRUE,FALSE)	Booleanos	Filtros, máscaras, reglas, condiciones	&, `	, !, if, which()
integer	10L, 1:5	Enteros	Conteos, índices, IDs discretos	+ - *, seq(), length()	Distingue 10 (double) vs 10L(integer)
double / numeric	3.14	Reales (punto flotante)	Cálculo numérico y estadística	mean(), sd(), %*%, lm()	Puede haber errores de redondeo (0.1+0.2 != 0.3)
complex	1+2i	Complejos	FFT, señales, álgebra compleja	Re(), Im(), Mod(), Arg()	Menos común en tabular clásico, más en DSP
character	"texto"	Texto	Etiquetas, categorías crudas, parsing, rutas	paste(), sprintf(), nchar(), grepl()	Para “categorías”, suele convertirse a factor o one-hot
raw	charToRaw("Hi")	Bytes	IO binario, hashes, protocolos	rawToChar(), lectura/escritura binaria	No se usa para cálculos; es para datos binarios
factor	factor(c("A","B"))	Categorico con niveles	Variables cualitativas, clase objetivo en clasificación	levels(), relevel(), table()	Los niveles importan; cuidado al predecir con niveles nuevos
Date	as.Date("2026-01-20")	Fecha (día)	Series por día, calendarios, cortes temporales	+/- días, difftime()	No tiene hora; para hora usa POSIXct
POSIXct	as.POSIXct("...")	Fecha-hora	Timestamps, sensores, logs	format(), difftime(), lubridate	Siempre considera tz (zona horaria)

Tipos de datos

```
# logical
flag <- TRUE
x <- c(TRUE, FALSE, TRUE)
```

```
# integer
n <- 10L
idx <- 1:5
```

```
# double/numeric
a <- 3.14159
v <- c(0.1, 2.5, -7)
```

```
# complex
z <- 3+5i
Conj(z)
z1 <- complex(real=3, imaginary=-4)
Re(z1)
Im(z1)
z2 <- complex(modulus = 2, argument = pi/3)
Mod(z2)
Arg(z2)
```

```
# character
a <- "Reconocimiento de Patrones"
names <- c("Ana", "María", "Luis")
paste(names[1], names[2])
sprintf("acc=%.3f", 0.92567)
nchar(a)
```

```
sort(names)
order(names)
substr(a, 5, 8)
strsplit(x, "e")
grep("a", names)
grepl("a", names)
gregexpr("a", names)
toupper(a)
tolower(a)
unique(c("A", "B", "A"))
table(c("A", "B", "A"))
```

```
# factor
factor(x, levels = ..., labels = ..., ordered = FALSE)
x <- c("A", "B", "A", "C")
f <- factor(x)
is.factor(f)
levels(f)
nlevels(f)
str(f)
f2 <- relevel(f, ref = "B")
f_ord <- factor(c("bajo", "alto", "medio"), levels =
c("bajo", "medio", "alto"), ordered = TRUE)
table(f)
summary(f)
```


Ejemplo práctico

```
datos <- read_csv("data/")
```

TÍTULO