

# Curso Básico R - parte1

Daniel Enriquez

2023-12-28

# Directorio de Trabajo *(wd, working directory)*

- ▶ Dirección o ubicación (carpeta o folder) para importar/exportar/cargar/guardar archivos.
- ▶ Tip: *Evitar espacios, tildes o caracteres en el nombre.*
- ▶ Comandos útiles:
  - ▶ `getwd()`:Cuál es la dirección actual?
  - ▶ `setwd()`: Definir la dirección
  - ▶ `dir()`: Generar una lista de archivos y carpetas en el *wd*.
  - ▶ `ls()`: Genera una lista de objetos abiertos en R.

# Ejemplo N1

Los comentarios dentro del código son incluidos con #.

```
#Ejemplo en MacOS, en Windows "C:/Docum..")  
setwd("/Users/denriquez/Documents/Rstats/RepoGit/R-estudiar")  
#Comprobar el wd  
getwd()
```

```
## [1] "/Users/denriquez/Documents/Rstats/RepoGit/R-estudiar"
```

```
#Lista de archivos en wd  
dir()
```

```
## [1] "prueba.RData"          "R_1.html"              "R_1.Rmd"  
## [4] "R-estudiantes.Rproj"  "R-Github.Rmd"
```

# Objetos

Ojo: Cada Función trabaja con un tipo específico de Obj.  
getClassDef Tipos de Obj.:

- ▶ **Vectores:** números, caracteres, fórmulas.
- ▶ **Matrices:** varias columnas del mismo tipo y longitud.
- ▶ **Arrays:** como Matrices pero con más dimensiones.
- ▶ **Data frames:** clásica base de datos con columnas de diferentes tipos.
- ▶ **Listas:** lista de objetos de diferente tipo.
- ▶ **Factores:** Un vector con niveles.
- ▶ **Obj. Avanzados:** S3, S4 (Listas con información interconectada). Ejemplo: plots o información genética.

# Operaciones básicas

- ▶ Crear un vector (`<-` o `=`), concatenar (`c()`) y mostrar resultados. \*para valores repetidos: `rep`

```
dev<-c(1,2,3,4,5,6,7,8,9,10)
dev
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

- ▶ Características del objeto:

```
length(dev) #longitud
str(dev) #resumen o estructura
```

```
## [1] 10
```

```
## num [1:10] 1 2 3 4 5 6 7 8 9 10
```

- ▶ Otros: `class()`, `mode()`.

# Operaciones básicas(II)

Manipular un vector.

```
dev[4] #Seleccionar el 4to elemento  
dev[5:8] #Seleccionar del 5to al 8vo elemento.  
dev2<-dev[-6] #Eliminar el 6to elemento  
dev2 #Mostrar el resultado de la operación anterior
```

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

```
## [1] 5 6 7 8
```

```
## [1] 1 2 3 4 5 7 8 9 10
```

## Operaciones básicas(III)

Crear una matriz.

```
dev.m<-matrix(data=11:20, nrow=2, ncol=5) #2 filas y 5 col  
is.matrix(dev.m) # es dev.m una matriz?  
dev.m #Mostrar la matriz  
dim(dev.m) #dimensiones. (2 x 5)  
dev.m[,3] #Mostrar la 3ra columna  
dev.m[2,] #Mostrar la 2da fila  
dev.m[2,3] #Mostrar un elemento con las coordenadas
```

```
## [1] TRUE  
##      [,1] [,2] [,3] [,4] [,5]  
## [1,]  11  13  15  17  19  
## [2,]  12  14  16  18  20  
## [1] 2 5  
## [1] 15 16  
## [1] 12 14 16 18 20  
## [1] 16
```

## Operaciones básicas(IV)

Manipular una matriz.

```
dev.m[,2:4] #Mostrar valores de las columnas 2 a 4  
dev1<-c(dev, dev.m[2,]) #concatenar el vector dev a la fila 2  
dev1  
dev1<-c(dev.m[1,]) #reemplazar vector por la fila 1  
dev1  
rm(dev1) #elimina dev1  
dev1<-rep(x=10, each=5) #5 repeticiones del valor 10  
dev1[dev1==10]<-"diez" #Operacion logica  
dev1
```

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



# Operaciones básicas(V)

```
summary(dev.m) #Estadísticas básicas
```

##	V1	V2	V3	V
##	Min. :11.00	Min. :13.00	Min. :15.00	Min.
##	1st Qu.:11.25	1st Qu.:13.25	1st Qu.:15.25	1st Qu.
##	Median :11.50	Median :13.50	Median :15.50	Median
##	Mean :11.50	Mean :13.50	Mean :15.50	Mean
##	3rd Qu.:11.75	3rd Qu.:13.75	3rd Qu.:15.75	3rd Qu.
##	Max. :12.00	Max. :14.00	Max. :16.00	Max.
##	V5			
##	Min. :19.00			
##	1st Qu.:19.25			
##	Median :19.50			
##	Mean :19.50			
##	3rd Qu.:19.75			
##	Max. :20.00			

## Operaciones básicas(VI)

```
colnames(dev.m)<-c("A", "B", "C", "D", "E") #Renombrar col  
dev.m  
dev.m[,c("C","E")] #Extraer dos columnas por su nombre  
t(dev.m) #Transponer elementos
```

```
##           A  B  C  D  E  
## [1,]  11 13 15 17 19  
## [2,]  12 14 16 18 20  
##           C  E  
## [1,]  15 19  
## [2,]  16 20  
##    [,1] [,2]  
## A     11     12  
## B     13     14  
## C     15     16  
## D     17     18  
## E     19     20
```

## Operaciones básicas(VII)

```
dev.m2<-matrix(rep(c(1:6)), nrow=6, ncol=6)
diag(dev.m2) #diagonal de la matrix
diag(dev.m2)<-rep(x=c(1,2), times=3)
dev.m2
```

```
## [1] 1 2 3 4 5 6
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    1    1    1    1    1
## [2,]    2    2    2    2    2    2
## [3,]    3    3    1    3    3    3
## [4,]    4    4    4    2    4    4
## [5,]    5    5    5    5    1    5
## [6,]    6    6    6    6    6    2
```

# Bases de Datos *(DF, data frame)*

*#¿Qué tipo de variable tenemos?*

```
dev.m2<-matrix(rep(c(1:6)), nrow=6, ncol=6)
```

```
is.matrix(dev.m2)
```

```
class(dev.m2)
```

*#Conversión a DF*

```
dev.df<-as.data.frame(dev.m2)
```

```
class(dev.df)
```

```
dev.df[dev.df==6]<-"NA" #reemplazar 6 por NA (not available)
```

```
dev.df$V6 #Acceder a la variable o ex-columna 6
```

```
## [1] TRUE
```

```
## [1] "matrix" "array"
```

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

```
## [1] "1" "2" "3" "4" "5" "NA"
```

```
rm(list=ls()) #limpiar todo el entorno
```

## Bases de Datos(II) *(DF. data frame)*

```
#Guardar todos los objetos
save(list=ls(), file="prueba.RData")
#Cargar todos los objetos
load("prueba.RData")
fix() #Editar objetos
rm() #eliminar objetos
```

R incluye un DF de prueba: iris

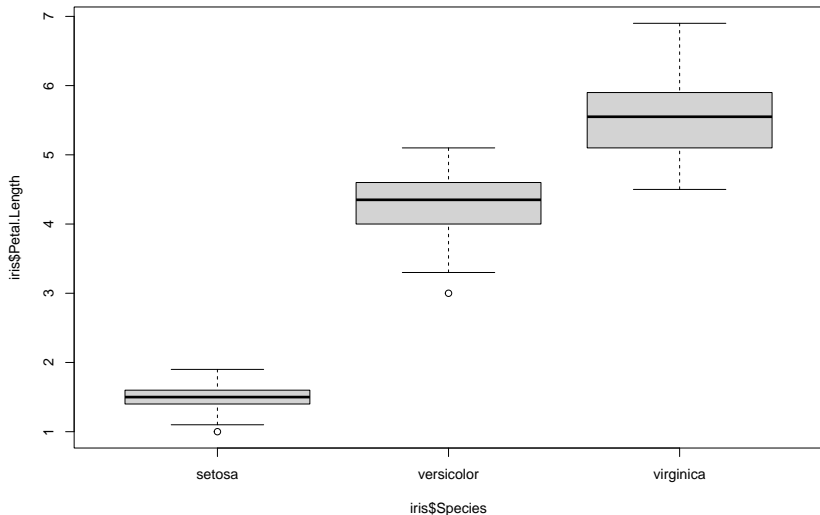
```
summary(iris) #estadísticas básicas
```

##	Sepal.Length	Sepal.Width	Petal.Length	Petal
##	Min. :4.300	Min. :2.000	Min. :1.000	Min.
##	1st Qu.:5.100	1st Qu.:2.800	1st Qu.:1.600	1st Qu.
##	Median :5.800	Median :3.000	Median :4.350	Median
##	Mean :5.843	Mean :3.057	Mean :3.758	Mean
##	3rd Qu.:6.400	3rd Qu.:3.300	3rd Qu.:5.100	3rd Qu.
##	Max. :7.900	Max. :4.400	Max. :6.900	Max.
##	Species			

## Bases de Datos(III) *(DF, data frame)*

```
#un boxplot
```

```
boxplot(formula=iris$Petal.Length~iris$Species)
```



## Bases de Datos(IV) *(DF, data frame)*

### Estadísticas Básicas

```
#Extraer solo información de una especie  
setosa<-iris[which(iris$Species=="setosa"),]  
#Correlación entre 'length' y 'width'  
cor.test(x=setosa$Sepal.Length, y=setosa$Sepal.Width)
```

```
##
```

```
##  Pearson's product-moment correlation
```

```
##
```

```
## data:  setosa$Sepal.Length and setosa$Sepal.Width
```

```
## t = 7.6807, df = 48, p-value = 6.71e-10
```

```
## alternative hypothesis: true correlation is not equal to 0
```

```
## 95 percent confidence interval:
```

```
##  0.5851391 0.8460314
```

```
## sample estimates:
```

```
##           cor
```

```
## 0.7425467
```

# Visualización de Datos *(Plotting)*

*#Plot correlación entre 'length' y 'width'*

```
plot(x=setosa$Sepal.Length, y=setosa$Sepal.Width, main="Sépalos",  
      xlab="Longitud", ylab="Ancho", pch=16, cex=1, col="blue",  
      abline(reg=lm(setosa$Sepal.Width~setosa$Sepal.Length), col="red"))
```

