

# Primeros Pasos en R

## Clase 3: Operaciones Básicas en R

Ana Alvarado, María Constanza Prado y Riva Quiroga

# Contenido del curso

---

## Módulo 1: Introducción a R.

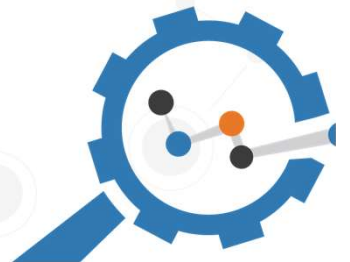
- Descripción software R.
- Diferencia entre R y R Studio.
- Interfaz (consola, editor, menú).
- Lenguaje de programación en R.
- Instalación de paquetes.

## Módulo 2: Operaciones básicas en R.

- Importar bases de datos.
- Crear, poblar y grabar bases de datos.
- Concatenar / separar archivos (registros y variables) de Estructuras Estadísticas
- Transformar/re codificar variables.
- Creación de objetos.
- Creación de funciones.

## Módulo 3: Análisis exploratorio y descriptivo en R.

- Calcular estadísticas de resumen.
- Generar tablas.
- Generar gráficos.



# La clase de hoy:

---



## Parte I

- Consultas y comentarios de la clase anterior
- Repaso
- Manipulación de Objetos
- Importación de archivos

## Parte II

- Análisis descriptivo de datos
  - Datos Cuantitativos
  - Datos Cualitativos
- Taller práctico

# Consultas y comentarios clase anterior



# Repaso



# Operadores

---

## Aritméticos

+	Suma
-	Resta
*	Multiplicación
/	División
%/%	División entera

## Comparación

<	menor que
>	mayor que
<=	menor o igual que
>=	mayor o igual que
==	igualdad
!=	diferencia

## Lógicos

!x	negación
x & y	intersección
x   y	unión

# Tipos de estructuras de datos (objetos)

---

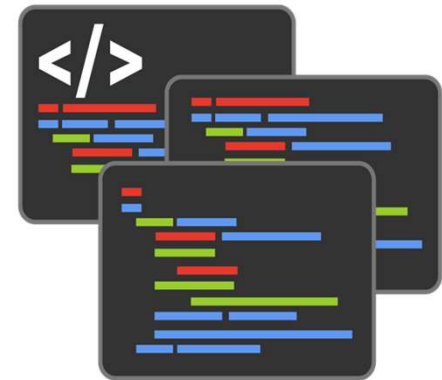
Escalares	<code>x &lt;- 3</code>
Vectores	<code>c(25,34,45,40,32,23)</code> <code>c(Lun = 5, Mar = 3, Mie = 6)</code>
Data frames (lista de vectores de igual longitud)	<code>data.frame(mes = c("Ene", "Feb", "Mar", "Abr"), num = c(34,56,1,23))</code>
listas (vector de diferentes objetos)	<code>list( A = "Hola", num = 1:4, B = c(1,2,3))</code>
Otros: matrices, arrays	<code>matrix(1:20,nrow=5,ncol=4)</code>

También podemos encontrarnos con [tibbles](#) es un tipo de data frame que se usa en el entorno [tidyverse](#).

# Funciones

---

<b>c()</b>	Concatena objetos (variables, textos, números, etc.)
<b>help()</b>	Ayuda respecto de alguna función
<b>library()</b>	Carga de librerías
<b>ls()</b>	Lista de objetos
<b>rm()</b>	Eliminar objetos
<b>abs()</b>	Valor absoluto
<b>sqrt()</b>	Raíz cuadrada
<b>exp()</b>	Exponencial
<b>log10()</b>	Logaritmo base 10
<b>log()</b>	Logaritmo natural
<b>round()</b>	Redondear
<b>mean()</b>	Promedio aritmético
<b>sum()</b>	Suma





# Packages

---

```
install.packages("nombre_paquete")
```

```
library(nombre_paquete)
```

readxl

MASS

rgl

caret

randomforest

randomForestSRC

qcc

shiny

zoo

forecast

plyr

Knitr

xtable

tidyverse

ggplot2

tibble

tidyr

readr

purrr

dplyr

stringr

forcats

gbm

xgboost

e1071

Liblinear

kernelab

glmnet

gam

cluster

La función **search()** permite ver los *packages* actuales en funcionamiento.

# Manipulación de Objetos



# Manipulación de objetos

---

Para encontrar uno o más elementos que pertenecen a un objeto, se puede acceder a través del uso de los [paréntesis de corchete](#).

## Selección por posición

Un `vector[i]` entregará el valor en la posición `i`.

Un `vector[-i]` entregará todos los valores, excepto el de la posición `i`.

```
> vector <- c("A", "B", "C", "D", "E")  
> vector[2]  
[1] "B"
```

# Manipulación de objetos

---

## Selección por comparación

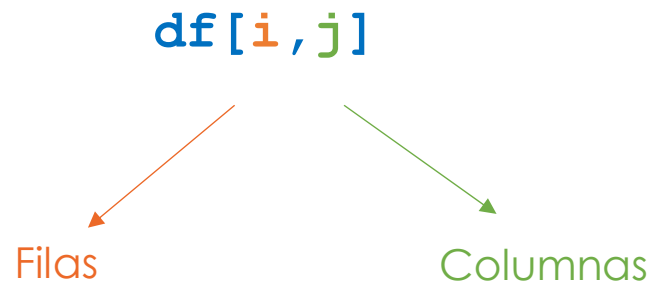
Se selecciona el elemento cuya condición establecida por el operador de comparación es VERDADERA

```
> vector <- c(1,2,3,4,5)
> selec <- vector > 3
> selec
[1] FALSE FALSE FALSE TRUE TRUE
> vector[selec]
[1] 4 5
```

# Manipulación de Data Frames

---

Análogo para matriz y data frame, donde se utiliza:



```
df <- data.frame(...)
```

# Manipulación de Data Frames

---

	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

Selecciona filas 2 a 5 y columna 2

`df[2:5,2]`  
`df[2:5, "Sepal.Length"]`

# Manipulación de Data Frames

---

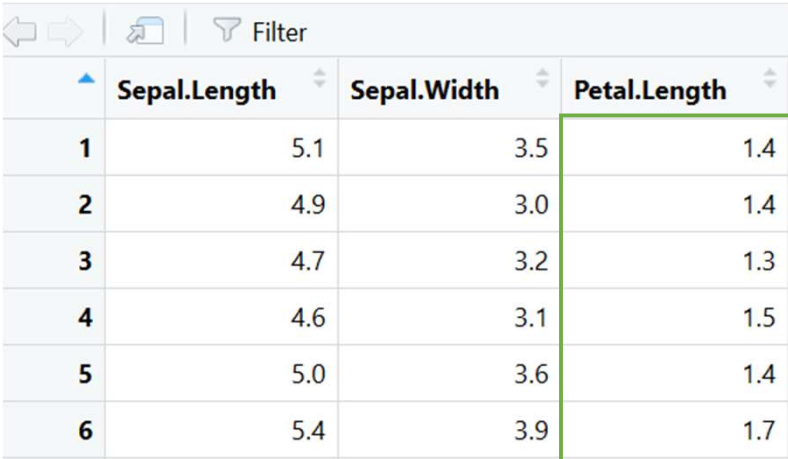
	Sepal.Length	Sepal.Width	Petal.Length
1	5.1	3.5	1.4
2	4.9	3.0	1.4
3	4.7	3.2	1.3
4	4.6	3.1	1.5
5	5.0	3.6	1.4
6	5.4	3.9	1.7

Selecciona las filas 2 y 3

`df[2:3,]`

# Manipulación de Data Frames

Por otra parte, los objetos se pueden conservar encadenados a través de otro objeto con el comando `$`, por ejemplo, `df$X`. De esta forma manipularemos objetos de un data frame en este curso.



	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



`df$Petal.Length`



# Manipulación de Data Frames

La función `subset()` funciona como un atajo para hacer lo mismo que hizo en los ejercicios anteriores.

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa



	Sepal.Length	Species
1	5.1	setosa
6	5.4	setosa
11	5.4	setosa
15	5.8	setosa
16	5.7	setosa
17	5.4	setosa
18	5.1	setosa

```
subset(x = iris, subset = Sepal.Length > 5.0,  
       select = c("Sepal.Length", "Species"))
```

# Manipulación de Data Frames

---

Los comandos `is.` y `as.` llevan a **identificar** objetos. Las posibilidades son numeric, vector, array, matrix, etc.

En modo de ejemplo, `is.vector(objeto)` es una consulta lógica, mientras que `as.vector(objeto)` es una orden.

# Importación de Archivos



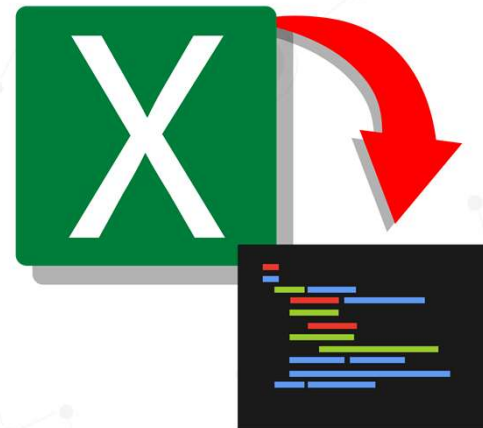
# Importación de archivo

---

Cosas a considerar:

- Tipo de archivo
- Título en la primera fila
- Separador
- Tipo de missing
- Decimales
- Otros...

El comando usual será `read`.



# Importación de archivos

---

Funciones en R base: `read.formato()`

`read.csv()`

`read.table()`

`read.delim()`

...

¿Importar un excel? →

Paquete `readxl`



`readxl::read_excel()`

`library(readxl)`  
`read_excel()`

`readxl::read_excel("ruta/nombredelarchivo.xlsx")`

# El paquete readr

---



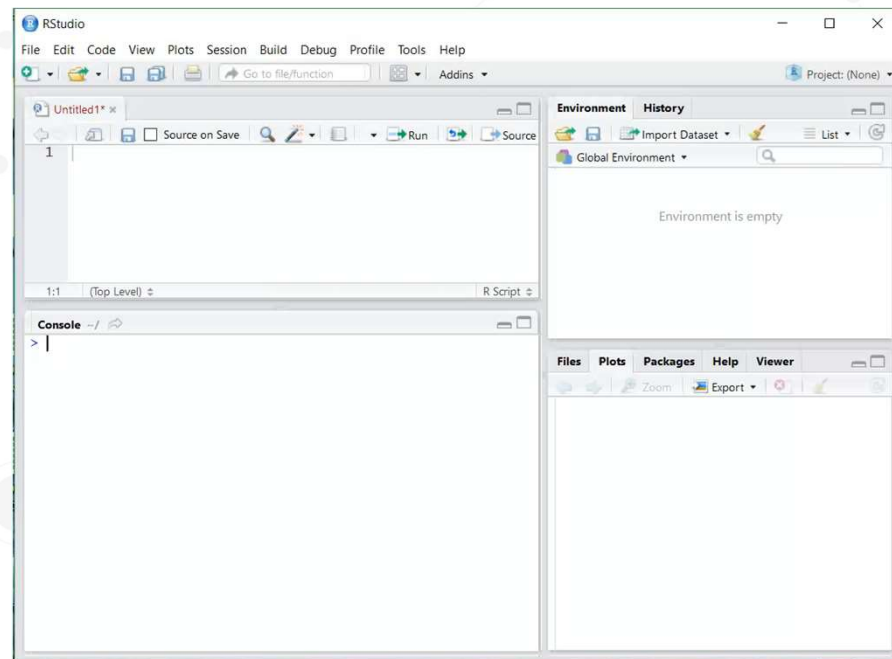
- `read_csv()` - archivos delimitados por comas
- `read_csv2()` - archivos separados por punto y coma (común en países donde se utiliza , como separador decimal)
- `read_tsv()` - archivos delimitados por tabulaciones
- `read_delim()` - lee archivos con cualquier delimitador
- `read_fwf()` - archivos de ancho fijo
- `read_table()` - variación común de archivos de ancho fijo donde las columnas están separadas por espacios en blanco
- ...

# Importación de archivos

También podemos importar archivos a través de las herramientas facilitadores de [Rstudio](#).

**File /  
Import Dataset /  
From ...**

Se ingresa una base de datos a través de "clic" de mouse, mientras que en una ventana auxiliar, se mostrara la vista previa, la opción para elegir la ubicación, un comando con opciones varias sobre la base de datos, y un cuadro auxiliar con el script necesario para ejecutar la acción.



# Exploración de Data Frame

---

Luego que la base de datos (df) esta cargada, los primeros comandos para operar son los siguientes:

**head(df, k) :** Muestra los primeros k registros.

**tail(df, k) :** Muestra los últimos k registros.

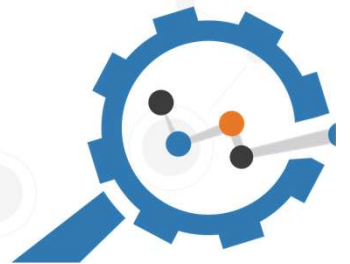
**dim(df) :** Filas y columnas de un objeto.

**length(df) :** Número de objetos dentro del objeto BD

**str(df) :** Estructura de la base de datos BD.

**class(df) :** Naturaleza del objeto (similar a is).

**names(df) :** Nombres.





# Actividad I ...

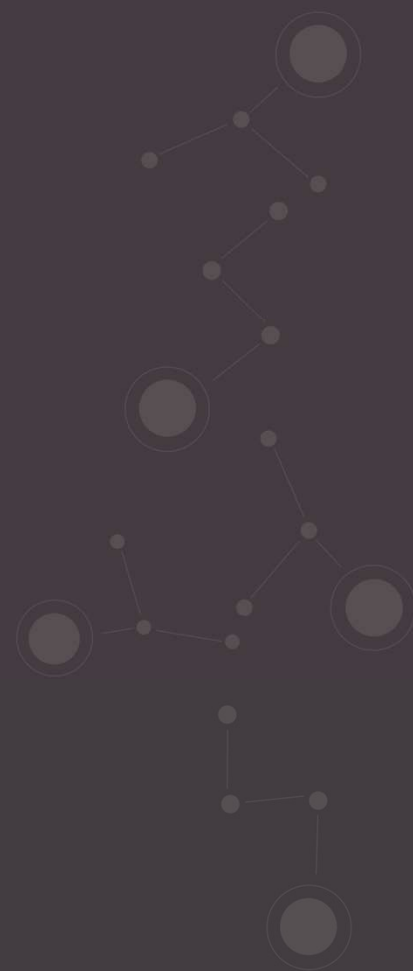
- Importa el archivo [nombres.xlsx](#)
- Filtra tu nombre usando sintaxis (R base) (es decir filtra usando subsetting con []).
- Encuentra los nombres más populares en 2016 y 2017. Si te hacen falta funciones, busca/usa el material compartido como referencias y material complementario (en especial las cheatsheets)



# Actividad II...

El archivo [Encuesta.xlsx](#) posee un extracto de la Encuesta Nacional de Seguridad Ciudadana, con sus principales variables.

- i. Importa la base de datos
- ii. Explora la estructura de sus datos. Usa: [head\(\)](#), [name\(\)](#), [dim\(\)](#), [length\(\)](#), [srt\(\)](#), [class\(\)](#).
- iii. Construya un nuevo data frame que contenga sólo la información de los Hombres.
- iv. Construya un nuevo data frame que contenga sólo la información de los Hombres de Valparaíso.
- v. Del objeto anterior conserve sólo las variables P1, P3, P8, P9, P21, P64 y P156



# Análisis descriptivo



# Análisis de datos cuantitativos

---

La función `summary()` entrega un resumen descriptivo de todas las variables. Mientras que `mean()`, `median()`, `min()`, `max()` y `quantile()` entregan valores individuales.

La función `aggregate()` divide los datos en subconjuntos Y, calcula alguna función estadística FUN sobre X para cada uno de ellos y devuelve el resultado en un formato apropiado.

`aggregate(X~Y, FUN)`

El package `agricolae` posee la función `table.freq(hist(x, plot = FALSE))` que reporta una tabla de intervalos.

# Análisis de datos cuantitativos

---

<code>summary(var) , summary(data)</code>	Resumen estadístico
<code>min(var)</code>	Mínimo
<code>max(var)</code>	Máximo
<code>range(var)</code>	Rango
<code>mean(var)</code>	Media aritmética
<code>median(var)</code>	Mediana
<code>length(var)</code>	Tamaño
<code>sd(var)</code>	Desviación estándar
<code>var(var)</code>	Varianza
<code>cov(var1,var2) , cor(data)</code>	Covarianza
<code>cor(var1, var2) , cor(data)</code>	Correlación
<code>quantile(var,0.25)</code>	Cuantil Q1
<code>quantile(var,0.75)</code>	Cuantil Q3

# Análisis de datos cuantitativos

---

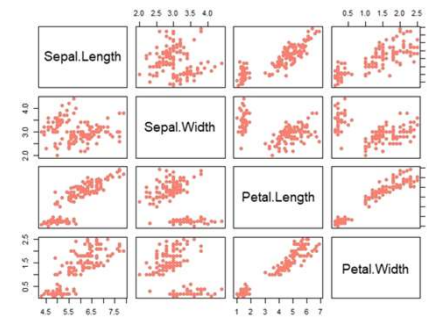
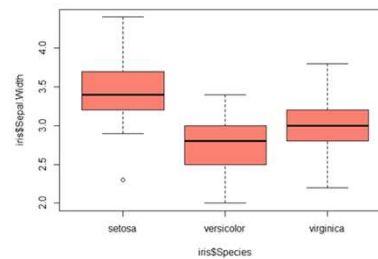
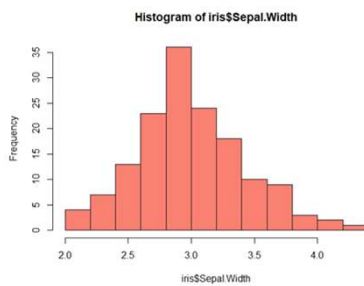
Con la librería `corrplot` puede dibujar una matriz de correlaciones. En la siguiente sección usaremos `ggcorrplot` para gráficos equivalentes.

La librería `PerformanceAnalytics` también entrega visualizaciones para evaluar la relación lineal entre las variables.

Otras librerías de interés: `ggplot2`, `scatterplot3d`, `highcharter`, `plotly`, `car`, `RColorBrewer`.

# Análisis gráfico de datos cuantitativos

<code>pie(table(var))</code>	Gráfico de torta
<code>barplot(table(var))</code>	Gráfico de barras
<code>hist(var)</code>	Histograma
<code>boxplot(var)</code>	Boxplot
<code>plot(var1,var2)</code>	Gráfico de dispersión
<code>pairs(data)</code>	Gráfico de dispersión cruzado
<code>ts.plot(var)</code>	Gráfico de series





# Análisis de datos cualitativos

---

`table(x)` entrega una tabla de frecuencia simple, mientras que `prop.table(table(x))` transforma dicha tabla en proporciones.

`table(x, y)` permite usar más variables para tablas de dos o más entradas.

Transforme una variable cualitativa o categórica con la función `factor(x, levels, labels)` o `as.factor()` para que sea considerada como tal.

# Actividad III...

En el archivo [viviendas.csv](#) se presentan los datos del censo de California de 1990. No es exactamente reciente (aún podría permitirse una bonita casa en el Bay Area), pero tiene muchas cualidades para practicar en la ciencia de datos.

longitud	Una medida de qué tan al oeste está una casa.
latitud	Una medida de qué tan al norte está una casa.
edad_prom_viv	Edad promedio de una casa dentro de un bloque.
total_hab	Número total de habitaciones dentro de un bloque.
total_dorm	Número total de dormitorios dentro de un bloque.
npoblacion	Número total de personas que residen dentro de un bloque.
nhogares	Número total de hogares en un bloque.
ingreso_prom	Ingreso promedio para hogares dentro de un bloque (medido en decenas de miles de dólares estadounidenses)
valor_prom_viv	Valor promedio de la casa dentro de un bloque (medido en dólares estadounidenses)
prox_oceano	Ubicación de la casa con respecto al océano

# Actividad III...

Realice un análisis descriptivo completo de los datos de las viviendas.

- i. Reporte un resumen descriptivo de todas las variables en el conjunto de datos. Interprete sus resultados.
- ii. Construya una tabla que indique la cantidad de missing en cada variable. Revise si hay mayor cantidad de missing en algún tipo de vivienda.
- iii. Realice un gráfico que permita observar la distribución de la edad promedio de la vivienda.
- iv. Construya una tabla que entregue el promedio del precio de la vivienda por tramo de antigüedad. Puede usar la función `cut()`.
- v. Realice un análisis de las correlaciones entre las variables.

# Actividad IV...

Usando los datos de la encuesta de seguridad ciudadana para los hombres de Valparaíso explore sus variables para responder las siguientes preguntas:

- i. ¿Cuántos hombres de Valparaíso están casados?
- ii. Obtenga una tabla de frecuencias relativas para el estado civil de los hombres de Valparaíso.
- iii. Obtenga el promedio de edad (P8) para aquellos que creen que serán víctimas de un delito (P64).
- iv. ¿Existe relación entre la edad y percepción de seguridad?

# Taller Práctico



# Taller práctico

---

Se disponen los datos epidemiológicos provenientes del Ministerio de Salud (MINSAL) y datos de otras fuentes, documentados y abiertos para el análisis de la comunidad, respecto al Covid19. Ver <http://www.minciencia.gob.cl/COVID19> para más información.

La base de datos "Covid.csv" cuenta con información de casos confirmados acumulados diarios por región.

Fuente:

<https://github.com/MinCiencia/Datos-COVID19/blob/master/output/producto3/CasosTotalesCumulativo.csv>

# Taller práctico

---

- i. Importe la base de datos Casos (Covid.csv).
- ii. Explore el contenido de la base de datos, sus dimensiones, nombres, etc.
- iii. Construya una variable que represente el porcentaje de casos nuevos, dejando la última fecha como la fecha actual.
- iv. Construya una variable que represente la zona (de manera aproximada) "Norte", "Centro", "Sur.
- v. Obtenga los casos totales nuevos promedio por Zona.
- vi. Calcule una variable que represente la proporción de casos nuevos en comparación al total de casos del día anterior.
- vii. Obtenga la proporción de casos nuevos promedio por Zona.
- viii. Escoja una región de su interés y grafique la serie de tiempo de casos confirmados de los últimos 30 días.

# Referencias y material complementario

- <https://cran.r-project.org/web/packages/janitor/vignettes/janitor.html> (Ejemplos de funciones del paquete janitor) este paquete contiene funciones para examinar y limpiar datos. (La función `clean_names()` es muy útil cuando importamos un set de datos y los nombres de las variables no tienen un formato adecuado).
- <https://cran.r-project.org/web/views/> Paquetes de R organizados por tema.