

## 03-Construir DF

Adrian

14/1/2022

### Construyendo DF

`data.frame(vector_1,...,vector_n)` -> Construir un DF a partir de vectores introducidos en el orden en el que queremos disponer las columnas de la tabla - R Considera del mismo tipo de datos todas las entradas de una columna - Las variables tomarán los nombres de los vectores. Estos nombres se pueden especificar como argumento de la forma `variable = vector - rownames`: Especificar los identificadores de las filas - `stringsAsFactors` para evitar la transformacion de las columnas de tipo palabra en factores

```
Algebra = c(1,2,0,5,4,6,7,5,5,8)
Analysis = c(3,3,2,7,9,5,6,8,5,6)
Statistics = c(4,5,4,8,8,9,6,7,9,10)
grades = data.frame(Alg = Algebra, An = Analysis, Stat = Statistics)
str(grades)
```

```
## 'data.frame':    10 obs. of  3 variables:
##  $ Alg : num  1 2 0 5 4 6 7 5 5 8
##  $ An  : num  3 3 2 7 9 5 6 8 5 6
##  $ Stat: num  4 5 4 8 8 9 6 7 9 10
```

### Crear un DF

```
gender = c("H", "M", "M", "M", "H")
age     = c( 23,  45,  20,  30,  18)
family  = c(  2,   3,   4,   2,   5)

# Crear un dataframe
df = data.frame(genero = gender, edad = age, familia = family, stringsAsFactors = TRUE)

# Cambiar nombre de las filas
row.names(df) = c("P1", "P2", "P3", "P4", "P5")

str(df)
```

```
## 'data.frame':    5 obs. of  3 variables:
##  $ genero : Factor w/ 2 levels "H","M": 1 2 2 2 1
##  $ edad   : num  23 45 20 30 18
##  $ familia: num  2 3 4 2 5
```

```
# names() -> cambiar los nombres de las variables
# Cambiar el nombre de las variables de las filas y las columnas
dimnames(df) = list(
  c("Antonio", "Ricardo", "Adrian", "Maria", "Lucia"),
  c("Sexo", "Años", "MiembrosFamilia")
)
# Añadir nueva fila con datos
df = rbind(df, c("H", 31, 9))
# Añadir nueva columna con datos
df$Ingresos = c(10000,12000,15000,9000,20000,15000)
df
```

```
##           Sexo Años MiembrosFamilia Ingresos
## Antonio    H    23             2    10000
## Ricardo    M    45             3    12000
## Adrian     M    20             4    15000
## Maria       M    30             2     9000
## Lucia       H    18             5    20000
## 6           H    31             9    15000
```

## Cambiar tipos de datos

```
# Cambiar a caracteres
as.character(df$MiembrosFamilia)
```

```
## [1] "2" "3" "4" "2" "5" "9"
```

```
# Cambiar a enteros
as.numeric(df$MiembrosFamilia)
```

```
## [1] 2 3 4 2 5 9
```

```
# Cambiar a factor
as.factor(df$MiembrosFamilia)
```

```
## [1] 2 3 4 2 5 9
## Levels: 2 3 4 5 9
```

```
# Crear nuevo DF con un solo factor
df_m <- df[df$Sexo=="M"]
# Eliminar el factor "H" sobrante
df_m = droplevels(df_m)
str(df_m)
```

```
## 'data.frame':   6 obs. of  3 variables:
## $ Años          : chr  "23" "45" "20" "30" ...
## $ MiembrosFamilia: chr  "2" "3" "4" "2" ...
## $ Ingresos       : num  10000 12000 15000 9000 20000 15000
```

## Mas sobre sub-data Frames

```
# Tidyverse
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.5      v purrr 0.3.4
## v tibble 3.1.6       v dplyr 1.0.7
## v tidyr 1.1.4        v stringr 1.4.0
## v readr 2.1.1        v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

# Obtener las columnas que empiezan con cierta palabra
iris_petal = select(iris, starts_with("Petal"))
# Obtener las columnas que terminan con cierta palabra
iris_petal = select(iris, ends_with("Petal"))
head(iris_petal)

##      Petal.Length Petal.Width
## 1           1.4          0.2
## 2           1.4          0.2
## 3           1.3          0.2
## 4           1.5          0.2
## 5           1.4          0.2
## 6           1.7          0.4

# Extraer del DF las filas que cumplen la condicion
setosa <- subset(iris, Species == "setosa")
# Reiniciar el nombre de las filas
rownames(setosa) = 1:nrow(setosa)
head(setosa, 5)

##      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
```