

# Ejercicio 1.2: Codificación y Transformación

Silvia Pineda

## Índice

Datos Estrés . . . . .	1
1. Clasifica las variables en cuantitativas y cualitativas . . . . .	2
2. ¿Los nombres de variables son consistentes? . . . . .	2
3. Examina la estructura de los datos ( <b>str</b> ) ¿Cómo son los tipos de variables?	
¿Cambiarías alguna? . . . . .	2
4. ¿Hay algún NA? Utiliza <b>any(is.na(datos))</b> . . . . .	3
5. ¿Qué ID tiene la persona con mejor sueldo? . . . . .	3
6. ¿Cuántos trabajadores tienen un nivel de estrés mayor de 5? . . . . .	4
7. ¿Cuántos trabajadores tienen estrés mayor de 5 y además una carga física >	
3? . . . . .	4
8. Nos hemos equivocado en la introducción de datos y para todos los valores	
2 de carga física, queríamos haber puesto un 3. <i>Recodifica</i> esta variable	
incluyendo este cambio. Utiliza <b>if_else()</b> de la librería dplyr . . . . .	4
9. Crea una nueva variable llamada <i>expo_total</i> que se obtenga como suma de	
los valores de exposición en <i>carga física</i> , <i>carga psíquica</i> y <i>ruido</i> . . . . .	5
10. <i>Categoriza</i> la variable <i>estres</i> en otra nueva variable que llamaremos <i>es-</i>	
<i>tres_grupo</i> que incluya tres categorías: <i>bajo</i> : 1 a 2; <i>medio</i> : 3 a 7; <i>alto</i> :	
8 a 10. ¿Cuántas personas hay en cada categoría de la nueva variable.	
Utiliza <b>cut()</b> y <b>breaks()</b> . . . . .	5
11. Guarda la base de datos corregida . . . . .	6

## Datos Estrés

```
library(dplyr)
```

```
datos<-read.csv("estres.csv",header = T, sep = "\t")
```

## 1. Clasifica las variables en cuantitativas y cualitativas

V. Cuantitativas: id, estres, edad, peso, sueldo, carfisica, carpsiquica, ruido

V. Cualitativas: grupo, genero, contrato, jornada

## 2. ¿Los nombres de variables son consistentes?

```
datos <- rename(datos, ID = id, car_fisica = carfisica, car_psiquica = carpsiquica)
```

## 3. Examina la estructura de los datos (str) ¿Cómo son los tipos de variables? ¿Cambiarías alguna?

```
str(datos)
```

```
'data.frame':  90 obs. of  12 variables:
 $ ID      : int  75 76 77 78 79 80 81 82 83 84 ...
 $ estres  : int  8 9 2 2 2 2 2 2 8 8 ...
 $ grupo   : chr  "otros" "otros" "otros" "otros" ...
 $ edad    : int  25 27 27 29 30 30 31 31 31 26 ...
 $ peso    : int  68 65 69 60 71 61 62 69 73 58 ...
 $ genero  : chr  "hombre" "hombre" "hombre" "hombre" ...
 $ sueldo  : int  23000 25000 21000 22000 24000 27000 23000 26000 24000 25000 ...
 $ contrato : chr  "fijo" "fijo" "fijo" "fijo" ...
 $ jornada : chr  "completa" "completa" "completa" "completa" ...
 $ car_fisica : int  3 5 1 1 1 1 1 1 1 1 ...
 $ car_psiquica: int  2 2 2 2 2 2 2 2 2 2 ...
 $ ruido    : int  3 3 2 4 3 0 2 4 3 0 ...
```

```
##Convertimos grupo, genero, contrato y jornada en factores
columns_to_factor<-c("grupo","genero","contrato","jornada")
datos[,columns_to_factor]<-
  lapply(datos[,columns_to_factor],as.factor)

#Con dplyr
datos <- mutate(datos, across(all_of(columns_to_factor), as.factor))

str(datos)
```

```
'data.frame': 90 obs. of 12 variables:
 $ ID      : int  75 76 77 78 79 80 81 82 83 84 ...
 $ estres   : int  8 9 2 2 2 2 2 2 8 8 ...
 $ grupo    : Factor w/ 4 levels "administrativo",...: 4 4 4 4 4 4 4 4 4 4 ...
 $ edad     : int  25 27 27 29 30 30 31 31 31 26 ...
 $ peso     : int  68 65 69 60 71 61 62 69 73 58 ...
 $ genero   : Factor w/ 2 levels "hombre","mujer": 1 1 1 1 1 1 1 1 1 1 ...
 $ sueldo   : int  23000 25000 21000 22000 24000 27000 23000 26000 24000 25000 ...
 $ contrato : Factor w/ 2 levels "fijo","temporal": 1 1 1 1 1 1 1 2 2 1 ...
 $ jornada  : Factor w/ 2 levels "completa","parcial": 1 1 1 1 1 1 1 1 1 1 ...
 $ car_fisica : int  3 5 1 1 1 1 1 1 1 1 ...
 $ car_psiquica: int  2 2 2 2 2 2 2 2 2 2 ...
 $ ruido    : int  3 3 2 4 3 0 2 4 3 0 ...
```

#### 4. ¿Hay algún NA? Utiliza `any(is.na(datos))`

```
any(is.na(datos))
```

```
[1] FALSE
```

#### 5. ¿Qué ID tiene la persona con mejor sueldo?

```
datos$ID[which(datos$sueldo==max(datos$sueldo))]
```

```
[1] 86 71
```

```
datos$ID[which.max(datos$sueldo)] ###Cuidado con which.max que solo te da un valor!
```

```
[1] 86
```

```
#Con dplyr
filter(datos, sueldo == max(sueldo))
```

	ID	estres	grupo	edad	peso	genero	sueldo	contrato	jornada	car_fisica
1	86	7	otros	28	66	hombre	28000	fijo	completa	1
2	71	2	otros	23	65	hombre	28000	fijo	completa	2
		car_psiquica	ruido							
1		2	2							
2		3	2							

**6. ¿Cuántos trabajadores tienen un nivel de estrés mayor de 5?**

```
length(which(datos$estres>5))
```

```
[1] 35
```

```
#Con dplyr  
nrow(filter(datos,estres > 5))
```

```
[1] 35
```

**7. ¿Cuántos trabajadores tienen estrés mayor de 5 y además una carga física > 3?**

```
length(which(datos$estres>5 & datos$car_fisica>3))
```

```
[1] 10
```

```
#Con dplyr  
nrow(filter(datos,(estres > 5 & car_fisica > 3)))
```

```
[1] 10
```

**8. Nos hemos equivocado en la introducción de datos y para todos los valores 2 de carga física, queríamos haber puesto un 3. Recodifica esta variable incluyendo este cambio. Utiliza if\_else() de la librería dplyr**

```
library(dplyr)  
  
#ifelse y if_else hacen lo mismo, pero la segunda mantiene el tipo de variable,  
#es más estricto y por tanto evita muchos posibles errores, sobre todo con los missing.  
datos$car_fisica_corregida<-ifelse(datos$car_fisica==2,3,datos$car_fisica)  
  
datos$car_fisica_corregida<-if_else(datos$car_fisica==2,3,datos$car_fisica)  
  
table(datos$car_fisica,datos$car_fisica_corregida)
```

	1	3	4	5
1	14	0	0	0
2	0	37	0	0
3	0	25	0	0
4	0	0	13	0
5	0	0	0	1

9. Crea una nueva variable llamada *expo\_total* que se obtenga como suma de los valores de exposición en *carga física*, *carga psíquica* y *ruido*.

```
datos$expo_total<-datos$car_fisica+datos$car_psiquica+datos$ruido
head(datos)
```

	ID	estres	grupo	edad	peso	genero	sueldo	contrato	jornada	car_fisica
1	75	8	otros	25	68	hombre	23000	fijo	completa	3
2	76	9	otros	27	65	hombre	25000	fijo	completa	5
3	77	2	otros	27	69	hombre	21000	fijo	completa	1
4	78	2	otros	29	60	hombre	22000	fijo	completa	1
5	79	2	otros	30	71	hombre	24000	fijo	completa	1
6	80	2	otros	30	61	hombre	27000	fijo	completa	1

  

	car_psiquica	ruido	car_fisica_corregida	expo_total
1	2	3	3	8
2	2	3	5	10
3	2	2	1	5
4	2	4	1	7
5	2	3	1	6
6	2	0	1	3

10. Categoriza la variable *estres* en otra nueva variable que llamaremos *estres\_grupo* que incluya tres categorías: *bajo*: 1 a 2; *medio*: 3 a 7; *alto*: 8 a 10. ¿Cuántas personas hay en cada categoría de la nueva variable. Utiliza *cut()* y *breaks()*

```
datos$estres_grupo<-cut(datos$estres,breaks=c(1,2,7,10),
                        labels = c("bajo", "medio", "alto"))

table(datos$estres_grupo)
```

bajo	medio	alto
30	35	24

## 11. Guarda la base de datos corregida

```
write.csv(datos,"estres_corregido.csv")
```