

Interculturales

Introducción a técnicas de ciencias sociales computacionales

Centro de Estudios Interculturales e Indígenas
Sesión 1 - 2025

Bienvenidos al taller metodológico!

Esperamos que este espacio sea nutritivo para su aprendizaje y que sea un lugar en el cuál podamos discutir sobre las implicancias del uso de técnicas computacionales para la investigación social.

El taller se estructura en dos partes: Presentación, con una introducción a los conceptos fundamentales de la sesión, y un código práctico en el cuál aplicaremos lo aprendido.

Sobre mí



Matías Deneken

Sociólogo

Asistente de Investigación en el
área cuantitativa

Sociólogo, Magíster en Sociología y candidato a Magíster en Data Science. Actualmente desempeña labores en investigación social aplicada en el Centro CIIR y OLES, con temáticas vinculadas a interculturalidad y violencia. Dentro de sus méritos académicos fue Premio Universidad de Concepción, Becario ANID - Magíster Nacional y es beneficiario del programa Fulbright para iniciar un PhD en Estados Unidos.

Intereses

- Relaciones Intergrupales
- Historia de la Sociología Chilena
- Ciencias Sociales Computacionales
- Text as data

Educación

-  **Sociólogo**
Universidad de Concepción
-  **Magíster en Sociología**
Pontificia Universidad Católica de Chile
-  **Magíster en Data Science**
Universidad de Concepción

Temario de hoy

1. ¿Qué son las ciencias sociales computacionales?
2. Introducción a Tidyverse
3. Funciones fundamentales de Tidyverse
4. Aplicación a datos reales
5. Comunicación de resultados
6. Código práctico

GESIS Fall Seminar in Computational Social Science 2025

register now



Date: 01.09 - 26.09.2025 ics-file

Location: Mannheim, B6 4-5, Online via Zoom

Week 1

[Introduction to Computational Social Science with Python](#)

01.09.2025 - 05.09.2025 - John McLevey

[Introduction to Computational Social Science with R](#)

01.09.2025 - 05.09.2025 - Johannes Gruber

Week 2

[Web Data Collection with Python](#)

08.09.2025 - 12.09.2025 - Iulia Cloroianu

[Web Data Collection with R](#)

08.09.2025 - 12.09.2025 - Iulia Cloroianu

Week 3

[Introduction to Machine Learning for Text Analysis with Python](#)

15.09.2025 - 19.09.2025 - Rupert Kiddle

[Computer Vision for Image and Video Data Analysis](#)

15.09.2025 - 19.09.2025 - Andreu Casas

[Advanced Methods for Social Network Analysis](#)

15.09.2025 - 19.09.2025 - Lorien Jasny, Laura Roldan Gomez (Teaching Assistant)

Week 4

[From Embeddings to LLMs: Advanced Text Analysis with Python](#)

22.09.2025 - 26.09.2025 - Hauke Licht

[Agent-Based Computational Modeling](#)

22.09.2025 - 26.09.2025 - Daniel Mayerhoffer

[Causal Machine Learning](#)

22.09.2025 - 26.09.2025 - Marica Valente

Ciencias Sociales Computacionales

Explosión de datos sociales digitales

- Redes sociales, sensores, plataformas digitales
- Nuevas fuentes de datos generadas en tiempo real

Avances tecnológicos

- Aumento del poder computacional
- Acceso a técnicas como machine learning y visualización avanzada

Limitaciones de los métodos tradicionales

- Crece la necesidad de *escalar* el análisis sin perder profundidad teórica

Transformación del mundo social

- Lo digital ya no es un “medio”, sino parte del tejido social.
- La vida social deja huellas de datos

“¿Está el K-Pop detrás de la protesta chilena?": Medios coreanos reaccionaron al cuestionado informe de Big Data

Ciencias Sociales Computacionales



¿Se puede ser crítico?

¿Los datos son neutrales o reflejan relaciones de poder? ¿Qué queda fuera cuando modelamos o cuantificamos fenómenos sociales?



Crítica en CSC implica:

Cuestionar quién produce los datos y con qué fines
Reflexionar sobre los modelos, métricas y sesgos
Proponer formas más justas de representar la realidad



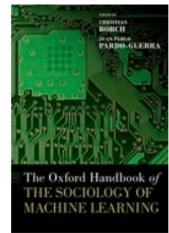
Corrientes críticas

👉 Feminismo de datos: ¿A quién se nombra, a quién se invisibiliza? → Data Feminism (D'Ignazio & Klein).



👉 Decolonización de los datos: → ¿Qué epistemologías quedan fuera del análisis computacional? → Datos como forma de poder colonial.

Ciencias Sociales Computacionales



The Oxford Handbook of the Sociology of Machine Learning

Christian Borch (ed.), Juan Pablo Pardo-Guerra (ed.)

<https://doi.org/10.1093/oxfordhb/9780197653609.001.0001>

Published online: 20 November 2023 Published in print: 18 June 2025

9780197653630

Print ISBN: 9780197653609

Online ISBN:

Search in this book

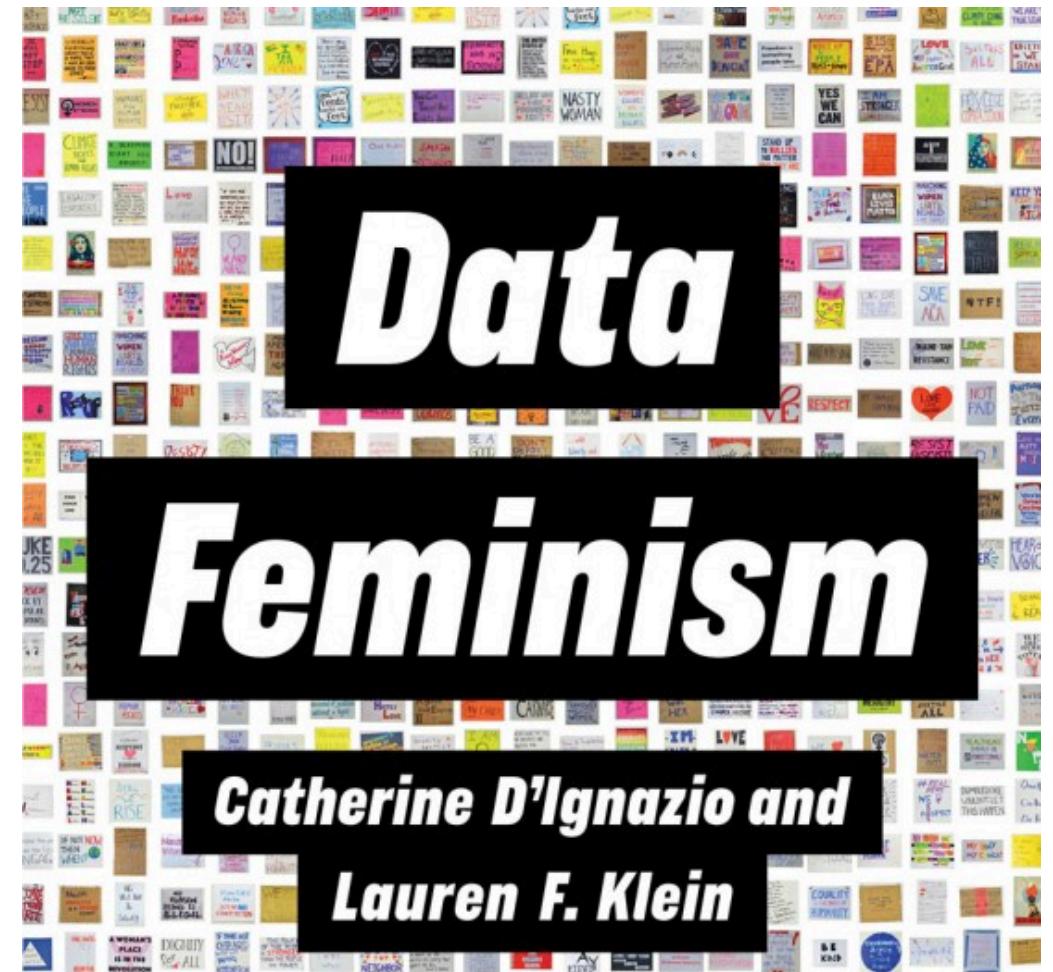
CHAPTER

6 Chinese Computational Sociology: Decolonizing Applications of Machine Learning and Natural Language Processing Methods in Chinese-Language Contexts

Linda Hong Cheng, Yao Lu

<https://doi.org/10.1093/oxfordhb/9780197653609.013.2> Pages 99–120

Published: 22 May 2024



Introducción: R Base y Tidyverse

 **R Base** Es el conjunto de funciones que vienen integradas por defecto en R. Permiten hacer casi todo (leer datos, analizarlos, graficarlos), pero con una sintaxis más técnica, menos legible y a veces más compleja para tareas comunes

 **Tidyverse**

Es un ecosistema moderno de paquetes construidos sobre R base, diseñado para que trabajar con datos sea más intuitivo, legible y coherente.

 En resumen

R base es el “idioma nativo” de R. Tidyverse es una forma moderna y fluida de hablar ese idioma.

Ejecución de funciones

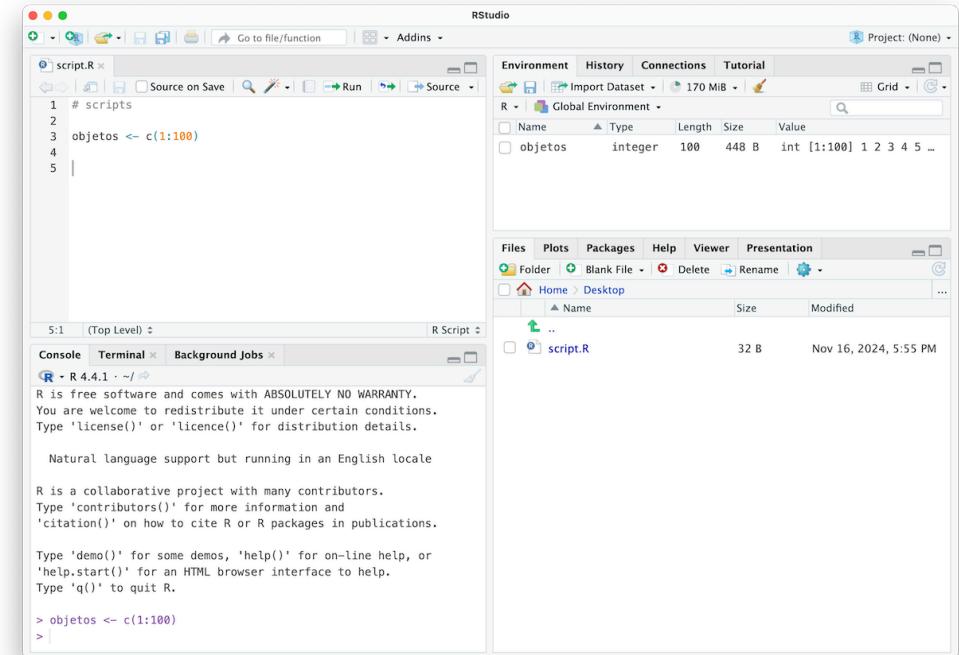
```
library(tidyverse)
```

Sobre R

Al trabajar en R, realizamos nuestros análisis y pruebas mediante scripts. Los scripts son archivos de texto que terminan en .R. En estos archivos de texto escribimos nuestro código.

La consola es la forma directa de interactuar con el lenguaje R. En ella, las instrucciones se escriben y se presiona enter para enviar los comandos.

Si escribimos nuestros comandos en scripts, los comandos van a quedar guardados. Si una línea contiene un signo gato #, todo lo que esté después del gato se transformará un comentario



Sobre R

R es un lenguaje de programación diseñado especialmente para el análisis de datos, la estadística y la visualización. Con R puedes realizar desde operaciones matemáticas básicas, como si fuera una calculadora, hasta tareas mucho más complejas como definir funciones propias, manipular grandes bases de datos, crear gráficos personalizados, aplicar modelos estadísticos, realizar análisis de texto.

Gracias a su comunidad, R cuenta con miles de paquetes que permiten extender sus funcionalidades en áreas como machine learning y análisis espacial.



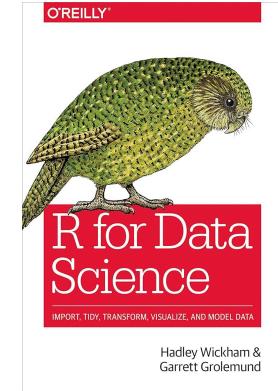
Introducción: El Ecosistema Tidyverse



Hadley Wickham



Tidyverse



R for Data Science

Esto no es una pipa!

¿Qué es la "pipa" en tidyverse? La pipa
%>% permite *encadenar operaciones* de forma clara y legible.

```
penguins %>%  
  group_by(species) %>%  
  summarise(avg_bill_length =  
            mean(bill_length_mm,  
                  na.rm = TRUE))
```

```
## # A tibble: 3 × 2  
##   species     avg_bill_length  
##   <fct>           <dbl>  
## 1 Adelie        38.8  
## 2 Chinstrap     48.8  
## 3 Gentoo        47.5
```

🎨 **¿Por qué se usa la imagen de Magritte para explicar la pipa?**

El operador %>% se pronuncia comúnmente como pipe (tubería en inglés), y su símbolo recuerda una tubería por donde fluye algo.

Juego conceptual: La obra de Magritte muestra una imagen de una pipa con el texto "Esto no es una pipa".

De modo similar, el operador %>% no es literalmente una pipa.

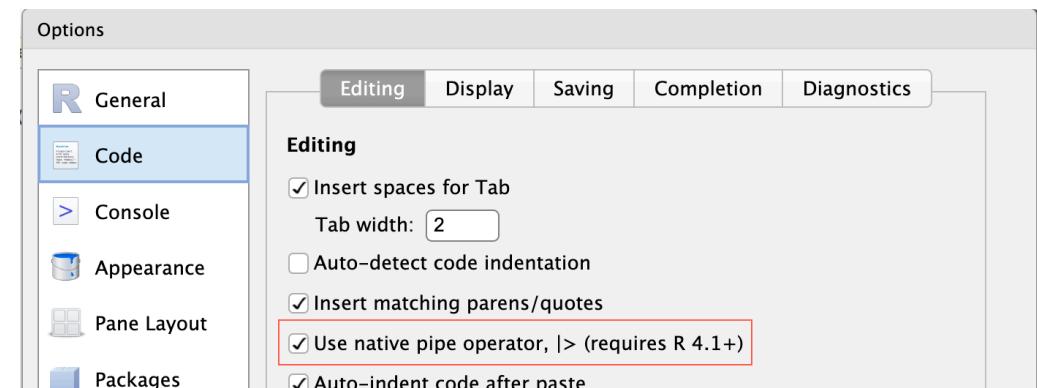
Esto no es una pipa!



La pipa se ejecuta con **Command/Control + Shift + M**.

Dos formas: Funcionan igual.

- `%>%`
- `|>`



Tidyverse: Ejemplos con data Penguins



Funciones del R Base

R es un software basado en objetos. Un objeto es una estructura que almacena información en la memoria.

```
pepino <- 7
manzana <- 4

pepino * manzana #multiplicación
```

```
## [1] 28
```

```
pepino/manzana #división
```

```
## [1] 1.75
```

Funciones del R Base

```
# Crear vector de frutas (caracteres)
frutas <- c("Manzana", "Banana", "Pera", "Naranja", "Uva")
# Crear vector con longitudes en centímetros (números)
longitud_cm <- c(8.5, 20.0, 10.0, 9.0, 2.5)

# Combinar en una tabla usando data.frame
tabla_frutas <- data.frame(
  Fruta = frutas,
  Longitud_cm = longitud_cm
)

# Ver tabla
print(tabla_frutas)
```

```
##          Fruta Longitud_cm
## 1      Manzana        8.5
## 2     Banana       20.0
## 3       Pera       10.0
## 4   Naranja        9.0
```

Funciones del R Base

```
summary(penguins)
```

```
##           species      island   bill_length_mm   bill_depth_mm
## Adelie     :152    Biscoe     :168      Min.    :32.10      Min.    :13.10
## Chinstrap: 68    Dream      :124     1st Qu.:39.23     1st Qu.:15.60
## Gentoo    :124   Torgersen: 52    Median  :44.45    Median  :17.30
##                               Mean    :43.92    Mean    :17.15
##                               3rd Qu.:48.50    3rd Qu.:18.70
##                               Max.    :59.60    Max.    :21.50
##                               NA's    :2       NA's    :2
##   flipper_length_mm   body_mass_g      sex          year
##   Min.    :172.0      Min.    :2700   female:165      Min.    :2007
##   1st Qu.:190.0      1st Qu.:3550   male  :168      1st Qu.:2007
##   Median  :197.0      Median :4050   NA's   : 11      Median  :2008
##   Mean    :200.9      Mean    :4202                    Mean    :2008
##   3rd Qu.:213.0      3rd Qu.:4750                    3rd Qu.:2009
##   Max.    :231.0      Max.    :6300                    Max.    :2009
##   NA's    :2          NA's    :2
```

Tidyverse: `select()` Extrae columnas específicas del dataset

```
penguin |> glimpse()
```

```
## # A tibble: 344 × 2
##   species     island
##   <fct>       <fct>
## 1 Adelie      Torgersen
## 2 Adelie      Torgersen
## 3 Adelie      Torgersen
## 4 Adelie      Torgersen
## 5 Adelie      Torgersen
## # ... with 339 more rows
```

```
penguin %>%
  select(species) |>
  head(5)
```

```
## # A tibble: 5 × 1
##   species
##   <fct>
## 1 Adelie
## 2 Adelie
```

Tidyverse: `mutate()` : Crea nuevas columnas o transforma existentes

```
penguins %>%  
  mutate(bmi = body_mass_g / flipper_length_mm) |>  
  select(species, bmi, body_mass_g, flipper_length_mm) |>  
  head(2)
```

```
## # A tibble: 2 × 4  
##   species     bmi body_mass_g flipper_length_mm  
##   <fct>     <dbl>      <int>             <int>  
## 1 Adelie    20.7       3750              181  
## 2 Adelie    20.4       3800              186
```

Tidyverse: `mutate()`: Crea nuevas columnas o transforma existentes

```
# Clasificación condicional con case_when()
penguins %>%
  mutate(size_class = case_when(
    body_mass_g < 3000 ~ "liviano",
    body_mass_g < 4000 ~ "medio",
    TRUE ~ "pesado"
  )) |> select(species, size_class, body_mass_g) |> head(2)
```

```
## # A tibble: 2 × 3
##   species size_class body_mass_g
##   <fct>    <chr>        <int>
## 1 Adelie   medio          3750
## 2 Adelie   medio          3800
```

Tidyverse: filter()

```
# Igualdad y conjunción (AND)
penguins %>%
  filter(species == "Adelie", # Esta especia
         island == "Biscoe") #Esta lista
```

```
# Disyunción (OR)
penguins %>%
  filter(species == "Adelie" | species == "Gentoo") #Cualquiera de ambas
```

Tidyverse: filter()

```
# Negación
penguins %>%
  filter(species != "Chinstrap") # Esa no

# Pertenencia múltiple
penguins %>%
  filter(species %in% c("Adelie", "Gentoo")) #Cualquiera de las dods

# Filtrar valores faltantes
penguins %>%
  filter(!is.na(island)) # No NA
```

Tidyverse: group_by() + summarise()

```
penguins %>%  
  group_by(species) %>%  
  summarise(  
    promedio_masa = mean(body_mass_g, na.rm = TRUE),  
    mediana_masa = median(body_mass_g, na.rm = TRUE),  
    conteo = n()  
)
```

```
## # A tibble: 3 × 4  
##   species  promedio_masa mediana_masa conteo  
##   <fct>        <dbl>       <dbl>     <int>  
## 1 Adelie      3701.       3700      152  
## 2 Chinstrap   3733.       3700      68  
## 3 Gentoo     5076.       5000     124
```

Aplicación de datos



```
casen2022 <- readRDS("data/casen2022_sample.rds") |> glimpse()
```

```
## Rows: 10,000
## Columns: 11
## $ region          <hvn_lbll> 5, 4, 7, 4, 9, 5, 13, 2, 5, 6, 13, 13, 13, 13, 3, 1...
## $ nse             <hvn_lbll> 4, 2, 1, 1, 6, 4, 2, 2, 2, 1, 7, 3, 2, 2, 1, 6, ...
## $ sexo            <hvn_lbll> 1, 1, 1, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, ...
## $ ecivil          <hvn_lbll> 2, 2, 8, 8, 8, 1, NA, 5, 8, 7, 8, 8, 5, 8, 8, 2...
## $ educ            <hvn_lbll> 5, 5, 5, 4, 5, 2, 1, 4, 9, 3, 9, 3, 6, 3, 7, 11...
## $ pueblos_indigenas <hvn_lbll> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ pobreza         <hvn_lbll> 3, 3, 3, 2, 3, 3, 3, 3, 3, 3, 3, 2, 3, 3, 3, 3, ...
## $ p9              <dbl> 3, 5, 5, 5, 3, 4, 5, 1, 3, 2, 3, 8, 3, 7, 3, 3, 6, 5...
## $ edad            <dbl> 43, 27, 26, 16, 22, 43, 7, 60, 24, 45, 30, 15, 51, 1..26
## $ yautecor        <dbl> 200000, 100000, 100000, NA, NA, 200000, NA, 260000, N...
```

Aplicación de datos

```
casen2022 |> select(pueblos_indigenas, yautcor, region, educ) |>  
  group_by(pueblos_indigenas) |> # Indígena (0 = No, 1 = Sí)  
  summarise(  
    promedio_ingreso = mean(yautcor, na.rm = TRUE))
```

```
## # A tibble: 2 × 2  
##   pueblos_indigenas  promedio_ingreso  
##             <hvn_lbl>          <dbl>  
## 1                      0            624245.  
## 2                      1            494642.
```

Comunicación de resultados: R orientado a objetos

```
tabla <- casen2022 |> select(pueblos_indigenas, yautcor, region, educ) |>  
  group_by(pueblos_indigenas) |> # Indígena (0 = No, 1 = Sí)  
  summarise(  
    promedio_ingreso = mean(yautcor, na.rm = TRUE))
```

```
tabla
```

```
## # A tibble: 2 × 2  
##   pueblos_indigenas promedio_ingreso  
##   <dbl>                <dbl>  
## 1 0                  624245.  
## 2 1                  494642.
```

Comunicación de resultados

```
readr::write_csv(tabla, "table/tabla.csv") #Guardar en carpeta table
```

tabla

	pueblos_indigenas	promedio_ingreso
1	0	617099.018981019
2	1	465818.709677419

Muchas Gracias

Vamos al código!