

R para contextos humanitarios de emergencia

Introducción

Violeta Roizman

Hola!

- Me llamo **Violeta**
- Soy **Matemática e Instructora Certificada por RStudio**
- Me gusta **visualizar datos, visitar museos, tomar té, etc, etc.**



Objetivos del taller

- Entender qué son y para qué sirven R y RStudio
- Saber buscar ayuda en el mundo R
- Poder hacer un análisis de datos sencillo y completo, todo en R
- Poder aprovechar R para introducirlo en su flujo de trabajo habitual

Modalidad del taller

- Presentación de los temas con ejemplos
- Ejercicios intercalados para interactuar
- "Live coding"

Requisitos

- Asumimos que ya tienen instalado R y RStudio
- En caso contrario: seguir los pasos en este video

Programa tentativo

	Día 1 (25/11)	Día 2 (26/11)	Día 3 (27/11)
09:00-10:00	Intro al curso	Visualizacion de datos	
10:00-10:05	Pausa	Pausa	
10:05-10:55	Intro a R y RStudio	Visualizacion + Manipulacion	<ul style="list-style-type: none">- Paquete koboloader- QualMiner: Análisis de texto- Tratamiento y procesamiento de datos - 5W
10:55-11:10	Pausa	Pausa	
11:10-11:40	RStudio y proyectos	Mas manipulacion y funciones	
11:40-11:45	Pausa	Pausa	
11:45-13:00	RMarkdown	Interactuar con Kobo y Power BI	
13:00-14:00	Almuerzo	Almuerzo	
14:00-15:00	Lectura de datos	Gráficos, reportes y tablas	
15:00-16:00	Manipulacion de datos	Gráficos, reportes y tablas	

Desafíos de hoy

- Leer correctamente una base de datos
- Generar nuevas columnas a partir de información de otras columnas y generar resúmenes a partir de los datos
- Generar gráficos a partir de la base de datos usada

Qué herramientas para análisis de datos utilizan?

- Excel
- GIS
- Power BI
- Word

Ventajas

- Intuitivos para usar

Desventajas

- Son muy específicos para cada tarea

Si ya tenemos:

- Excel
- Kobo
- GIS
- Power BI
- Word
- InDesign
- Ilustrator

Entonces por qué R?

Porque con R podemos:

- Realizar la mayoría de esas tareas
- Automatizar procesos
- Interactuar fácilmente con muchas herramientas

Y además... es abierto y gratuito

Desventaja



Algunas cosas que podemos hacer en R

- Leer datos
- Transformar y resumir datos
- Visualizar datos
- Generar reportes automáticos
- Crear sitios web. p. ej.: <https://violetr.netlify.app/>
- Desarrollar aplicaciones interactivas:
 - <https://shiny.rstudio.com/gallery/india-blood-banks.html>
 - https://violetr.shinyapps.io/hiv_hnp/
 - https://vac-lshtm.shinyapps.io/ncov_tracker

R en contextos humanitarios de emergencia

- Hay un grupo de usuarios y desarrolladores de R que trabajan en esta área:
HumanitaRian useR group
 - Creado en una reunión de la UNHCR
 - Blog y grupo de Skype
- Documentación y Reproducibilidad de los procesos
- Hay funciones específicamente hechas para manejar este tipo de datos y conectarse con herramientas populares como Kobo
- Existe manuales de buenas prácticas sobre como crear reportes. Por ejemplo,
How to quickly produce statistical reports? The UNHCR “R-Cookbook”

Material del curso

El material del taller está disponible en

- url

Licencia: [Creative Commons Attribution-ShareAlike 4.0 International License](#).

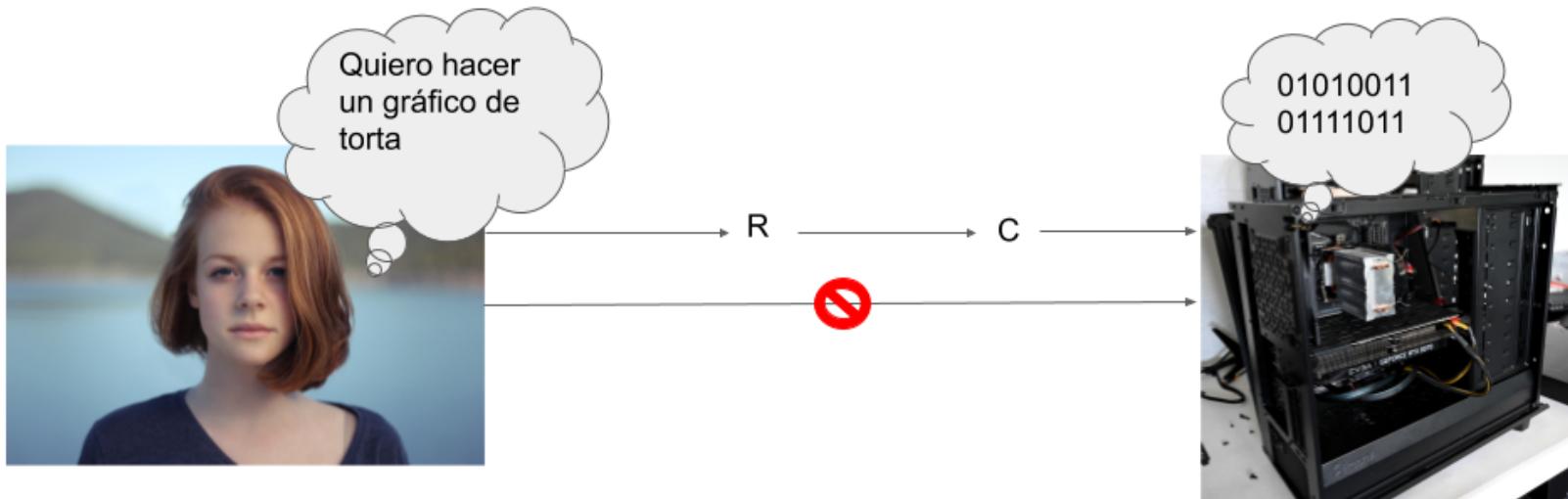
Este material está inspirado en:

- [R para Clima](#) de Paola Corrales y Elio Campitelli
- [Master the Tidyverse](#) de Garrett Grolemund

Empezamos!

Qué es R? Qué es RStudio?

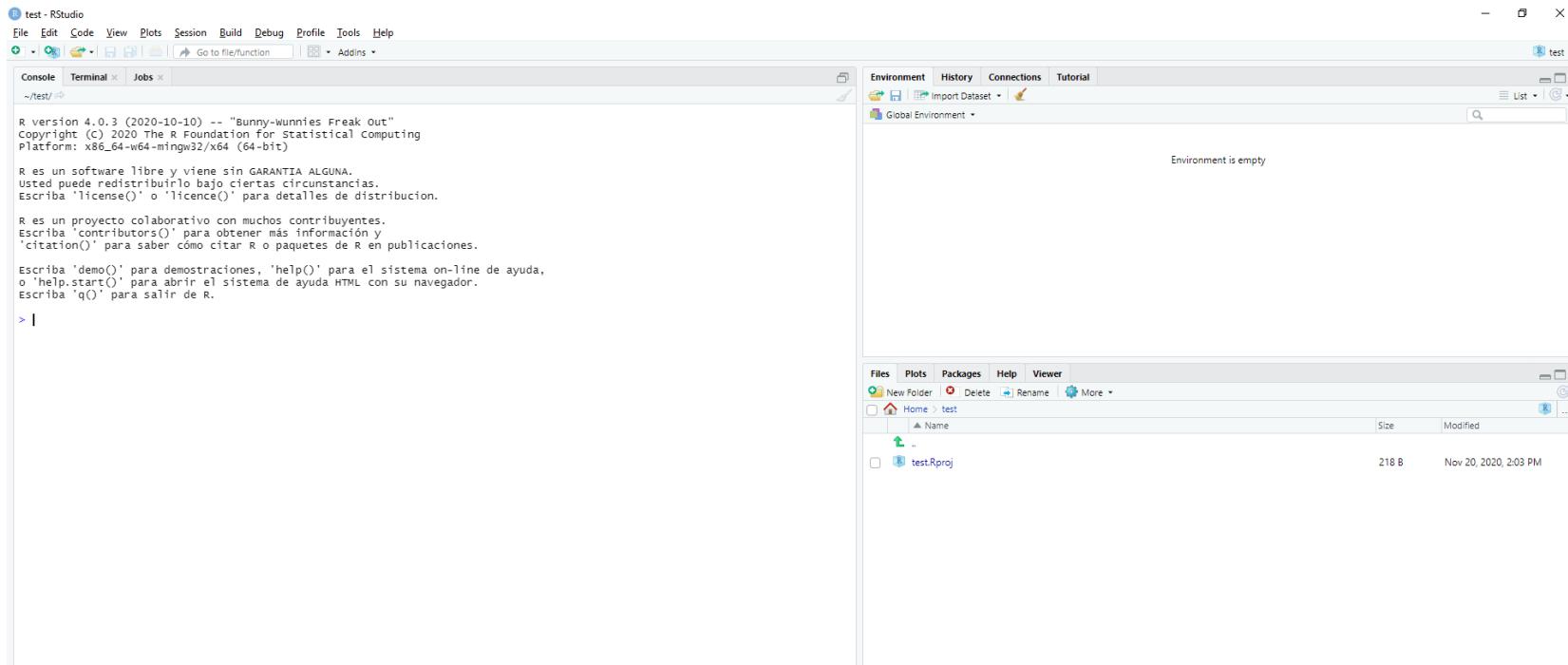
- R es el lenguaje de programación, es la forma de darle instrucciones a la computadora
- RStudio es la interfaz gráfica (lo que vemos)



RStudio: Como se ve la primera vez

A la izquierda, en la consola, podemos escribir código en R

Este se ejecutará al apretar Enter

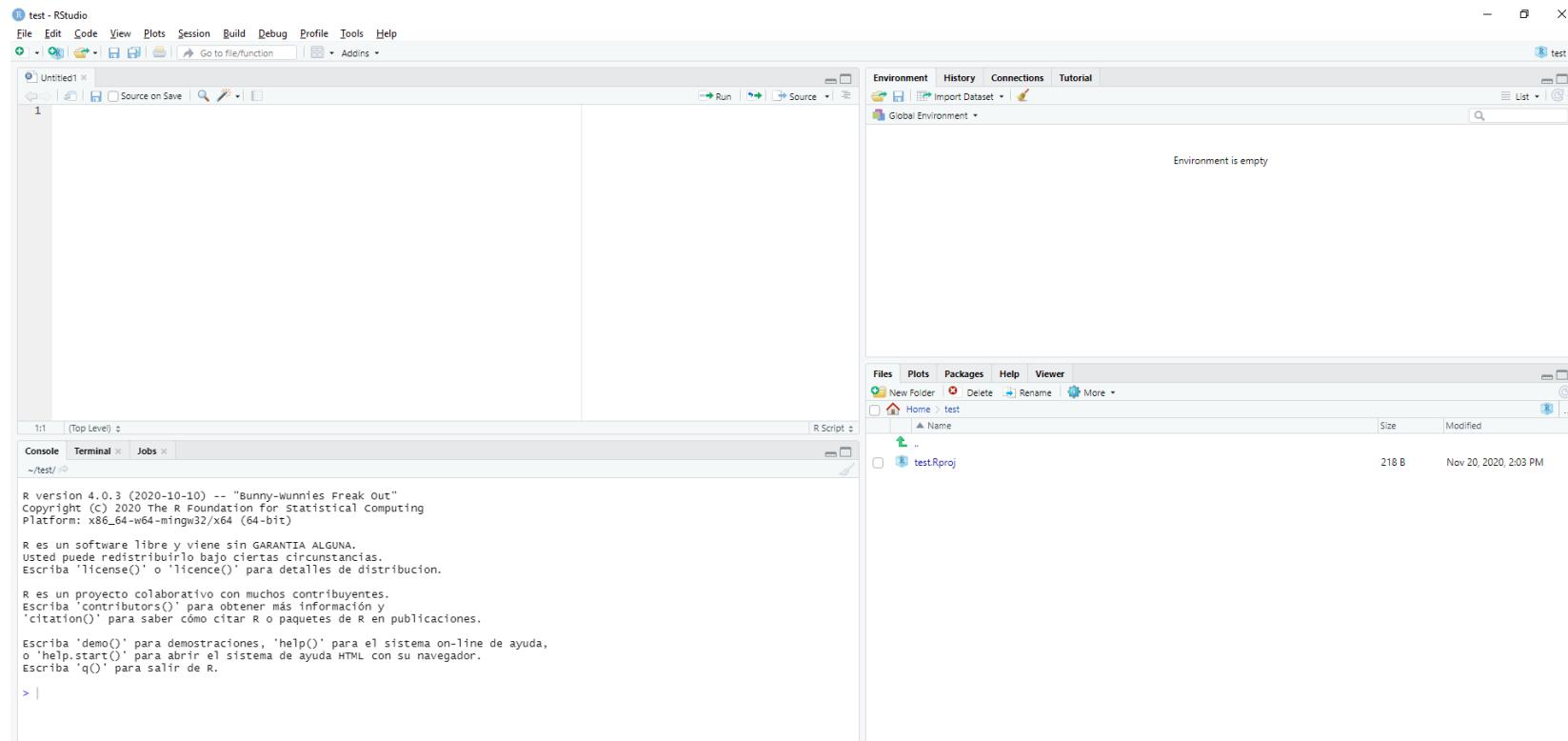


RStudio: Como se ve habitualmente

Vamos a hacer cosas interesantes, y vamos a querer guardarlas

Por eso la configuración normal de RStudio será así, con cuatro paneles

El de arriba a la izquierda es un bloc de notas



RStudio culinario

The screenshot shows the RStudio interface with several cooking-related metaphors overlaid:

- Recetas**: A script icon is shown above the code editor, which contains text about saving code for reuse.
- mesada**: A kitchen counter icon is shown above the Environment pane, which lists ingredients (data) and results (tortas).
- El horno**: An oven icon is shown above the Console pane, which displays R code execution.
- Los paquetes son utensilios o herramientas**: A fork and knife icon is shown above the Packages pane, which lists available packages.
- Files es la alacena donde se guardan los ingredientes y scripts**: A cabinet icon is shown above the Files pane, which shows the file structure.

Adaptado de @JKRward

Cómo le damos instrucciones a R?

Como ya vieron en el video para instalar R, podemos pedirle a R que compute operaciones básicas. Y también no tan básicas.

```
2 + 2
```

```
## [1] 4
```

```
sqrt(2)
```

```
## [1] 1.414214
```

Esto solo nos devuelve el valor

Pero si queremos guardarlo en algún lugar (para que quede disponible en la mesada):

```
raiz_dos <- sqrt(2)  
raiz_dos
```

```
## [1] 1.414214
```

Variables en R

```
raiz_dos <- sqrt(2)  
raiz_dos
```

```
## [1] 1.414214
```

En Excel:

Es parecido a hacer =SQRT(2) (o =RCUAD(2)) en la celda A2

Sabemos que ese resultado está en la celda A2

En R:

Ese resultado está en la variable `raiz_dos` que aparece en el environment (la mesada)

Pero... La analogía no es 100% util:

https://PollEv.com/multiple_choice_polls/P2xOvcUGbMWv91CLwX6o1/respond

```
x <- 2  
y <- x + 2  
x <- 3
```

Vectores en R

- Es la unidad básica en R
- Casi todo es un vector (por eso vemos el [1] a la izquierda en los resultados)
- Todos los elementos de un vector son de un mismo tipo (numérico, carácter, etc)

```
vector1 <- c(3, 5, 6.1, 7)  
vector1
```

```
## [1] 3.0 5.0 6.1 7.0
```

```
vector2 <- 1:20  
vector2
```

```
## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

Puedo calcular el promedio de los elementos de un vector

```
mean(vector2)
```

```
## [1] 10.5
```

Vectores en R

```
vector1 <- c(3, 5, 6.1, 7)  
vector1
```

```
## [1] 3.0 5.0 6.1 7.0
```

Puedo acceder al elemento que está en el lugar 2 del vector

```
vector1[2]
```

```
## [1] 5
```

Puedo obtener todos los valores de un vector salvo el elemento del lugar 2 así:

```
vector1[-2]
```

```
## [1] 3.0 6.1 7.0
```

Listas en R

Qué pasa si quiero guardar más de un tipo de elemento en una variable?

Listas! se crean con `list()` y pueden contener cualquier tipo de datos, incluso listas

```
mi_agenda <- list(Carlos = c(tel = 41232342),
                     Maria = c(tel = 25362819, edad = 25))
mi_agenda$Carlos
```

```
##      tel
## 41232342
```

```
mi_agenda[[2]]
```

```
##      tel      edad
## 25362819        25
```

Funciones en R

R tiene funciones útiles incorporadas directamente

- | `sqrt`, `+`, `-`, `mean`, `sum`, etc

Se aplican a una o más variables y devuelven un valor

Las entradas de las funciones (puede ser 1 o más) se llaman parámetros

Pueden ser obligatorios u opcionales

- | `sqrt` se aplica a un número positivo y devuelve su raíz cuadrada

Pero... nuestro objetivo es **realizar análisis de datos**

☆ **necesitamos más funciones!** ☆

Tenemos acceso a más funciones que las básicas a partir de **paquetes** 🎁

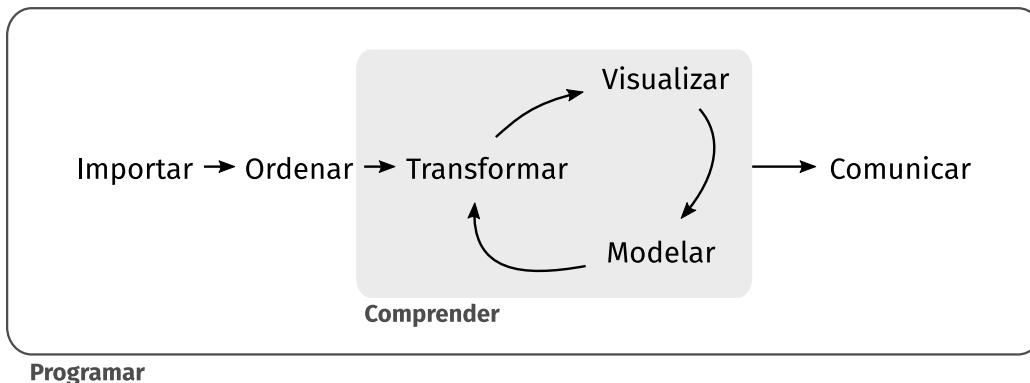
Hay muuuuchos paquetes de R

- De distintas áreas del conocimiento:
 - Genética,
 - Ciencia de datos en general,
 - R humanitario, etc.
- Podemos hacer nuestro propio paquete para nuestro flujo de trabajo



Ciencia de datos

Flujo de trabajo:



Existe un **conjunto de paquetes** para realizar estas tareas



tidyverse

- tidy = ordenado
- ...verse = universo



R packages for data science

The tidyverse is an opinionated **collection of R packages** designed for data science. All packages share an underlying design philosophy, grammar, and data structures.

Install the complete tidyverse with:

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

Instalar paquetes en R

Para instalar paquetes, usamos la función `install.packages()`

```
install.packages("readr")
```

instala un paquete que contiene funciones para leer datos.

Usando el comando

```
library(readr)
```

Las funciones del paquete `readr` estarán disponibles. p. ej.: `read_csv`

Tu turno: Instalar el paquete `readr`

Tu turno Bis: Instalar `tidyverse`

Pedir ayuda en R

Hay muchas formas de pedir ayuda en R

Una opción es usar la **ayuda de R**

```
?nombre_funcion
```

- **Description:** breve descripción de la función
- **Usage**
- **Arguments:** argumentos de la función
- **Details**
- **Value:** lo que devuelve la función
- **Examples**

Tu turno: Abre y lee la ayuda de la función `sd()`. Puede que haya cosas que aún no entiendas, pero trata de captar la idea general. ¿Qué hace esa función? ¿Qué argumentos acepta?

Pedir ayuda en R

Otra opción es **preguntar a instructor@ o colegas**

Otra, **googlear**

Fórmula "óptima" (para mi) para googlear:

| R + pregunta o error descriptivo y específico

o, si estamos usando tidyverse

| tidyverse + pregunta o error descriptivo y específico

Es mejor buscar en inglés (hay más material)

Algunas buenas fuentes:

- stackoverflow (Preguntas y respuestas)
- rstudio community (Foro)

Material del curso

El material del taller está disponible en

- url

Licencia y materiales

Licencia: [Creative Commons Attribution-ShareAlike 4.0 International License](#).

Este material está inspirado y utiliza explicaciones de:

- [R para Clima](#) de Paola Corrales y Elio Campitelli
- [Master the Tidyverse](#) de Garrett Grolemund