Factores, fechas y caracteres

Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Construimos un ejemplo

```
N <- 100
edad <- sample(seq(18, 40, 1), N, replace=TRUE)
summary(edad)
```

```
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
18.00   23.00   29.00   28.72   35.00   40.00
```

```
sexo <- sample(c('H', 'M'), N, replace=TRUE)
class(sexo)
summary(sexo)
```

```
[1] "character"
   Length     Class      Mode
      100 character character
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Una variable cualitativa se define con `factor`

```
sexo <- factor(sexo)
```

```
class(sexo)
```

```
[1] "factor"
```

```
summary(sexo)
```

```
 H  M
42 58
```

```
levels(sexo)
```

```
[1] "H" "M"
```

```
nlevels(sexo)
```

```
[1] 2
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Los `factor` sirven para agrupar

- Con la función `table`

```
table(edad > 30, sexo)
```

```
     sexo
       H  M
FALSE 23 35
TRUE  19 23
```

```
table(edad %in% 20:30, sexo)
```

```
     sexo
       H  M
FALSE 24 30
TRUE  18 28
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Los `factor` sirven para agrupar

- Con `tapply` o `aggregate`

```
tapply(edad,sexo, mean)
```

```
       H        M
29.19048 28.37931
```

```
aggregate(edad ~ sexo, FUN=median)
```

```
  sexo edad
1    H 29.5
2    M 28.0
```

Los factores sirven para separar

```
edadSexo <- split(edad, sexo)
class(edadSexo)
```

```
[1] "list"
```

```
sapply(edadSexo, mean)
```

```
       H        M
29.19048 28.37931
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Los `factor` se pueden generar a partir de variables numéricas

- Por ejemplo, con `cut`

```
gEdad <- cut(edad, breaks=4)
class(gEdad)
```

```
[1] "factor"
```

```
levels(gEdad)
```

```
[1] "(18,23.5]" "(23.5,29]" "(29,34.5]" "(34.5,40]"
```

- Nuevamente `table`

```
table(gEdad)
```

```
gEdad
(18,23.5] (23.5,29] (29,34.5] (34.5,40]
       28        24        21        27
```

```
table(gEdad, sexo)
```

```
           sexo
gEdad        H  M
  (18,23.5]  9 19
  (23.5,29] 12 12
  (29,34.5] 10 11
  (34.5,40] 11 16
```

Date

```
as.Date('2013-02-06')
```

```
[1] "2013-02-06"
```

```
as.Date('2013/02/06')
```

```
[1] "2013-02-06"
```

```
as.Date('06.02.2013')
```

```
Error en charToDate(x) :
  character string is not in a standard unambiguous format
```

```
as.Date('06.02.2013', format='%d.%m.%Y')
```

```
[1] "2013-02-06"
```

```
as.Date(37, origin='2013-01-01')
```

```
[1] "2013-02-07"
```

Secuencias temporales con Date

```
seq(as.Date('2004-01-01'), by='day', length=10)
```

```
[1] "2004-01-01" "2004-01-02" "2004-01-03" "2004-01-04" "2004-01-05"
[6] "2004-01-06" "2004-01-07" "2004-01-08" "2004-01-09" "2004-01-10"
```

```
seq(as.Date('2004-01-01'), by='month', length=10)
```

```
[1] "2004-01-01" "2004-02-01" "2004-03-01" "2004-04-01" "2004-05-01"
[6] "2004-06-01" "2004-07-01" "2004-08-01" "2004-09-01" "2004-10-01"
```

```
seq(as.Date('2004-01-01'), by='10 day', length=10)
```

```
[1] "2004-01-01" "2004-01-11" "2004-01-21" "2004-01-31" "2004-02-10"
[6] "2004-02-20" "2004-03-01" "2004-03-11" "2004-03-21" "2004-03-31"
```

POSIXct

- help(format.POSIXct)

```
as.POSIXct('2013-02-06')
```

```
[1] "2013-02-06 CET"
```

```
ISOdate(2013, 2, 7)
```

```
[1] "2013-02-07 12:00:00 GMT"
```

```
hoy <- as.POSIXct('2013-02-06')

format(hoy, '%Y')
```

```
[1] "2013"
```

```
format(hoy, '%d')
```

```
[1] "06"
```

```
format(hoy, '%m')
```

```
[1] "02"
```

```
format(hoy, '%b')
```

```
[1] "feb"
```

POSIxct

```
hora <- Sys.time()
hora
```

```
[1] "2015-04-19 17:39:47 CEST"
```

```
format(hora, '%H:%M:%S')
```

```
[1] "17:39:47"
```

```
format(hora, '%H␣horas,␣%M␣minutos␣y␣%S␣segundos')
```

```
[1] "17 horas, 39 minutos y 47 segundos"
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Secuencias temporales con POSIXct

```
seq(as.POSIXct('2004-01-01'), by='month', length=10)
```

```
[1] "2004-01-01 CET"  "2004-02-01 CET"  "2004-03-01 CET"  "2004-04-01 CEST"
[5] "2004-05-01 CEST" "2004-06-01 CEST" "2004-07-01 CEST" "2004-08-01 CEST"
[9] "2004-09-01 CEST" "2004-10-01 CEST"
```

```
seq(as.POSIXct('2004-01-01 10:00:00'), by='15 min',
    length=10)
```

```
[1] "2004-01-01 10:00:00 CET" "2004-01-01 10:15:00 CET"
[3] "2004-01-01 10:30:00 CET" "2004-01-01 10:45:00 CET"
[5] "2004-01-01 11:00:00 CET" "2004-01-01 11:15:00 CET"
[7] "2004-01-01 11:30:00 CET" "2004-01-01 11:45:00 CET"
[9] "2004-01-01 12:00:00 CET" "2004-01-01 12:15:00 CET"
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Zonas horarias

```
as.POSIXct('2013-02-06 15:30:00',
           tz='GMT')
```

```
[1] "2013-02-06 15:30:00 GMT"
```

```
as.POSIXct('2013-02-06 15:30:00',
           tz='Europe/Madrid')
```

```
[1] "2013-02-06 15:30:00 CET"
```

```
hawaii <- as.POSIXct('2013-02-06 15:30:00', tz='HST')
## Character
format(hawaii, tz='GMT')
```

```
[1] "2013-02-07 01:30:00"
```

```
## POSIXct
as.POSIXct(format(hawaii, tz='GMT'), tz='GMT')
```

```
[1] "2013-02-07 01:30:00 GMT"
```

Bastan unas simples comillas

```
cadena <- "Hola mundo"
class(cadena)
```

```
[1] "character"
```

```
nchar(cadena)
```

```
[1] 10
```

Un vector de character

```
cadenaVec <- c("Hola mundo", "Hello world")
nchar(cadenaVec)
```

```
[1] 10 11
```

```
length(cadenaVec)
```

```
[1] 2
```

```
cadenaVec[1]
```

```
[1] "Hola mundo"
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io

Para mostrarlos usamos `cat` o `print`

```
a <- 2
b <- 3
```

```
cat('La suma de', a, 'y', b, 'es', a + b, fill=TRUE
    )
```

```
La suma de 2 y 3 es 5
```

```
cat('La suma de', a, 'y', b, 'es', a + b, '\n',
    'La multiplicación de', a, 'por', b, 'es', a*b,
        '\n')
```

```
La suma de 2 y 3 es 5
La multiplicación de 2 por 3 es 6
```

Los character se pueden unir...

```
paste('Hello', 'World', sep='_')
```

```
[1] "Hello_World"
```

```
paste('X', 1:5, sep='.')
```

```
[1] "X.1" "X.2" "X.3" "X.4" "X.5"
```

```
paste(c('A', 'B'), 1:5, sep='.')
```

```
[1] "A.1" "B.2" "A.3" "B.4" "A.5"
```

```
paste(c('A', 'B'), 1:5, sep='.', collapse='|')
```

```
[1] "A.1|B.2|A.3|B.4|A.5"
```

... y también se pueden separar...

```
strsplit(cadenaVec, split='␣')
```

```
[[1]]
[1] "Hola"  "mundo"

[[2]]
[1] "Hello" "world"
```

```
strsplit(cadenaVec, split='')
```

```
[[1]]
 [1] "H" "o" "l" "a" " " "m" "u" "n" "d" "o"

[[2]]
 [1] "H" "e" "l" "l" "o" " " "w" "o" "r" "l" "d"
```

```
chSep <- strsplit(cadenaVec, split='␣')
class(chSep)
```

```
[1] "list"
```

```
length(chSep)
```

```
[1] 2
```

```
sapply(chSep, nchar)
```

```
     [,1] [,2]
[1,]    4    5
[2,]    5    5
```

... y, por supuesto, manipular

```
sub('o', '0', 'Hola Mundo')
```

```
[1] "H0la Mundo"
```

```
gsub('o', '0', 'Hola Mundo')
```

```
[1] "H0la Mund0"
```

```
substring(cadena, 1) <- 'HOLA'
cadena
```

```
[1] "HOLA mundo"
```

```
tolower(cadena)
```

```
[1] "hola mundo"
```

```
toupper(cadena)
```

```
[1] "HOLA MUNDO"
```


Oscar Perpiñán Lamigueiro
http://oscarperpinan.github.io