

Introducción a R y Tidyverse

Sesión 02

Laboratorio de Innovación en Salud

2022-04-02

 @healthinnovation
 @innovalab_imt
 innovalab.info



InnovaLab

Contenidos

- Uso de funciones y argumentos
- Instalación y manejo de paquetes
- Introducción a tidyverse
- Importación de datos de principales formatos

Exploración competencial

¿Qué es una función?

Las funciones son módulos de código autónomo que realizan una tarea específica y generalmente, toman algún tipo de estructura de datos (**vector**, **dataframes**, etc.), lo procesan y devuelven un resultado.

El uso general de una función es el nombre de la función seguida de paréntesis

```
function_name(input)
```

Los inputs se denominan argumentos e incluyen:

- El objeto físico (cualquier estructura de datos) en el que la función lleva a cabo una tarea
- Especificaciones que alteran la forma en que opera la función

```
> sd()
```

◆ x =

◆ na.rm =

x

a numeric vector or an **R** object but not a factor coercible to numeric by `as.double(x)`.

Press F1 for additional help

Buscando ayuda sobre las funciones

La mejor forma de averiguar esta información es utilizar `?` seguido del nombre de la función. Al hacer esto, se abrirá el manual de ayuda en el panel inferior derecho de RStudio que proporcionará una descripción de la función, uso, argumentos, detalles y ejemplos:

```
?sd()
```

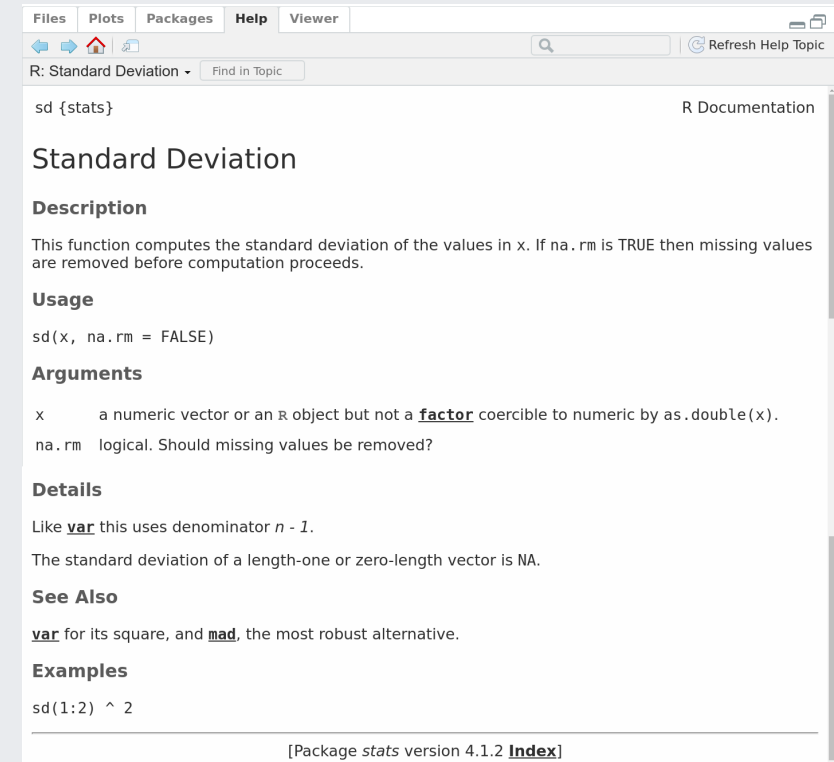
```
help(sd)
```

```
> sd() Presionar F1
```

Alternativamente, si está familiarizado con la función pero solo necesita recordar los nombres de los argumentos, puede usar:

```
args(sd)
```

```
## function (x, na.rm = FALSE)  
## NULL
```



The screenshot shows the RStudio interface with the Help panel open. The title bar indicates 'R: Standard Deviation'. The panel content is titled 'sd {stats}' and 'Standard Deviation'. It includes sections for Description, Usage, Arguments, Details, See Also, and Examples. The Description states: 'This function computes the standard deviation of the values in x. If na.rm is TRUE then missing values are removed before computation proceeds.' The Usage section shows: 'sd(x, na.rm = FALSE)'. The Arguments section lists: 'x a numeric vector or an R object but not a factor coercible to numeric by as.double(x). na.rm logical. Should missing values be removed?'. The Details section notes: 'Like var this uses denominator n - 1. The standard deviation of a length-one or zero-length vector is NA.' The See Also section mentions: 'var for its square, and mad, the most robust alternative.' The Examples section shows: 'sd(1:2) ^ 2'. At the bottom, it says '[Package stats version 4.1.2 Index]'.

Ejemplo de una función

Se tiene el número 3.15181930, pero solo necesitamos dos decimales. Para ello, utilizaremos la función `round()` que redondea los números de acuerdo a la cantidad de decimales que asignemos. En este caso, solo necesitaremos 2.

```
round(3.15181930, digits = 2)
```

```
## [1] 3.15
```

Como se puede observar, se ha utilizado el argumento `digits` para regular la cantidad de decimales.

Nota: Si proporcionamos los argumentos en el mismo orden en el que han sido definidos, no es necesario nombrarlos

```
round(3.15181930, 2)
```

```
## [1] 3.15
```

¿Data.frame?

- Estructura de datos 2D
- Admite datos con diferente tipo de variable (lo opuesto a matrices)
- Similar a Microsoft Excel

Se crean con la función:

```
data.frame(  
  Var1 = elementos1,  
  Var2 = elementos2  
)
```

```
var1 ← c("Peru", "Argentina", "Bolivia")  
var2 ← rep("aceptado",3)  
var3 ← seq(1000,1200,100)  
  
df ← data.frame(var1, var2, var3)  
df
```

```
##          var1      var2 var3  
## 1      Peru aceptado 1000  
## 2 Argentina aceptado 1100  
## 3  Bolivia aceptado 1200
```

```
df ← data.frame(  
  var1 = c("Peru", "Argentina", "Bolivia"),  
  var2 = rep("aceptado",3),  
  var3 = seq(1000,1200,100)  
)  
df
```

¿Tibble?

- Son la versión mejorada del data.frame
- Disponible en el **paquete** tibble y por lo tanto en el tidyverse.

Se crean con la función:

```
library(tibble)
tibble(
  Var1 = elementos1,
  Var2 = elementos2
)
```

```
install.packages("tibble")
```

```
library(tibble)
tibble(
  var1 = c("Peru", "Argentina", "Bolivia"),
  var2 = rep("aceptado", 3),
  var3 = seq(1000, 1200, 100)
)
```

```
## # A tibble: 3 × 3
##   var1      var2      var3
##   <chr>    <chr>    <dbl>
## 1 Peru      aceptado  1000
## 2 Argentina aceptado  1100
## 3 Bolivia   aceptado  1200
```


data.frame() v.s. tibble()

Ambas funciones tienen sus versiones `as.*` o `as_*` que permite transformar algo en lo que se desea. En este caso se estaría usando `as.data.frame` para convertir algo a **data.frame** y `as_tibble` para ese mismo objetivo.

```
class(iris)
```

```
## [1] "data.frame"
```

```
iris
```

```
##      Sepal.Length Sepal.Width Petal.Length
## 1           5.1           3.5           1.4
## 2           4.9           3.0           1.4
## 3           4.7           3.2           1.3
## 4           4.6           3.1           1.5
## 5           5.0           3.6           1.4
## 6           5.4           3.9           1.7
## 7           4.6           3.4           1.4
```

```
as_tibble(iris)
```

```
## # A tibble: 150 × 5
##   Sepal.Length Sepal.Width Petal.Length
##         <dbl>         <dbl>         <dbl>
## 1           5.1           3.5           1.4
## 2           4.9           3           1.4
## 3           4.7           3.2           1.3
## 4           4.6           3.1           1.5
## 5           5           3.6           1.4
## 6           5.4           3.9           1.7
## 7           4.6           3.4           1.4
## 8           5           3.4           1.5
## 9           4.4           2.9           1.4
## 10          4.9           3.1           1.5
```

¡Hazlo tú mismo!

1. Crea un tibble a partir de los ganadores de las últimas copas mundiales. La estructura de la data debe contener las siguiente variables:
 - Año
 - Lugar
 - Ganador
2. Existe una base de datos integrada dentro de R que se llama `airquality`.
Conviértela en un tibble, y guárdalo en otro objeto con distinto nombre.
3. Hacer lo mismo con la data `Titanic`.

08:00



¿Qué es un paquete?

Los paquetes son colecciones de funciones, datos y código compilado de R en un formato bien definido, creados para agregar una funcionalidad específica.

Hay un conjunto de paquetes estándar (o base) que se consideran parte del código fuente de R y están disponibles automáticamente como parte de su instalación de R.

Instalación de paquetes desde CRAN

La forma de instalar un paquete dependerá de dónde se encuentre. Entonces, para los paquetes disponibles públicamente, esto significa a qué repositorio pertenece. La forma más común es usar el repositorio CRAN, luego solo necesita el nombre del paquete y usa el siguiente comando:

```
install.packages("paquete")
```

Después de ejecutar esto, recibirá algunos mensajes en la pantalla. Dependerán del sistema operativo que esté utilizando, las dependencias y si el paquete se instaló correctamente.

Instalación de paquetes vía remotes

Cada repositorio tiene su propia forma de instalar un paquete a partir de ellos, por lo que en el caso de que utilice regularmente paquetes de diferentes fuentes, este comportamiento puede ser un poco frustrante. Una forma más eficiente es probablemente usar el paquete *remotes* para simplificar este proceso.

```
install.packages("remotes")
```

Después de haber instalado *remotes* podemos utilizar algunas de sus funciones para la instalación de paquetes:

- `remotes::install_bioc()` desde Bioconductor
- `remotes::install_github()` desde GitHub
- `remotes::install_version()` para instalar una versión específica de CRAN.

Base de datos a utilizar

Usaremos una base de datos proporcionado por Gan et al. (2018) en su estudio:

Article | [Open Access](#) | [Published: 13 June 2018](#)

Efficacy of two different dosages of levofloxacin in curing *Helicobacter pylori* infection: A Prospective, Single-Center, randomized clinical trial

[Huo-Ye Gan](#) , [Tie-Li Peng](#), [You-Ming Huang](#), [Kai-Hua Su](#), [Lin-Li Zhao](#), [Li-Ya Yao](#) & [Rong-Jiao Yang](#)

[Scientific Reports](#) **8**, Article number: 9045 (2018) | [Cite this article](#)

21k Accesses | **9** Citations | **1** Altmetric | [Metrics](#)

Abstract

Bismuth + proton pump inhibitor (PPI) + amoxicillin + levofloxacin is one of the bismuth quadruple therapy regimens widely used for the eradication of *H. pylori* infection. The recommended dosage of levofloxacin is 500 mg once daily or 200 mg twice daily to eradicate *H. pylori* infection. The aim of the present open-label, randomized control trial was to compare the effectiveness, safety, and compliance of different dosages of levofloxacin used to cure *Helicobacter pylori* infection. Eligible patients were randomly assigned to receive esomeprazole, amoxicillin, colloidal bismuth pectin and levofloxacin 500 mg once/day (group A) or levofloxacin 200 mg twice/day (group B) for 14 days. The primary outcome was the

Download PDF



Sections

Figures

References

[Abstract](#)

[Introduction](#)

[Materials and Methods](#)

[Results](#)

[Discussion](#)

[References](#)

[Author information](#)

[Ethics declarations](#)

[Additional information](#)

[Rights and permissions](#)

[About this article](#)

[Comments](#)





Base de datos a utilizar

Dicho estudio es un ensayo clínico aleatorizado (ECA) de 400 participantes en el que se compara 2 tipos de tratamientos para erradicar la infección por *Helicobácter Pylori*.

- **Grupo A** de tratamiento: Esomeprazol, amoxicilina, bismuto coloidal con pectina y levofloxacin 500mg una vez al día.
- **Grupo B** de tratamiento: Levofloxacin 200mg dos veces al día, durante un periodo de 14 días.

Randomized group	n
group A	200
group B	200

Importación de datos

Función	Tipo de archivo
<code>readxl::read_excel()</code>	
<code>readr::read_csv()</code>	
<code>haven::read_dta()</code>	
<code>haven::read_sav()</code>	

Importación de datos

Vamos a importar el archivo excel que tiene una extensión `.xlsx` y se encuentra dentro de la carpeta `data`.

Recuerda: Al importar el archivo debemos asignarlo a un objeto, para poder guardar la información. En este caso llamaremos a este objeto como `trial_data`.

```
trial_data ← readxl::read_excel("data/researchdata.xlsx")
```

```
trial_data
```

```
## # A tibble: 400 × 10
##   `Patient number` `Baseline 13C-UBT` `Randomized gro...` `Follow-up13C-...` `adverse drug ...`
##           <dbl> <chr>           <chr>           <chr>           <chr>
## 1             1 Positive         group B         Negative        No
## 2             2 Positive         group A         Negative        Yes
## 3             3 Positive         group B         Negative        No
## 4             4 Positive         group A         Positive        Yes
## 5             5 Positive         group A         Negative        Yes
## 6             6 Positive         group A         Negative        No
```

Importación de datos 2


En algunos casos estaremos frente a base de datos que contengan cierta información en sus primeras filas que no sean relevantes como datos a considerar para el análisis. La mayor cantidad de veces tienen un propósito meramente informativo. Ejemplo:

researchdata2.xlsx

Home
 Insert
 Layout
 Formula
 Data
 Pivot Table
 Collaboration
 Protection
 View
 Plugins

Times New Roman
 11
 A⁺
 A⁻
 T
 B
 I
 U
 A₂
 A
 General
 Normal
 Neutral
 Bad

I18
 NA

	B	C	D	E	F	G	H	I
1	<div>  </div>							
2								
3								
4	Baseline ¹³ C-UBT	Randomized group	Follow-up ¹³ C-UBT (4 weeks after therapy)	adverse drug reactions	adverse drug reactions content	Adverse drug reaction classified	Complete the study	Uncompleted Reason
5	Positive	group B	Negative	No	NA	NA	Yes	NA
6	Positive	group A	Negative	Yes	Fatigue	Mild	Yes	NA
7	Positive	group B	Negative	No	NA	NA	Yes	NA
8	Positive	group A	Positive	Yes	Abdominal pain	Mild	Yes	NA
9	Positive	group A	Negative	Yes	Drowsiness	Mild	Yes	NA
10	Positive	group A	Negative	No	NA	NA	Yes	NA

Importación de datos 2

Situación	Problema	Solución
-----------	----------	----------

En estas situaciones tenemos 2 alternativas:

a) Editar el archivo en cuestión y eliminar las filas que no sean relevantes. b) Durante la importación indicar que se omitan las primeras filas o filtrarlas una vez se haya importado.

Siempre será una mejor opción manejar los cambios desde el código, ya que de esa manera mantenemos los archivos originales y además podemos tener un registro de los cambios realizados.

Para hacer esto usaremos el argumento `skip` dentro de la función `readxl::read_excel()`, que indicará la cantidad de filas que deseamos omitir en la importación.

Importación de datos 2

Situación	Problema	Solución
-----------	----------	----------

Si intentamos importar la data sin especificar ningún argumento, veremos como se registran informaciones que no requerimos, y solo se encuentran en los datos a manera de información.

```
trial_data2 <- readxl::read_excel("data/researchdata2.xlsx")

trial_data2

## # A tibble: 403 × 10
##   ... 1      `Data de ejemplo para ...` ... 3   ... 4   ... 5   ... 6   ... 7   ... 8   ... 9   ... 10
##   <chr>      <chr>      <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 <NA>      <NA>      <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 2 <NA>      <NA>      <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 3 Patient number Baseline 13C-UBT Rand... Foll... adve... adve... Adve... Comp... Unco... per ...
## 4 1          Positive grou... Nega... No      NA      NA      Yes     NA      Yes
## 5 2          Positive grou... Nega... Yes     Fati... Mild   Yes     NA      Yes
## 6 3          Positive grou... Nega... No      NA      NA      Yes     NA      Yes
```

Importación de datos 2

Situación	Problema	Solución
-----------	----------	----------

Ya que en este caso la base de datos llamada `researchdata2.xlsx` empieza a mostrar datos relevantes a partir de la fila 4, requeriremos omitir o saltar (`skip`) 3 filas, se la siguiente manera:

```
trial_data2 ← readxl::read_excel("data/researchdata2.xlsx",
                                skip = 3)
```

```
trial_data2
```

```
## # A tibble: 400 × 10
##   `Patient number` `Baseline 13C-UBT` `Randomized gro...` `Follow-up13C-...` `adverse drug ...`
##   <dbl> <chr>                <chr>                <chr>                <chr>
## 1           1 Positive          group B          Negative          No
## 2           2 Positive          group A          Negative          Yes
## 3           3 Positive          group B          Negative          No
## 4           4 Positive          group A          Positive          Yes
## 5           5 Positive          group A          Negative          Yes
```

¡Hazlo tú mismo!

Todas las importaciones requeridas serán asignadas en un objeto.

1. Realizar la importación del archivo excel
2. Realizar la importación del archivo csv
3. Realizar la importación del archivo spss

08:00



InnovaLab

Retroalimentación

¡Gracias!

✉ **imt.innovlab@oficinas-upch.pe**

🐙 **@healthinnovation**

🐦 **@innovalab_imt**

Estas diapositivas fueron creadas mediante el paquete `xaringan` y `xaringantheme`.