

Manipulación de datos en R (2)

Clase 3

Joaquin Cavieres G.

Estudiante doctorado

`j.cavieres.g@gmail.com`

4 de diciembre de 2019

MATRICES

- ▶ Las matrices representan otra forma de tener una colección de datos. Mientras los `data.frame` almacenan multiples objetos (numericos, caracteres, etc), las matrices deben ser de un sólo tipo.
- ▶ Se declara como `matrix()`, por ejemplo:

```
matrix(obs=NA, nrow=1, ncol=1, byrow=FALSE)
```

donde: `obs` son los datos observados, `nrow`, `ncol` numero de filas y columnas respectivamente, `byrow` hace que la matriz se llene por columnas (sino `byrow=TRUE`).

Ejemplo: `matrix(1:4, ncol=2)`

	[,1]	[,2]
[1,]	1	3
[2,]	2	4

MATRICES

- Crear una matriz con los datos de co2.

```
> mat.co2 <- matrix(c(years, co2), ncol=2,  
nrow=length(years))
```

```
> head(mat.co2)
```

	[,1]	[,2]
[1,]	1959	316.00
[2,]	1960	316.91
[3,]	1961	217.63

MATRICES

- Forma rápida de crear matrices.

```
> mat.co2 <- cbind(years, co2)
```

```
> head(mat.co2)
```

	[,1]	[,2]
[1,]	1959	316.00
[2,]	1960	316.91
[3,]	1961	217.63

MATRICES

- ▶ `rbind()` crea una matriz indexando dos o mas filas de vectores.

```
> mat.co2.row <- rbind(years, co2)
```

```
> t(mat.co2.row) # transpone la matriz creada
```

	[,1]	[,2]
[1,]	1959	316.00
[2,]	1960	316.91
[3,]	1961	217.63

ARRAYS

- Son matrices n-dimensionales.

```
> array.co2 <- array(data=c(years, co2), dim=c(nyears,  
2))
```

```
> head(array.co2) # transpone la matriz creada
```

	[,1]	[,2]
[1,]	1959	316.00
[2,]	1960	316.91
[3,]	1961	217.63

- Una matriz es un caso especial de un array, pero no un `data.frame`.

```
> is.array(array.co2)
```

```
[1] TRUE
```

```
> is.array(mat.co2)
```

```
[1] TRUE
```

```
> is.array(co2.Data)
```

```
[1] FALSE
```

Son estructura de datos más flexibles que las nombradas anteriormente. Cada variable puede ser de diferente característica y varias en largo (número de observaciones).

```
descripción <- "Concentración de co2 (ppm)"  
  
> lista.co2 <- list(Descri=descripción, years=nyears,  
data=mat.co2)
```


Listas

```
> lista.co2
```

```
$Descri
```

```
[1] ‘‘Concentración de co2 (ppm)’’
```

```
$years
```

```
[1] 46
```

```
$data
```

```
[1]
```

	[,1]	[,2]
[1,]	1959	316.00
[2,]	1960	316.91
[3,]	1961	217.63

Extraer elementos de una Lista

```
> data.lista <- as.list(data.frame)
```

```
$id
```

```
[1] 31 62 50 99 53 75 54 58 4 74
```

```
$age
```

```
[1] 12 18 20 17 14 8 12 24 24 21
```

```
$sex
```

```
[1] M F F M F M M F F M
```

```
Levels: F M
```

Extraer elementos

- ▶ Con la declaración `[[]]` también se pueden extraer elementos sin tener que anteponer `$` con el nombre de una variable (como en un `data.frame`).

```
> data.lista[[1]]
```

```
[1] 31 62 50 99 53 75 54 58 4 74
```

```
> data.lista$id
```

- ▶ Extraer primer elemento de una lista `[[]]` y primer elemento de un vector `[]`.

```
> data.lista[[1]] [1]
```

```
[1] 31
```

- ▶ Operadores mixtos

```
> data.lista$id [1]
```

```
[1] 31
```

Extraer elementos

```
> data.list$id <- matrix(data.lista$id, ncol=2)
```

```
> data.list
```

```
$id
```

	[,1]	[,2]
[1,]	31	75
[2,]	62	54
[3,]	50	58

```
$age
```

```
[1] 12 18 20 17 14 8 12 24 24 21
```

```
$sex
```

```
[1] M F F M F M M F F M
```

```
Levels: F M
```

Variables Categóricas

- ▶ Un factor puede ser utilizado para especificar una variable discreta (grupo), por ejemplo: Sexo.
- ▶ Los factores en R se usan para:
 - ▶ Manipulación rápida de datos.
 - ▶ Graficas.
 - ▶ Especificar modelos estadísticos.
- ▶ R especificará automáticamente las variables categóricas como factores cuando:
 - ▶ Se crean `data.frames`.
 - ▶ Leemos datos externos.
 - ▶ Leer archivos de otras aplicaciones.

LA FUNCIÓN `factor()`

Podemos especificar una variable categórica o numérica como factor declarandola como factor:

```
> estrato <- c("demersal", "demersal", "demersal",  
"pelagico", ...
```

```
> is.factor(estrato) # ¿Es factor "estrato"?
```

LA FUNCIÓN factor()

Podemos especificar una variable categórica o numérica como factor declarandola como factor:

```
> estrato <- c("demersal", "demersal", "demersal",  
"pelagico", ...
```

```
> is.factor(estrato) # ¿Es factor "estrato"?
```

```
[1] FALSE
```

LA FUNCIÓN `factor()`

Podemos especificar una variable categórica o numérica como factor declarandola como factor:

```
> estrato <- c("demersal", "demersal", "demersal",  
"pelagico", ...
```

```
> is.factor(estrato) # ¿Es factor "estrato"?
```

```
[1] FALSE
```

```
> estrato <- factor(estrato)
```


LA FUNCIÓN factor()

Podemos especificar una variable categórica o numérica como factor declarandola como factor:

```
> estrato <- c("demersal", "demersal", "demersal",  
"pelagico", ...
```

```
> is.factor(estrato) # ¿Es factor "estrato"?
```

```
[1] FALSE
```

```
> estrato <- factor(estrato)
```

```
[1] demersal demersal demersal pelagico
```

LA FUNCIÓN factor()

Podemos especificar una variable categórica o numérica como factor declarandola como factor:

```
> estrato <- c("demersal", "demersal", "demersal",  
"pelagico", ...
```

```
> is.factor(estrato) # ¿Es factor "estrato"?
```

```
[1] FALSE
```

```
> estrato <- factor(estrato)
```

```
[1] demersal demersal demersal pelagico
```

```
[1] Levels: demersal pelagico
```

EJERCICIOS

- ▶ Crear dos matrices A y B, donde A sea una matriz de 2x2 y B sea una matriz de 2x3, cada una con valores únicos.
- ▶ Crear un factor desde el siguiente vector, tal que 1 sea “fememino” y 2 sea “masculino”.

```
sexo <- c(1,1,2,1,2,2,2,1,1,1)
```

- ▶ Crear una lista que contenga las dos matrices y el factor (creados anteriormente).
- ▶ Extraer la primera fila de la matriz A de la lista.