

Ayuda básica para usar R

R es un programa y un lenguaje de programación diseñado para trabajar con datos. Durante el curso, vamos a trabajar mucho con R, ya que es una herramienta clave para analizar datos, construir modelos y visualizar resultados. No necesitás ser experto ni experta desde el inicio, pero cuanto mayor comodidad tengas usando R, más vas a poder aprovechar los contenidos.

Si tenés dudas o dificultades con R, no te preocupes: vas a tener recursos, práctica y acompañamiento a lo largo del curso. Y recordá que buscar en internet o consultar herramientas como ChatGPT también está totalmente permitido (¡y recomendado!).

A continuación, mencionamos algunos aspectos de R especialmente relevantes para entender los scripts de R y complementar (o recordar) lo que hayas visto en algún curso introductorio o en el uso que hayas hecho de R previamente.

Algunas generalidades de R

- Cuando se define un objeto o se le asigna un valor se usa el símbolo "<=". Equivale al "=" en otros lenguajes.
- R es sensible a mayúsculas y minúsculas en todos los nombres de objetos y funciones.
- Se puede usar el punto en el medio del nombre de un objeto o función, pero para los objetos que creemos preferimos usar el guion bajo.
- Los espacios no tienen significado para R, pero es de buena práctica usarlos con igual criterio que los usamos al escribir español.
- Los caracteres que siguen a un numeral ("#") R no los lee. Eso se usa para escribir comentarios o evitar que se ejecuten ciertas líneas de código.

Objetos de R

- R tiene los siguientes tipos de objetos: escalares, vectores, matrices, arrays, listas y dataframes. En todos ellos cada elemento puede recibir un rótulo (por ej., títulos para las columnas y las filas de una matriz o un dataframe).
- Las columnas de un dataframe y los elementos de una lista se pueden llamar por su nombre usando el signo "\$": por ej., "d\$id" llama la columna "id" del dataframe "d".
- Los elementos de un vector, una matriz y un dataframe se llaman por su posición o nombre entre corchetes. Por ej., "vector[2]" o "matriz[2,3]" o "df[2,'nombre_col']".
- "matriz[2,]" llama toda la segunda fila del objeto "matriz".

Funciones de R

- Para cualquier función (comando) de R se puede consultar la ayuda ejecutando el nombre de la función (sin paréntesis posteriores) y antecedido del símbolo "?" (por ej., "?sum")
- Las funciones de R siempre se escriben seguidas de un paréntesis que abre, los argumentos de la función (si los tiene) y un paréntesis que cierra. Esa es la sintaxis completa genérica para ejecutar cualquier función.
- Cada función tiene un orden por defecto para sus argumentos. Si los argumentos no se especifican en ese orden, es necesario explicitar qué argumento se está especificando. En la ayuda de cada función se puede encontrar el detalle de los argumentos.

Algunas particularidades de sintaxis

- "2:5" genera una secuencia que va de 2 a 5.
- "%>%" se llama pipe (del inglés). Si una escribe "d %>% sum()", equivale a "sum(d)".
- "c(1, 5, 8)" crea un vector con los elementos enumerados entre los paréntesis.

Consideraciones sobre los scripts del curso

- Los scripts del curso procuran tener una misma estructura con un mismo orden: un título, una descripción del script ("Notas"), cuestiones del entorno ("Entorno"), cargar las librerías/paquetes ("Paquetes"), definir todos los objetos a los que se les asignan los valores "a mano" ("Definiciones") y después diferentes títulos describiendo las grandes secciones de los scripts.
- Los subtítulos de mayor jerarquía empiezan con un solo "#". Los del siguiente nivel de jerarquía, con "##" y así.
- Varias líneas de código tienen comentarios explicativos a su derecha o en la línea previa.
- Los comentarios en la línea previa empiezan con un "#" (al igual que los subtítulos), pero está todo en minúscula y sin los "----" al final de la línea (que sirven para colapsar la sección de ese subtítulo).
- Los comentarios que empiezan con "##" en la misma línea a continuación del código dan más detalle de la sintaxis de esa línea.