

Laboratorio_6.R

GEMA SC

2022-05-13

```
# J_Garcia
# 13/05/22
# HW_04:Laboratorio 6: Variables y Datos en R

# Bases de Los vectores -----
--

wins = c(52, 51, 47, 47, 42)
losses = c(20, 21, 25, 25, 30)
win_loss_perc = wins/(wins + losses)
win_loss_perc # variables cuantitativas

## [1] 0.7222222 0.7083333 0.6527778 0.6527778 0.5833333

teams = c("UtJ", "PhS", "DnN", "LAC", "DIM")
# VECTOR de caracteres (variable cualitativa)

# Manipulación de vectores: subconjuntos -----
--

# extraer elementos de un vector utilizando corchetes []
# para acceder a los elementos de un vector.
# Dentro de los corchetes puede especificar uno o más valores numéricos
que correspondan a la(s) posición (es) de los elementos del vector

# PRIMER ELEMENTO DE "wins"
wins[1]

## [1] 52

# TERCER ELEMENTO DE "losses"
losses[3]

## [1] 25

# ÚLTIMO NOMBRE EN "teams"
teams[5]

## [1] "DIM"

length(teams) # da el número de valores
```

```

## [1] 5
teams[length(teams)]

## [1] "DIM"

sort(wins, decreasing = T) # ordena los valores de forma creciente o
decreciente

## [1] 52 51 47 47 42
rev(wins) # invierte los valores

## [1] 42 47 47 51 52

# Subconjuntos con índices lógicos -----
--

# para los subconjuntos lógicos se utiliza un vector lógico
# es un tipo particular de vector que toma los valores especiales
VERDADERO y FALSO, así como NA(No disponible).

# VICTORIA DE Utah Jazz
wins[teams == "UtJ"]

## [1] 52

# EQUIPOS CON VICTORIAS > 40
teams[wins > 40]

## [1] "UtJ" "PhS" "DnN" "LAC" "DIM"

# NOMBRE DE LOS EQUIPOS CON DERROTAS ENTRE 10 Y 29
teams[losses >= 10 & losses <= 29]

## [1] "UtJ" "PhS" "DnN" "LAC"

# Factores y variables cualitativas -----
--

# VECTOR NUMÉRICO
num_vector <- c(1, 2, 3, 1, 2, 3, 2)
# CREAR UN FACTOR A PARTIR DE num_vector
first_factor <- factor(num_vector)
first_factor

## [1] 1 2 3 1 2 3 2
## Levels: 1 2 3

# tomar el vector teams y convertirlo como factor
teams = factor(teams)
teams

```

```

## [1] UtJ PhS DnN LAC DIM
## Levels: DIM DnN LAC PhS UtJ

# Secuencias -----
--

# OPERADOR DOS PUNTOS :
1:5

## [1] 1 2 3 4 5

1:10

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

-3:7

## [1] -3 -2 -1 0 1 2 3 4 5 6 7

10:1

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

# FUNCIÓN SECUENCIA
seq(from = 1, to = 10)

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

seq(from = 1, to = 10, by = 1)

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

seq(from = 1, to = 10, by = 2)

## [1] 1 3 5 7 9

seq(from = -5, to = 5, by = 1)

## [1] -5 -4 -3 -2 -1 0 1 2 3 4 5

# Vectores repetidos -----
--

rep(1, times = 5) # repetir 1 cinco veces

## [1] 1 1 1 1 1

rep(c(1, 2), times = 3) # repetir 1 y 2 tres veces

## [1] 1 2 1 2 1 2

rep(c(1, 2), each = 2)

## [1] 1 1 2 2

```

```

rep(c(1, 2), length.out = 5)
## [1] 1 2 1 2 1
rep(c(3, 2, 1), times = 3, each = 2)
## [1] 3 3 2 2 1 1 3 3 2 2 1 1 3 3 2 2 1 1
# De vectores a estructura tabular (data frame) -----
--

dat = data.frame(
  Teams = teams,
  Wins = wins,
  Losses = losses,
  WLperc = win_loss_perc
)
dat

##   Teams Wins Losses   WLperc
## 1  UtJ   52    20 0.7222222
## 2  PhS   51    21 0.7083333
## 3  DnN   47    25 0.6527778
## 4  LAC   47    25 0.6527778
## 5  DIM   42    30 0.5833333

# extraer los valores en la columna teams usando $
dat$Teams

## [1] UtJ PhS DnN LAC DIM
## Levels: DIM DnN LAC PhS UtJ

# utilizar la notación de corchetes en la columna extraída como con
cualquier tipo de vector
dat$Wins[1]

## [1] 52

dat$Wins[5]

## [1] 42

# Del mismo modo, puede hacer subconjuntos lógicos:

# Victorias del equipo Utah
dat$Wins[dat$Teams == 'UtJ']

## [1] 52

# equipos con victorias > 40
dat$Teams[dat$Wins > 40]

```

```

## [1] UtJ PhS DnN LAC DIM
## Levels: DIM DnN LAC PhS UtJ

# nombre de Los equipos con derrotas entre 10 y 29
dat$Teams[dat$Losses >= 10 & dat$Losses <= 29]

## [1] UtJ PhS DnN LAC
## Levels: DIM DnN LAC PhS UtJ

# Tu Turno -----
--
teams[1]

## [1] UtJ
## Levels: DIM DnN LAC PhS UtJ

wins[1] - wins

## [1] 0 1 5 5 10

losses[1] - wins

## [1] -32 -31 -27 -27 -22

win_loss_perc[1] - win_loss_perc

## [1] 0.00000000 0.01388889 0.06944444 0.06944444 0.13888889

order(teams[1], na.last = TRUE, decreasing = FALSE)

## [1] 1

order(wins[1] - wins, na.last= TRUE, decreasing= FALSE)

## [1] 1 2 3 4 5

order(losses[1] - wins, na.last= TRUE, decreasing=FALSE)

## [1] 1 2 3 4 5

order(win_loss_perc[1] - win_loss_perc, na.last= TRUE, decreasing=FALSE)

## [1] 1 2 3 4 5

data.frame(teams[1], order (teams[1]))

## teams.1. order.teams.1..
## 1 UtJ 1

data.frame(wins[1] - wins, order(wins[1] - wins))

## wins.1....wins order.wins.1....wins.
## 1 0 1
## 2 1 2
## 3 5 3

```

```
## 4          5          4
## 5          10         5

data.frame(losses[1]- wins)

##   losses.1....wins
## 1          -32
## 2          -31
## 3          -27
## 4          -27
## 5          -22

data.frame(win_loss_perc[1]- win_loss_perc)

##   win_loss_perc.1....win_loss_perc
## 1          0.00000000
## 2          0.01388889
## 3          0.06944444
## 4          0.06944444
## 5          0.13888889
```