Sesion 3. Limpieza de datos (I)

Curso: POL304 - Estadística para el análisis político 2

Jefes de práctica: Airam Bello, Chiara Zamora y Alexander Benites

Ciclo 2023-2

¿Recuerdan la base de datos sobre urbanización?. Para esta sesión, vamos a necesitar cargarla de nuevo.

1. Para quienes exportaron la base de datos, podemos cargarla usando el comando import

```
library(rio)
data="https://github.com/WendyAdrianzenRossi/Statistics/raw/main/midata.csv"
urban=import(data)
```

1. Limpieza de datos

Vemos que tenemos los datos, pero no están bien.

Usualmente las bases de datos que necesitamos para nuestros análisis estadísticos los conseguimos de fuentes de información oficial, por lo que ya están listas para ser utilizadas.

No obstante, con frecuencia nos toparemos con datos que necesitan ser manipulados para su correcto uso. Por ejemplo, al usar scrapping de páginas de internet.

Además, la limpieza de datos también es útil cuando una base de datos contiene exceso de información, o información que deseas manipular para un objetivo específico: por ejemplo, fusionar una data con otra, reconfigurar una variable de cadena a numérica, entre otros.

1.1 Primera columna

Veamos los datos de la primera fila y la primera columna

```
# fila 1, columna 1
urban[1,1]
```

```
## [1] "Afghanistan\n          \n      \n      "
```

Vemos que hay expresiones irregulares que pueden afectar nuestro análisis. Vamos a utilizar el comando **trimws()**, que elimina espacios en blanco antes y despues de texto.

Este comando utiliza dos argumentos:

- El argumento which indica si se eliminan los espacios a la izquierda de la cadena de caracteres (left), después de (right) o para ambos lados (both).
- El espacio en blanco

```
urban$COUNTRY =trimws(urban$COUNTRY,which=c("right"),whitespace = "[\\h\\v]") # el espacio en blanco se
```

Veamos de nuevo la primera fila y la primera columna:

```
urban[1,1]
```

```
## [1] "Afghanistan"
```

1.2 Segunda columna

Ahora veamos los datos de la primera fila y la segunda columna

```
# fila 1, columna 2
urban[1,2]
```

```
## [1] "\n      \n        urban population: 24% of total population (2008) \n      rate of urbanization
```

Aqui hay dos variables:

1. Población urbana (urban population)
2. Tasa de urbanización (rate of urbanization)

Estos dos porcentajes deberían estar en diferentes columnas. En esta sección, veremos uno de los caminos para separar los datos.

a. Uso de expresiones regulares

Expresión regular, también conocida como regex, es una secuencia de caracteres que forma un patrón de búsqueda, principalmente utilizada para la búsqueda de patrones de cadenas de caracteres u operaciones de sustituciones.

En este caso, vamos a usar el comando **str_extract_all()** (instalar librería "stringr"), el cual extrae cada patrón y lo ubica en un vector diferente por cada resultado.

Este comando utiliza dos argumentos:

- string = vector de caracteres que se trabajará
- pattern = patrón a buscar

"\d": Recoge todos los casos por separado que R reconoce como números. No reconoce palabras ni símbolos de porcentaje.

"\d"+: Recoge números adyacentes.

\d+\.|d: Recoge los casos con decimales

\d+\.|d\%: Recoge decimales y también porcentajes.

\d+\.|d)(?=\%): Recoge decimales, porcentajes pero elimina el símbolo %.

Veamos algunos ejemplos

Imaginemos que tenemos la siguiente información '25.3%,0% y 23.5% 13 34 hola 5 6 chau' y queremos extraer solo los números.

Para utilizar este camino, debemos instalar y utilizar la librería "string"

```
library(stringr)
# En este caso, le estamos pidiendo que nos traiga cada numero
str_extract_all(string = '25.3%,0% y 23.5% 13 34 hola 5 6 chau 200',pattern="\\d")
```

```
## [[1]]
##  [1] "2" "5" "3" "0" "2" "3" "5" "1" "3" "3" "4" "5" "6" "2" "0" "0"
```

```
# me trae números adyacentes:
str_extract_all('25.3%,0% y 23.5% 13 34 hola 5 6 chau', pattern = "\\d+") # + es al menos 1
```

```
## [[1]]
## [1] "25" "3"  "0"  "23" "5"  "13" "34" "5"  "6"
```

```
# numero entero, seguido opcionalmente de punto, más número de una o más cifras.
str_extract_all('25.3%,0% y 23.5% 13 34 hola 5 6 chau', pattern="\\d+\\.*\\d*")
```

```
## [[1]]
## [1] "25.3" "0"    "23.5" "13"   "34"   "5"    "6"
```

```
# numero entero, seguido opcionalmente de punto, más número de una o más cifras, seguido de %.
str_extract_all('25.3%,0% y 23.5% 13 34 hola 5 6 chau',pattern = "\\d+\\.*\\d*\\%")
```

```
## [[1]]
## [1] "25.3%" "0%"    "23.5%"
```

```
# porcentaje sin el simbolo

#  que antes de (?=\\%) haya (\\d+\\.*\\d*)

str_extract_all('25.3%,0% y 23.5% 13 34 hola 5 6 chau',pattern = "(\\d+\\.*\\d*)(?=\\%)")
```

```
## [[1]]
## [1] "25.3" "0"    "23.5"
```

```
# porcentaje sin el símbolo pero con signo negativo, si hubiese

str_extract_all('25.3%,0% y -23.5% 13 34 hola', pattern = "(\\-*\\d+\\.*\\d*)(?=\\%)")
```

```
## [[1]]
## [1] "25.3"  "0"     "-23.5"
```

Finalizando con [[1]][1]: permite visualizar un valor específico.

```
# primer valor es
str_extract_all('25%, 0% y 23.5%',
                "(\\-*\\d+\\.*\\d*)(?=\\%)")[[1]][1]
```

```
## [1] "25"
```

```
# segundo valor es
str_extract_all('25%, 0% y 23.5% hola',
                "(\\-*\\d+\\.*\\d*)(?=\\%)")[[1]][2]
```

```
## [1] "0"
```

```
# tercer valor es
str_extract_all('25%, 0% y 23.5% fwFRWFWF',
                "(\\-*\\d+\\.*\\d*)(?=\\%)")[[1]][3]
```

```
## [1] "23.5"
```

Ahora, apliquemos lo aprendido a nuestra columna de urbanización. Recordemos lo que tenemos:

```
# fila 1, columna 2
urban[4,2]
```

```
## [1] "\n      \n        urban population: 92% of total population (2008) \n      rate of urbanization
```

```
str_extract_all(urban$URBANIZATION,"(\\-*\\d+\\.*\\d*)(?=\\%)")
```

La estructura obtenida no facilitaría el trabajo de producir dos columnas (porque es una lista de vectores). El usar simplify lo hace mas fácil.

str_extract_all: Finalizando con simplify = T : permite visualizar cada columna de información por separado.

```
str_extract_all(urban$URBANIZATION,
                "(\\-*\\d+\\.*\\d*)(?=\\%)",
                simplify = T)
```

Esa matriz anterior me permite acceder a cada columna:

```
# Creamos objetos
PATRON="(\\-*\\d+\\.*\\d*)(?=\\%)"
COLSUCIA=urban$URBANIZATION

# UNA COLUMNA
urban$pop_urb=str_extract_all(string = COLSUCIA,
                              pattern= PATRON,
                              simplify = T)[,1]

# OTRA COLUMNA
urban$rate_urb=str_extract_all(string = COLSUCIA,
                               pattern=PATRON,
                               simplify = T)[,2]
```

Veamos, ejecutemos en la consola: head(urban[,-2])

```
head(urban[-2])
```

```
##          COUNTRY pop_urb rate_urb
## 1    Afghanistan      24      5.4
## 2        Albania      47      1.9
## 3        Algeria      65      2.5
## 4 American Samoa      92      2.4
## 5        Andorra      89     -0.2
## 6         Angola      57      4.4
```

b. Uso de particiones

```
# Recordamos:
urban[1,2]
```

```
## [1] "\n      \n        urban population: 24% of total population (2008) \n      rate of urbanization
```

El comando **str_split** nos permite separar un string en varias piezas.

str_split(string, pattern, simplify = T/F)

Aqui busco un texto que me permita dividir esa cadena de texto:

```
str_split(string = urban$URBANIZATION,
          pattern = 'rate of urbanization: ')
```

Como podemos ver, se utiliza el patrón como punto de división. En un lado, quedan los valores previos al patrón y, en el otro lado, los valores posteriores.

A pesar de que aún esta “sucio”, vamos a crear dos columnas.

```
urban$pop_urb2=str_split(string = urban$URBANIZATION,
                         pattern = 'rate of urbanization:',
                         simplify = T)[,1]
```

```
urban$rate_urb2=str_split(string = urban$URBANIZATION,
                          pattern = 'rate of urbanization:',
                          simplify = T)[,2]
```

Ahora limpiamos la primera columna. Para ello, vamos a repetir el proceso

Si una celda luce así:

```
urban$pop_urb2[1]
```

```
## [1] "\n      \n        urban population: 24% of total population (2008) \n      "
```

Puedo tener mejor texto si la divido y me quedo con el primer elemento:

```
str_split(urban$pop_urb2, '% of total', simplify = T)
```

```
##        [,1]                                  [,2]
##   [1,] "\n      \n        urban population: 24"   " population (2008) \n      "
##   [2,] "\n      \n        urban population: 47"   " population (2008) \n      "
##   [3,] "\n      \n        urban population: 65"   " population (2008) \n      "
##   [4,] "\n      \n        urban population: 92"   " population (2008) \n      "
##   [5,] "\n      \n        urban population: 89"   " population (2008) \n      "
##   [6,] "\n      \n        urban population: 57"   " population (2008) \n      "
##   [7,] "\n      \n        urban population: 100"  " population (2008) \n      "
##   [8,] "\n      \n        urban population: 30"   " population (2008) \n      "
##   [9,] "\n      \n        urban population: 92"   " population (2008) \n      "
##  [10,] "\n      \n        urban population: 64"   " population (2008) \n      "
##  [11,] "\n      \n        urban population: 47"   " population (2008) \n      "
##  [12,] "\n      \n        urban population: 89"   " population (2008) \n      "
##  [13,] "\n      \n        urban population: 67"   " population (2008) \n      "
##  [14,] "\n      \n        urban population: 52"   " population (2008) \n      "
##  [15,] "\n      \n        urban population: 84"   " population (2008) \n      "
##  [16,] "\n      \n        urban population: 89"   " population (2008) \n      "
##  [17,] "\n      \n        urban population: 27"   " population (2008) \n      "
##  [18,] "\n      \n        urban population: 40"   " population (2008) \n      "
##  [19,] "\n      \n        urban population: 73"   " population (2008) \n      "
##  [20,] "\n      \n        urban population: 97"   " population (2008) \n      "
##  [21,] "\n      \n        urban population: 52"   " population (2008) \n      "
##  [22,] "\n      \n        urban population: 41"   " population (2008) \n      "
##  [23,] "\n      \n        urban population: 100"  " population (2008) \n      "
##  [24,] "\n      \n        urban population: 35"   " population (2008) \n      "
##  [25,] "\n      \n        urban population: 66"   " population (2008) \n      "
##  [26,] "\n      \n        urban population: 47"   " population (2008) \n      "
##  [27,] "\n      \n        urban population: 60"   " population (2008) \n      "
##  [28,] "\n      \n        urban population: 86"   " population (2008) \n      "
##  [29,] "\n      \n        urban population: 40"   " population (2008) \n      "
##  [30,] "\n      \n        urban population: 75"   " population (2008) \n      "
##  [31,] "\n      \n        urban population: 71"   " population (2008) \n      "
##  [32,] "\n      \n        urban population: 20"   " population (2008) \n      "
##  [33,] "\n      \n        urban population: 33"   " population (2008) \n      "
##  [34,] "\n      \n        urban population: 10"   " population (2008) \n      "
##  [35,] "\n      \n        urban population: 22"   " population (2008) \n      "
```

```
##  [36,] "\n      \n        urban population: 57"   " population (2008) \n      "
##  [37,] "\n      \n        urban population: 80"   " population (2008) \n      "
##  [38,] "\n      \n        urban population: 60"   " population (2008) \n      "
##  [39,] "\n      \n        urban population: 100"  " population (2008) \n      "
##  [40,] "\n      \n        urban population: 39"   " population (2008) \n      "
##  [41,] "\n      \n        urban population: 27"   " population (2008) \n      "
##  [42,] "\n      \n        urban population: 88"   " population (2008) \n      "
##  [43,] "\n      \n        urban population: 43"   " population (2008) \n      "
##  [44,] "\n      \n        urban population: 74"   " population (2008) \n      "
##  [45,] "\n      \n        urban population: 28"   " population (2008) \n      "
##  [46,] "\n      \n        urban population: 34"   " population (2008) \n      "
##  [47,] "\n      \n        urban population: 61"   " population (2008) \n      "
##  [48,] "\n      \n        urban population: 74"   " population (2008) \n      "
##  [49,] "\n      \n        urban population: 63"   " population (2008) \n      "
##  [50,] "\n      \n        urban population: 49"   " population (2008) \n      "
##  [51,] "\n      \n        urban population: 57"   " population (2008) \n      "
##  [52,] "\n      \n        urban population: 76"   " population (2008) \n      "
##  [53,] "\n      \n        urban population: 70"   " population (2008) \n      "
##  [54,] "\n      \n        urban population: 73"   " population (2008) \n      "
##  [55,] "\n      \n        urban population: 87"   " population (2008) \n      "
##  [56,] "\n      \n        urban population: 87"   " population (2008) \n      "
##  [57,] "\n      \n        urban population: 74"   " population (2008) \n      "
##  [58,] "\n      \n        urban population: 69"   " population (2008) \n      "
##  [59,] "\n      \n        urban population: 66"   " population (2008) \n      "
##  [60,] "\n      \n        urban population: 43"   " population (2008) \n      "
##  [61,] "\n      \n        urban population: 61"   " population (2008) \n      "
##  [62,] "\n      \n        urban population: 39"   " population (2008) \n      "
##  [63,] "\n      \n        urban population: 21"   " population (2008) \n      "
##  [64,] "\n      \n        urban population: 69"   " population (2008) \n      "
##  [65,] "\n      \n        urban population: 17"   " population (2008) \n      "
##  [66,] "\n      \n        urban population: 92"   " population (2008) \n      "
##  [67,] "\n      \n        urban population: 41"   " population (2008) \n      "
##  [68,] "\n      \n        urban population: 52"   " population (2008) \n      "
##  [69,] "\n      \n        urban population: 63"   " population (2008) \n      "
##  [70,] "\n      \n        urban population: 77"   " population (2008) \n      "
##  [71,] "\n      \n        urban population: 52"   " population (2008) \n      "
##  [72,] "\n      \n        urban population: 85"   " population (2008) \n      "
##  [73,] "\n      \n        urban population: 57"   " population (2008) \n      "
##  [74,] "\n      \n        urban population: 72"   " population (2008) \n      "
##  [75,] "\n      \n        urban population: 53"   " population (2008) \n      "
##  [76,] "\n      \n        urban population: 74"   " population (2008) \n      "
##  [77,] "\n      \n        urban population: 50"   " population (2008) \n      "
##  [78,] "\n      \n        urban population: 100"  " population (2008) \n      "
##  [79,] "\n      \n        urban population: 61"   " population (2008) \n      "
##  [80,] "\n      \n        urban population: 84"   " population (2008) \n      "
##  [81,] "\n      \n        urban population: 31"   " population (2008) \n      "
##  [82,] "\n      \n        urban population: 93"   " population (2008) \n      "
##  [83,] "\n      \n        urban population: 49"   " population (2008) \n      "
##  [84,] "\n      \n        urban population: 31"   " population (2008) \n      "
##  [85,] "\n      \n        urban population: 34"   " population (2008) \n      "
##  [86,] "\n      \n        urban population: 30"   " population (2008) \n      "
##  [87,] "\n      \n        urban population: 28"   " population (2008) \n      "
##  [88,] "\n      \n        urban population: 47"   " population (2008) \n      "
##  [89,] "\n      \n        urban population: 100"  " population (2008) \n      "
```

```
##  [90,] "\n      \n        urban population: 48"   " population (2008) \n      "
##  [91,] "\n      \n        urban population: 100"  " population (2008) \n      "
##  [92,] "\n      \n        urban population: 68"   " population (2008) \n      "
##  [93,] "\n      \n        urban population: 92"   " population (2008) \n      "
##  [94,] "\n      \n        urban population: 29"   " population (2008) \n      "
##  [95,] "\n      \n        urban population: 52"   " population (2008) \n      "
##  [96,] "\n      \n        urban population: 68"   " population (2008) \n      "
##  [97,] "\n      \n        urban population: 67"   " population (2008) \n      "
##  [98,] "\n      \n        urban population: 61"   " population (2008) \n      "
##  [99,] "\n      \n        urban population: 51"   " population (2008) \n      "
## [100,] "\n      \n        urban population: 92"   " population (2008) \n      "
## [101,] "\n      \n        urban population: 68"   " population (2008) \n      "
## [102,] "\n      \n        urban population: 53"   " population (2008) \n      "
## [103,] "\n      \n        urban population: 66"   " population (2008) \n      "
## [104,] "\n      \n        urban population: 31"   " population (2008) \n      "
## [105,] "\n      \n        urban population: 78"   " population (2008) \n      "
## [106,] "\n      \n        urban population: 58"   " population (2008) \n      "
## [107,] "\n      \n        urban population: 22"   " population (2008) \n      "
## [108,] "\n      \n        urban population: 44"   " population (2008) \n      "
## [109,] "\n      \n        urban population: 63"   " population (2008) \n      "
## [110,] "\n      \n        urban population: 81"   " population (2008) \n      "
## [111,] "\n      \n        urban population: 98"   " population (2008) \n      "
## [112,] "\n      \n        urban population: 36"   " population (2008) \n      "
## [113,] "\n      \n        urban population: 31"   " population (2008) \n      "
## [114,] "\n      \n        urban population: 68"   " population (2008) \n      "
## [115,] "\n      \n        urban population: 87"   " population (2008) \n      "
## [116,] "\n      \n        urban population: 25"   " population (2008) \n      "
## [117,] "\n      \n        urban population: 60"   " population (2008) \n      "
## [118,] "\n      \n        urban population: 78"   " population (2008) \n      "
## [119,] "\n      \n        urban population: 14"   " population (2008) \n      "
## [120,] "\n      \n        urban population: 67"   " population (2008) \n      "
## [121,] "\n      \n        urban population: 82"   " population (2008) \n      "
## [122,] "\n      \n        urban population: 100"  " population (2008) \n      "
## [123,] "\n      \n        urban population: 67"   " population (2008) \n      "
## [124,] "\n      \n        urban population: 29"   " population (2008) \n      "
## [125,] "\n      \n        urban population: 19"   " population (2008) \n      "
## [126,] "\n      \n        urban population: 70"   " population (2008) \n      "
## [127,] "\n      \n        urban population: 38"   " population (2008) \n      "
## [128,] "\n      \n        urban population: 32"   " population (2008) \n      "
## [129,] "\n      \n        urban population: 94"   " population (2008) \n      "
## [130,] "\n      \n        urban population: 71"   " population (2008) \n      "
## [131,] "\n      \n        urban population: 41"   " population (2008) \n      "
## [132,] "\n      \n        urban population: 42"   " population (2008) \n      "
## [133,] "\n      \n        urban population: 77"   " population (2008) \n      "
## [134,] "\n      \n        urban population: 22"   " population (2008) \n      "
## [135,] "\n      \n        urban population: 42"   " population (2008) \n      "
## [136,] "\n      \n        urban population: 100"  " population (2008) \n      "
## [137,] "\n      \n        urban population: 57"   " population (2008) \n      "
## [138,] "\n      \n        urban population: 60"   " population (2008) \n      "
## [139,] "\n      \n        urban population: 14"   " population (2008) \n      "
## [140,] "\n      \n        urban population: 56"   " population (2008) \n      "
## [141,] "\n      \n        urban population: 37"   " population (2008) \n      "
## [142,] "\n      \n        urban population: 37"   " population (2008) \n      "
## [143,] "\n      \n        urban population: 100"  " population (2008) \n      "
```

```
## [144,] "\n      \n        urban population: 17"   " population (2008) \n      "
## [145,] "\n      \n        urban population: 82"   " population (2008) \n      "
## [146,] "\n      \n        urban population: 93"   " population (2008) \n      "
## [147,] "\n      \n        urban population: 65"   " population (2008) \n      "
## [148,] "\n      \n        urban population: 87"   " population (2008) \n      "
## [149,] "\n      \n        urban population: 57"   " population (2008) \n      "
## [150,] "\n      \n        urban population: 16"   " population (2008) \n      "
## [151,] "\n      \n        urban population: 48"   " population (2008) \n      "
## [152,] "\n      \n        urban population: 39"   " population (2008) \n      "
## [153,] "\n      \n        urban population: 91"   " population (2008) \n      "
## [154,] "\n      \n        urban population: 77"   " population (2008) \n      "
## [155,] "\n      \n        urban population: 72"   " population (2008) \n      "
## [156,] "\n      \n        urban population: 36"   " population (2008) \n      "
## [157,] "\n      \n        urban population: 81"   " population (2008) \n      "
## [158,] "\n      \n        urban population: 73"   " population (2008) \n      "
## [159,] "\n      \n        urban population: 12"   " population (2008) \n      "
## [160,] "\n      \n        urban population: 60"   " population (2008) \n      "
## [161,] "\n      \n        urban population: 71"   " population (2008) \n      "
## [162,] "\n      \n        urban population: 65"   " population (2008) \n      "
## [163,] "\n      \n        urban population: 0"    " population (2008) \n      "
## [164,] "\n      \n        urban population: 61"   " population (2008) \n      "
## [165,] "\n      \n        urban population: 59"   " population (2008) \n      "
## [166,] "\n      \n        urban population: 98"   " population (2008) \n      "
## [167,] "\n      \n        urban population: 96"   " population (2008) \n      "
## [168,] "\n      \n        urban population: 54"   " population (2008) \n      "
## [169,] "\n      \n        urban population: 73"   " population (2008) \n      "
## [170,] "\n      \n        urban population: 18"   " population (2008) \n      "
## [171,] "\n      \n        urban population: 39"   " population (2008) \n      "
## [172,] "\n      \n        urban population: 32"   " population (2008) \n      "
## [173,] "\n      \n        urban population: 28"   " population (2008) \n      "
## [174,] "\n      \n        urban population: 89"   " population (2008) \n      "
## [175,] "\n      \n        urban population: 47"   " population (2008) \n      "
## [176,] "\n      \n        urban population: 23"   " population (2008) \n      "
## [177,] "\n      \n        urban population: 94"   " population (2008) \n      "
## [178,] "\n      \n        urban population: 61"   " population (2008) \n      "
## [179,] "\n      \n        urban population: 82"   " population (2008) \n      "
## [180,] "\n      \n        urban population: 42"   " population (2008) \n      "
## [181,] "\n      \n        urban population: 52"   " population (2008) \n      "
## [182,] "\n      \n        urban population: 54"   " population (2008) \n      "
## [183,] "\n      \n        urban population: 38"   " population (2008) \n      "
## [184,] "\n      \n        urban population: 100"  " population (2008) \n      "
## [185,] "\n      \n        urban population: 56"   " population (2008) \n      "
## [186,] "\n      \n        urban population: 48"   " population (2008) \n      "
## [187,] "\n      \n        urban population: 18"   " population (2008) \n      "
## [188,] "\n      \n        urban population: 37"   " population (2008) \n      "
## [189,] "\n      \n        urban population: 61"   " population (2008) \n      "
## [190,] "\n      \n        urban population: 77"   " population (2008) \n      "
## [191,] "\n      \n        urban population: 15"   " population (2008) \n      "
## [192,] "\n      \n        urban population: 43"   " population (2008) \n      "
## [193,] "\n      \n        urban population: 75"   " population (2008) \n      "
## [194,] "\n      \n        urban population: 25"   " population (2008) \n      "
## [195,] "\n      \n        urban population: 85"   " population (2008) \n      "
## [196,] "\n      \n        urban population: 73"   " population (2008) \n      "
## [197,] "\n      \n        urban population: 54"   " population (2008) \n      "
```

```
## [198,] "\n      \n        urban population: 26"   " population (2008) \n      "
## [199,] "\n      \n        urban population: 25"   " population (2008) \n      "
## [200,] "\n      \n        urban population: 33"   " population (2008) \n      "
## [201,] "\n      \n        urban population: 27"   " population (2008) \n      "
## [202,] "\n      \n        urban population: 42"   " population (2008) \n      "
## [203,] "\n      \n        urban population: 0"    " population (2008) \n      "
## [204,] "\n      \n        urban population: 25"   " population (2008) \n      "
## [205,] "\n      \n        urban population: 13"   " population (2008) \n      "
## [206,] "\n      \n        urban population: 67"   " population (2008) \n      "
## [207,] "\n      \n        urban population: 69"   " population (2008) \n      "
## [208,] "\n      \n        urban population: 49"   " population (2008) \n      "
## [209,] "\n      \n        urban population: 92"   " population (2008) \n      "
## [210,] "\n      \n        urban population: 49"   " population (2008) \n      "
## [211,] "\n      \n        urban population: 13"   " population (2008) \n      "
## [212,] "\n      \n        urban population: 68"   " population (2008) \n      "
## [213,] "\n      \n        urban population: 78"   " population (2008) \n      "
## [214,] "\n      \n        urban population: 90"   " population (2008) \n      "
## [215,] "\n      \n        urban population: 82"   " population (2008) \n      "
## [216,] "\n      \n        urban population: 92"   " population (2008) \n      "
## [217,] "\n      \n        urban population: 37"   " population (2008) \n      "
## [218,] "\n      \n        urban population: 25"   " population (2008) \n      "
## [219,] "\n      \n        urban population: 93"   " population (2008) \n      "
## [220,] "\n      \n        urban population: 28"   " population (2008) \n      "
## [221,] "\n      \n        urban population: 95"   " population (2008) \n      "
## [222,] "\n      \n        urban population: 0"    " population (2008) \n      "
## [223,] "\n      \n        urban population: 72"   " population (2008) \n      "
## [224,] "\n      \n        urban population: 81"   " population (2008) \n      "
## [225,] "\n      \n        urban population: 48.6" " population (2005) \n      "
## [226,] "\n      \n        urban population: 31"   " population (2008) \n      "
## [227,] "\n      \n        urban population: 35"   " population (2008) \n      "
## [228,] "\n      \n        urban population: 37"   " population (2008) \n      "
```

```
urban$pop_urb2=str_split(urban$pop_urb2,
                         '% of total',
                         simplify = T)[,1]
```

Entonces, ahora tenemos:

```
urban$pop_urb2[1]
```

```
## [1] "\n      \n        urban population: 24"
```

Uso otro simbolo divisor y, en este caso,me quedo con la parte 2:

```
str_split(urban$pop_urb2,':',simplify = T)
```

```
##        [,1]                                [,2]
##   [1,] "\n      \n        urban population" " 24"
##   [2,] "\n      \n        urban population" " 47"
##   [3,] "\n      \n        urban population" " 65"
##   [4,] "\n      \n        urban population" " 92"
##   [5,] "\n      \n        urban population" " 89"
```

```
##   [6,] "\n      \n        urban population" " 57"
##   [7,] "\n      \n        urban population" " 100"
##   [8,] "\n      \n        urban population" " 30"
##   [9,] "\n      \n        urban population" " 92"
##  [10,] "\n      \n        urban population" " 64"
##  [11,] "\n      \n        urban population" " 47"
##  [12,] "\n      \n        urban population" " 89"
##  [13,] "\n      \n        urban population" " 67"
##  [14,] "\n      \n        urban population" " 52"
##  [15,] "\n      \n        urban population" " 84"
##  [16,] "\n      \n        urban population" " 89"
##  [17,] "\n      \n        urban population" " 27"
##  [18,] "\n      \n        urban population" " 40"
##  [19,] "\n      \n        urban population" " 73"
##  [20,] "\n      \n        urban population" " 97"
##  [21,] "\n      \n        urban population" " 52"
##  [22,] "\n      \n        urban population" " 41"
##  [23,] "\n      \n        urban population" " 100"
##  [24,] "\n      \n        urban population" " 35"
##  [25,] "\n      \n        urban population" " 66"
##  [26,] "\n      \n        urban population" " 47"
##  [27,] "\n      \n        urban population" " 60"
##  [28,] "\n      \n        urban population" " 86"
##  [29,] "\n      \n        urban population" " 40"
##  [30,] "\n      \n        urban population" " 75"
##  [31,] "\n      \n        urban population" " 71"
##  [32,] "\n      \n        urban population" " 20"
##  [33,] "\n      \n        urban population" " 33"
##  [34,] "\n      \n        urban population" " 10"
##  [35,] "\n      \n        urban population" " 22"
##  [36,] "\n      \n        urban population" " 57"
##  [37,] "\n      \n        urban population" " 80"
##  [38,] "\n      \n        urban population" " 60"
##  [39,] "\n      \n        urban population" " 100"
##  [40,] "\n      \n        urban population" " 39"
##  [41,] "\n      \n        urban population" " 27"
##  [42,] "\n      \n        urban population" " 88"
##  [43,] "\n      \n        urban population" " 43"
##  [44,] "\n      \n        urban population" " 74"
##  [45,] "\n      \n        urban population" " 28"
##  [46,] "\n      \n        urban population" " 34"
##  [47,] "\n      \n        urban population" " 61"
##  [48,] "\n      \n        urban population" " 74"
##  [49,] "\n      \n        urban population" " 63"
##  [50,] "\n      \n        urban population" " 49"
##  [51,] "\n      \n        urban population" " 57"
##  [52,] "\n      \n        urban population" " 76"
##  [53,] "\n      \n        urban population" " 70"
##  [54,] "\n      \n        urban population" " 73"
##  [55,] "\n      \n        urban population" " 87"
##  [56,] "\n      \n        urban population" " 87"
##  [57,] "\n      \n        urban population" " 74"
##  [58,] "\n      \n        urban population" " 69"
##  [59,] "\n      \n        urban population" " 66"
```

```
##  [60,] "\n      \n        urban population" " 43"
##  [61,] "\n      \n        urban population" " 61"
##  [62,] "\n      \n        urban population" " 39"
##  [63,] "\n      \n        urban population" " 21"
##  [64,] "\n      \n        urban population" " 69"
##  [65,] "\n      \n        urban population" " 17"
##  [66,] "\n      \n        urban population" " 92"
##  [67,] "\n      \n        urban population" " 41"
##  [68,] "\n      \n        urban population" " 52"
##  [69,] "\n      \n        urban population" " 63"
##  [70,] "\n      \n        urban population" " 77"
##  [71,] "\n      \n        urban population" " 52"
##  [72,] "\n      \n        urban population" " 85"
##  [73,] "\n      \n        urban population" " 57"
##  [74,] "\n      \n        urban population" " 72"
##  [75,] "\n      \n        urban population" " 53"
##  [76,] "\n      \n        urban population" " 74"
##  [77,] "\n      \n        urban population" " 50"
##  [78,] "\n      \n        urban population" " 100"
##  [79,] "\n      \n        urban population" " 61"
##  [80,] "\n      \n        urban population" " 84"
##  [81,] "\n      \n        urban population" " 31"
##  [82,] "\n      \n        urban population" " 93"
##  [83,] "\n      \n        urban population" " 49"
##  [84,] "\n      \n        urban population" " 31"
##  [85,] "\n      \n        urban population" " 34"
##  [86,] "\n      \n        urban population" " 30"
##  [87,] "\n      \n        urban population" " 28"
##  [88,] "\n      \n        urban population" " 47"
##  [89,] "\n      \n        urban population" " 100"
##  [90,] "\n      \n        urban population" " 48"
##  [91,] "\n      \n        urban population" " 100"
##  [92,] "\n      \n        urban population" " 68"
##  [93,] "\n      \n        urban population" " 92"
##  [94,] "\n      \n        urban population" " 29"
##  [95,] "\n      \n        urban population" " 52"
##  [96,] "\n      \n        urban population" " 68"
##  [97,] "\n      \n        urban population" " 67"
##  [98,] "\n      \n        urban population" " 61"
##  [99,] "\n      \n        urban population" " 51"
## [100,] "\n      \n        urban population" " 92"
## [101,] "\n      \n        urban population" " 68"
## [102,] "\n      \n        urban population" " 53"
## [103,] "\n      \n        urban population" " 66"
## [104,] "\n      \n        urban population" " 31"
## [105,] "\n      \n        urban population" " 78"
## [106,] "\n      \n        urban population" " 58"
## [107,] "\n      \n        urban population" " 22"
## [108,] "\n      \n        urban population" " 44"
## [109,] "\n      \n        urban population" " 63"
## [110,] "\n      \n        urban population" " 81"
## [111,] "\n      \n        urban population" " 98"
## [112,] "\n      \n        urban population" " 36"
## [113,] "\n      \n        urban population" " 31"
```

```
## [114,] "\n      \n        urban population" " 68"
## [115,] "\n      \n        urban population" " 87"
## [116,] "\n      \n        urban population" " 25"
## [117,] "\n      \n        urban population" " 60"
## [118,] "\n      \n        urban population" " 78"
## [119,] "\n      \n        urban population" " 14"
## [120,] "\n      \n        urban population" " 67"
## [121,] "\n      \n        urban population" " 82"
## [122,] "\n      \n        urban population" " 100"
## [123,] "\n      \n        urban population" " 67"
## [124,] "\n      \n        urban population" " 29"
## [125,] "\n      \n        urban population" " 19"
## [126,] "\n      \n        urban population" " 70"
## [127,] "\n      \n        urban population" " 38"
## [128,] "\n      \n        urban population" " 32"
## [129,] "\n      \n        urban population" " 94"
## [130,] "\n      \n        urban population" " 71"
## [131,] "\n      \n        urban population" " 41"
## [132,] "\n      \n        urban population" " 42"
## [133,] "\n      \n        urban population" " 77"
## [134,] "\n      \n        urban population" " 22"
## [135,] "\n      \n        urban population" " 42"
## [136,] "\n      \n        urban population" " 100"
## [137,] "\n      \n        urban population" " 57"
## [138,] "\n      \n        urban population" " 60"
## [139,] "\n      \n        urban population" " 14"
## [140,] "\n      \n        urban population" " 56"
## [141,] "\n      \n        urban population" " 37"
## [142,] "\n      \n        urban population" " 37"
## [143,] "\n      \n        urban population" " 100"
## [144,] "\n      \n        urban population" " 17"
## [145,] "\n      \n        urban population" " 82"
## [146,] "\n      \n        urban population" " 93"
## [147,] "\n      \n        urban population" " 65"
## [148,] "\n      \n        urban population" " 87"
## [149,] "\n      \n        urban population" " 57"
## [150,] "\n      \n        urban population" " 16"
## [151,] "\n      \n        urban population" " 48"
## [152,] "\n      \n        urban population" " 39"
## [153,] "\n      \n        urban population" " 91"
## [154,] "\n      \n        urban population" " 77"
## [155,] "\n      \n        urban population" " 72"
## [156,] "\n      \n        urban population" " 36"
## [157,] "\n      \n        urban population" " 81"
## [158,] "\n      \n        urban population" " 73"
## [159,] "\n      \n        urban population" " 12"
## [160,] "\n      \n        urban population" " 60"
## [161,] "\n      \n        urban population" " 71"
## [162,] "\n      \n        urban population" " 65"
## [163,] "\n      \n        urban population" " 0"
## [164,] "\n      \n        urban population" " 61"
## [165,] "\n      \n        urban population" " 59"
## [166,] "\n      \n        urban population" " 98"
## [167,] "\n      \n        urban population" " 96"
```

```
## [168,] "\n      \n        urban population" " 54"
## [169,] "\n      \n        urban population" " 73"
## [170,] "\n      \n        urban population" " 18"
## [171,] "\n      \n        urban population" " 39"
## [172,] "\n      \n        urban population" " 32"
## [173,] "\n      \n        urban population" " 28"
## [174,] "\n      \n        urban population" " 89"
## [175,] "\n      \n        urban population" " 47"
## [176,] "\n      \n        urban population" " 23"
## [177,] "\n      \n        urban population" " 94"
## [178,] "\n      \n        urban population" " 61"
## [179,] "\n      \n        urban population" " 82"
## [180,] "\n      \n        urban population" " 42"
## [181,] "\n      \n        urban population" " 52"
## [182,] "\n      \n        urban population" " 54"
## [183,] "\n      \n        urban population" " 38"
## [184,] "\n      \n        urban population" " 100"
## [185,] "\n      \n        urban population" " 56"
## [186,] "\n      \n        urban population" " 48"
## [187,] "\n      \n        urban population" " 18"
## [188,] "\n      \n        urban population" " 37"
## [189,] "\n      \n        urban population" " 61"
## [190,] "\n      \n        urban population" " 77"
## [191,] "\n      \n        urban population" " 15"
## [192,] "\n      \n        urban population" " 43"
## [193,] "\n      \n        urban population" " 75"
## [194,] "\n      \n        urban population" " 25"
## [195,] "\n      \n        urban population" " 85"
## [196,] "\n      \n        urban population" " 73"
## [197,] "\n      \n        urban population" " 54"
## [198,] "\n      \n        urban population" " 26"
## [199,] "\n      \n        urban population" " 25"
## [200,] "\n      \n        urban population" " 33"
## [201,] "\n      \n        urban population" " 27"
## [202,] "\n      \n        urban population" " 42"
## [203,] "\n      \n        urban population" " 0"
## [204,] "\n      \n        urban population" " 25"
## [205,] "\n      \n        urban population" " 13"
## [206,] "\n      \n        urban population" " 67"
## [207,] "\n      \n        urban population" " 69"
## [208,] "\n      \n        urban population" " 49"
## [209,] "\n      \n        urban population" " 92"
## [210,] "\n      \n        urban population" " 49"
## [211,] "\n      \n        urban population" " 13"
## [212,] "\n      \n        urban population" " 68"
## [213,] "\n      \n        urban population" " 78"
## [214,] "\n      \n        urban population" " 90"
## [215,] "\n      \n        urban population" " 82"
## [216,] "\n      \n        urban population" " 92"
## [217,] "\n      \n        urban population" " 37"
## [218,] "\n      \n        urban population" " 25"
## [219,] "\n      \n        urban population" " 93"
## [220,] "\n      \n        urban population" " 28"
## [221,] "\n      \n        urban population" " 95"
```

```
## [222,] "\n      \n        urban population" " 0"
## [223,] "\n      \n        urban population" " 72"
## [224,] "\n      \n        urban population" " 81"
## [225,] "\n      \n        urban population" " 48.6"
## [226,] "\n      \n        urban population" " 31"
## [227,] "\n      \n        urban population" " 35"
## [228,] "\n      \n        urban population" " 37"
```

```
urban$pop_urb2=str_split(urban$pop_urb2,
                         ':',
                         simplify = T)[,2]
```

Luego tengo:

```
urban$pop_urb2[1]
```

```
## [1] " 24"
```

Si sigo la misma estrategia para la otra columna:

```
urban$rate_urb2[1]
```

```
## [1] " 5.4% annual rate of change (2005-10 est.) \n        "
```

Veo que puede ser trivial:

```
urban$rate_urb2=str_split(urban$rate_urb2,
                          pattern = '%',
                          simplify = T)[,1]
```

```
urban$rate_urb2[1]
```

```
## [1] " 5.4"
```

Veamos. Ejecutemos en la consola: head(urban[,-2])

OTRO EJEMPLO

¿Recuerdan la página web de películas? Utilicemos *rvest* para extraer la información. Ya conocemos esos pasos por la sesión pasada:

```
library(rvest)
url="https://www.filmaffinity.com/es/ranking.php?rn=ranking_fa_movies"
pagina_web=read_html(url)
```

```
css_nombre="div.mc-title" # contenemos la clase CSS en un objeto
nombre_html <- html_nodes(pagina_web,css_nombre) # con html_nodes y html_text, obtenemos el código html
nombre_texto <- html_text(nombre_html)
head(nombre_texto) #vemos los datos
```

```
## [1] "El padrino  (1972)   "              "El padrino. Parte II  (1974)   "
## [3] "Doce hombres sin piedad  (1957)   " "La lista de Schindler  (1993)   "
## [5] "Testigo de cargo  (1957)   "        "Luces de la ciudad  (1931)   "
```

```
css_director="div.mc-director" # contenemos la clase CSS en un objeto
director_html <- html_nodes(pagina_web,css_director) # con html_nodes y html_text, obtenemos el código
director_texto <- html_text(director_html)
head(director_texto) #vemos los datos
```

```
## [1] "Francis Ford Coppola" "Francis Ford Coppola" "Sidney Lumet"
## [4] "Steven Spielberg"     "Billy Wilder"         "Charles Chaplin"
```

```
css_cast="div.mc-cast" # contenemos la clase CSS en un objeto
cast_html <- html_nodes(pagina_web,css_cast) # con html_nodes y html_text, obtenemos el código html que
cast_texto <- html_text(cast_html)
head(cast_texto) #vemos los datos
```

```
## [1] "Marlon Brando, Al Pacino, James Caan, Robert Duvall, Diane Keaton, John Cazale, Talia Shire, Ri
## [2] "Al Pacino, Robert De Niro, Diane Keaton, Robert Duvall, John Cazale, Lee Strasberg, Talia Shire
## [3] "Henry Fonda, Lee J. Cobb, Jack Warden, E.G. Marshall, Martin Balsam, Ed Begley, John Fiedler, R
## [4] "Liam Neeson, Ben Kingsley, Ralph Fiennes, Caroline Goodall, Jonathan Sagall, Embeth Davidtz, No
## [5] "Tyrone Power, Marlene Dietrich, Charles Laughton, Elsa Lanchester, John Williams, Una O'Connor,
## [6] "Charles Chaplin, Virginia Cherrill, Florence Lee, Harry Myers, Allan Garcia, Hank Mann, Jack Al
```

```
css_data="li.data" # contenemos la clase CSS en un objeto
data_html <- html_nodes(pagina_web,css_data) # con html_nodes y html_text, obtenemos el código html que
data_texto <- html_text(data_html)
head(data_texto) #vemos los datos
```

```
## [1] "\n                    9,0\n                    174.532 \n                "
## [2] "\n                    8,9\n                    139.291 \n                "
## [3] "\n                    8,7\n                    70.716 \n                "
## [4] "\n                    8,6\n                    174.827 \n                "
## [5] "\n                    8,6\n                    44.868 \n                "
## [6] "\n                    8,6\n                    32.927 \n                "
```

```
data_movies <- data.frame(NOMBRE = nombre_texto, DIRECTOR = director_texto, CAST = cast_texto, POINTS =
head(data_movies)
```

```
##                               NOMBRE             DIRECTOR
## 1              El padrino  (1972)    Francis Ford Coppola
## 2    El padrino. Parte II  (1974)    Francis Ford Coppola
## 3 Doce hombres sin piedad  (1957)            Sidney Lumet
## 4   La lista de Schindler  (1993)        Steven Spielberg
## 5        Testigo de cargo  (1957)            Billy Wilder
## 6      Luces de la ciudad  (1931)         Charles Chaplin
##
## 1               Marlon Brando, Al Pacino, James Caan, Robert Duvall, Diane Keaton, John Cazale, Talia
## 2                 Al Pacino, Robert De Niro, Diane Keaton, Robert Duvall, John Cazale, Lee Strasberg
## 3                      Henry Fonda, Lee J. Cobb, Jack Warden, E.G. Marshall, Martin Balsam, Ed Begley
## 4 Liam Neeson, Ben Kingsley, Ralph Fiennes, Caroline Goodall, Jonathan Sagall, Embeth Davidtz, Norbe
## 5      Tyrone Power, Marlene Dietrich, Charles Laughton, Elsa Lanchester, John Williams, Una O'Conno
## 6   Charles Chaplin, Virginia Cherrill, Florence Lee, Harry Myers, Allan Garcia, Hank Mann, Jack Ale
##                                                                    POINTS
## 1 \n                    9,0\n                    174.532 \n
```

```
## 2 \n                    8,9\n                  139.291 \n
## 3  \n                    8,7\n                   70.716 \n
## 4 \n                    8,6\n                  174.827 \n
## 5  \n                    8,6\n                   44.868 \n
## 6  \n                    8,6\n                   32.927 \n
```

Ok! Tenemos los datos. Hoy hemos aprendido algunas técnicas para limpiar esta información y separar la información que nos interesa en diferentes vectores. Saquemos el año del vector nombre:

```
data_movies$YEAR = str_extract_all(data_movies$NOMBRE,
                    "\\d+\\.*\\d*",
                    simplify = T)[,1]
```

Ya tenemos los años por separado. ¿Qué tal si quisieramos a las y los miembros del cast por separado:

```
data_movies[3]
```

```
##
## 1                                Marlon Brando, Al Pacino, James Caan, Robert Duvall, Diane Keaton,
## 2                                  Al Pacino, Robert De Niro, Diane Keaton, Robert Duvall, John Caza
## 3                                       Henry Fonda, Lee J. Cobb, Jack Warden, E.G. Marshall, Martin
## 4                     Liam Neeson, Ben Kingsley, Ralph Fiennes, Caroline Goodall, Jonathan Sagall, Embe
## 5                          Tyrone Power, Marlene Dietrich, Charles Laughton, Elsa Lanchester, John Will
## 6                       Charles Chaplin, Virginia Cherrill, Florence Lee, Harry Myers, Allan Garcia, Ha
## 7                                    Tim Robbins, Morgan Freeman, Bob Gunton, James Whitmore, Gil Bell
## 8                              Charles Chaplin, Paulette Goddard, Henry Bergman, Chester Conklin, Stanle
## 9       Charles Chaplin, Paulette Goddard, Jack Oakie, Reginald Gardiner, Henry Daniell, Carter De Ha
## 10                                  John Travolta, Samuel L. Jackson, Uma Thurman, Bruce Willis, Ving
## 11                                   Robert Redford, Paul Newman, Robert Shaw, Charles Durning, Ray Wa
## 12                                      Carole Lombard, Jack Benny, Robert Stack, Stanley Ridges, Feli
## 13                             Tatsuya Nakadai, Rentarô Mikuni, Akira Ishihama, Shima Iwashita, Tetsurô
## 14                                William Holden, Gloria Swanson, Erich von Stroheim, Nancy Olson, Lloyd
## 15                              Roberto Benigni, Nicoletta Braschi, Giorgio Cantarini, Marisa Paredes,
## 16                                    Bette Davis, Anne Baxter, George Sanders, Celeste Holm, Gary Merri
## 17                               Kirk Douglas, George Macready, Adolphe Menjou, Ralph Meeker, Wayne Mori
## 18                              Toshirô Mifune, Kyôko Kagawa, Yutaka Sada, Takashi Shimura, Tatsuya Miha
## 19                           Toshirô Mifune, Takashi Shimura, Yoshio Inaba, Seiji Miyaguchi, Minoru Chiak
## 20                                                                George O'Brien, Janet Gaynor, Marg
## 21                               Fred MacMurray, Barbara Stanwyck, Edward G. Robinson, Tom Powers, Porter
## 22                                    Jack Lemmon, Shirley MacLaine, Fred MacMurray, Ray Walston, Edie A
## 23 Alexandre Rodrigues, Leandro Firmino, Phellipe Haagensen, Douglas Silva, Seu Jorge, Jonathan Haage
## 24                                                     Charles Chaplin, Jackie Coogan, Edna Purviance
## 25                                                          Philippe Leroy, Marc Michel, Michel Cons
## 26                                    Anthony Perkins, Janet Leigh, John Gavin, Vera Miles, John McIntir
## 27                                                    Charles Chaplin, Mack Swain, Georgia Hale, Tom
## 28                                     Ray Liotta, Robert De Niro, Joe Pesci, Lorraine Bracco, Paul Sorv
## 29                                Humphrey Bogart, Ingrid Bergman, Paul Henreid, Claude Rains, Conrad Veid
## 30                             Jack Nicholson, Louise Fletcher, Brad Dourif, William Redfield, Mews Small,
```

Un forma de separarlos sería la coma, ¿Verdad?

```
data_movies$DIR1 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,1]

data_movies$DIR2 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,2]

data_movies$DIR3 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,3]

data_movies$DIR4 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,4]

data_movies$DIR5 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,5]

data_movies$DIR6 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,6]

data_movies$DIR7 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,7]

data_movies$DIR8 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,8]

data_movies$DIR9 = str_split(data_movies$CAST,
                             ', ',
                             simplify = T)[,9]
```

Vemos que hay hasta 9 en algunos casos. Separemos todo en columnas:

```
str_split(data_movies$CAST,
                             ', ',
                             simplify = T)
```

```
##       [,1]                [,2]                [,3]
##  [1,] "Marlon Brando"     "Al Pacino"         "James Caan"
##  [2,] "Al Pacino"         "Robert De Niro"    "Diane Keaton"
##  [3,] "Henry Fonda"       "Lee J. Cobb"       "Jack Warden"
##  [4,] "Liam Neeson"       "Ben Kingsley"      "Ralph Fiennes"
##  [5,] "Tyrone Power"      "Marlene Dietrich"  "Charles Laughton"
##  [6,] "Charles Chaplin"   "Virginia Cherrill" "Florence Lee"
##  [7,] "Tim Robbins"       "Morgan Freeman"    "Bob Gunton"
##  [8,] "Charles Chaplin"   "Paulette Goddard"  "Henry Bergman"
##  [9,] "Charles Chaplin"   "Paulette Goddard"  "Jack Oakie"
## [10,] "John Travolta"     "Samuel L. Jackson" "Uma Thurman"
```

```
## [11,] "Robert Redford"      "Paul Newman"       "Robert Shaw"
## [12,] "Carole Lombard"      "Jack Benny"        "Robert Stack"
## [13,] "Tatsuya Nakadai"     "Rentarô Mikuni"    "Akira Ishihama"
## [14,] "William Holden"      "Gloria Swanson"    "Erich von Stroheim"
## [15,] "Roberto Benigni"     "Nicoletta Braschi" "Giorgio Cantarini"
## [16,] "Bette Davis"         "Anne Baxter"       "George Sanders"
## [17,] "Kirk Douglas"        "George Macready"   "Adolphe Menjou"
## [18,] "Toshirô Mifune"      "Kyôko Kagawa"      "Yutaka Sada"
## [19,] "Toshirô Mifune"      "Takashi Shimura"   "Yoshio Inaba"
## [20,] "George O'Brien"      "Janet Gaynor"      "Margaret Livingston"
## [21,] "Fred MacMurray"      "Barbara Stanwyck"  "Edward G. Robinson"
## [22,] "Jack Lemmon"         "Shirley MacLaine"  "Fred MacMurray"
## [23,] "Alexandre Rodrigues" "Leandro Firmino"   "Phellipe Haagensen"
## [24,] "Charles Chaplin"     "Jackie Coogan"     "Edna Purviance"
## [25,] "Philippe Leroy"      "Marc Michel"       "Michel Constantin"
## [26,] "Anthony Perkins"     "Janet Leigh"       "John Gavin"
## [27,] "Charles Chaplin"     "Mack Swain"        "Georgia Hale"
## [28,] "Ray Liotta"          "Robert De Niro"    "Joe Pesci"
## [29,] "Humphrey Bogart"     "Ingrid Bergman"    "Paul Henreid"
## [30,] "Jack Nicholson"      "Louise Fletcher"   "Brad Dourif"
##       [,4]                  [,5]                 [,6]
##  [1,] "Robert Duvall"       "Diane Keaton"       "John Cazale"
##  [2,] "Robert Duvall"       "John Cazale"        "Lee Strasberg"
##  [3,] "E.G. Marshall"       "Martin Balsam"      "Ed Begley"
##  [4,] "Caroline Goodall"    "Jonathan Sagall"    "Embeth Davidtz"
##  [5,] "Elsa Lanchester"     "John Williams"      "Una O'Connor"
##  [6,] "Harry Myers"         "Allan Garcia"       "Hank Mann"
##  [7,] "James Whitmore"      "Gil Bellows"        "William Sadler"
##  [8,] "Chester Conklin"     "Stanley Stanford"   "Hank Mann"
##  [9,] "Reginald Gardiner"   "Henry Daniell"      "Carter De Haven"
## [10,] "Bruce Willis"        "Ving Rhames"        "Harvey Keitel"
## [11,] "Charles Durning"     "Ray Walston"        "Eileen Brennan"
## [12,] "Stanley Ridges"      "Felix Bressart"     "Lionel Atwill"
## [13,] "Shima Iwashita"      "Tetsurô Tanba"      "Masao Mishima"
## [14,] "Nancy Olson"         "Lloyd Gough"        "Jack Webb"
## [15,] "Marisa Paredes"      "Giustino Durano"    "Horst Buchholz"
## [16,] "Celeste Holm"        "Gary Merrill"       "Hugh Marlowe"
## [17,] "Ralph Meeker"        "Wayne Morris"       "Joe Turkel"
## [18,] "Takashi Shimura"     "Tatsuya Mihashi"    "Tatsuya Nakadai"
## [19,] "Seiji Miyaguchi"     "Minoru Chiaki"      "Daisuke Kato"
## [20,] "Bodil Rosing"        "J. Farrell MacDonald" ""
## [21,] "Tom Powers"          "Porter Hall"        "Jean Heather"
## [22,] "Ray Walston"         "Edie Adams"         "Jack Kruschen"
## [23,] "Douglas Silva"       "Seu Jorge"          "Jonathan Haagensen"
## [24,] "Carl Miller"         "Tom Wilson"         "Henry Bergman"
## [25,] "Jean Kéraudy"        "Raymond Meunier"    "André Bervil"
## [26,] "Vera Miles"          "John McIntire"      "Martin Balsam"
## [27,] "Tom Murray"          "Malcom Waite"       "Henry Bergman"
## [28,] "Lorraine Bracco"     "Paul Sorvino"       "Chuck Low"
## [29,] "Claude Rains"        "Conrad Veidt"       "Sydney Greenstreet"
## [30,] "William Redfield"    "Mews Small"         "Sydney Lassick"
##       [,7]                    [,8]                          [,9]
##  [1,] "Talia Shire"           "Richard S. Castellano ..." ""
##  [2,] "Talia Shire"           "Gastone Moschin ..."       ""
```

```
##  [3,] "John Fiedler"         "Robert Webber ..."         ""
##  [4,] "Norbert Weisser"      "Martin S. Bergmann ..."    ""
##  [5,] "Henry Daniell"        "Norma Varden ..."          ""
##  [6,] "Jack Alexander"       "Tom Dempsey"               "Henry Bergman"
##  [7,] "Mark Rolston"         "Clancy Brown ..."          ""
##  [8,] "Louis Natheaux"       "Allan Garcia"              ""
##  [9,] "Grace Hayle"          "Maurice Moscovitch"        "Billy Gilbert"
## [10,] "Tim Roth"             "Amanda Plummer ..."        ""
## [11,] "Harold Gould"         "Dana Elcar ..."            ""
## [12,] "Sig Ruman"            "Tom Dugan ..."             ""
## [13,] "Ichirô Nakatani"      "Kei Sato ..."              ""
## [14,] "Fred Clark"           "Cecil B. DeMille ..."      ""
## [15,] "Sergio Bini Bustric"  ""                          ""
## [16,] "Gregory Ratoff"       "Barbara Bates ..."         ""
## [17,] "Richard Anderson"     "Timothy Carey ..."         ""
## [18,] "Kenji Kodama"         "Isao Kimura ..."           ""
## [19,] "Isao Kimura"          "Kamatari Fujiwara ..."     ""
## [20,] ""                     ""                          ""
## [21,] "Byron Barr"           "Richard Gaines ..."        ""
## [22,] "Joan Shawlee"         "Hope Holiday ..."          ""
## [23,] "Matheus Nachtergaele" "Jefechander Suplino ..."   ""
## [24,] "Lita Grey"            ""                          ""
## [25,] ""                     ""                          ""
## [26,] "Simon Oakland"        "Patricia Hitchcock"        ""
## [27,] "Betty Morrisey"       ""                          ""
## [28,] "Christopher Serrone"  "Debi Mazar ..."            ""
## [29,] "Peter Lorre"          "S.Z. Sakall ..."           ""
## [30,] "Will Sampson"         "Christopher Lloyd ..."     ""
```

¡Sencillo! Ahora queremos separar los valores numéricos que hemos guardado en el vector POINTS:

```
data_movies[4]
```

```
##                                                                  POINTS
## 1  \n                  9,0\n                  174.532 \n
## 2  \n                  8,9\n                  139.291 \n
## 3   \n                  8,7\n                   70.716 \n
## 4  \n                  8,6\n                  174.827 \n
## 5   \n                  8,6\n                   44.868 \n
## 6   \n                  8,6\n                   32.927 \n
## 7  \n                  8,6\n                  169.030 \n
## 8   \n                  8,6\n                   63.396 \n
## 9   \n                  8,6\n                   85.861 \n
## 10 \n                  8,6\n                  202.371 \n
## 11 \n                  8,5\n                  100.828 \n
## 12  \n                  8,5\n                   35.574 \n
## 13  \n                  8,5\n                   13.401 \n
## 14  \n                  8,5\n                   47.378 \n
## 15 \n                  8,5\n                  191.511 \n
## 16  \n                  8,4\n                   35.671 \n
## 17  \n                  8,4\n                   56.300 \n
## 18   \n                  8,4\n                    9.951 \n
## 19  \n                  8,4\n                   42.917 \n
```

```
## 20  \n                  8,4\n                  14.548 \n
## 21  \n                  8,4\n                  34.102 \n
## 22  \n                  8,4\n                  81.569 \n
## 23 \n                  8,4\n                 128.091 \n
## 24  \n                  8,4\n                  33.218 \n
## 25  \n                  8,4\n                  12.993 \n
## 26 \n                  8,4\n                 109.196 \n
## 27  \n                  8,4\n                  28.141 \n
## 28 \n                  8,4\n                 113.389 \n
## 29  \n                  8,4\n                  98.189 \n
## 30 \n                  8,3\n                 110.966 \n
```

Aquí, lo complicado es que tenemos el puntaje con número adyacentes, pero unidos por comas. Y la cantidad de usuarios son número adyacentes pero divididos por puntos. Juguemos con el código:

```
str_extract_all(data_movies$POINTS,pattern = "\\d+\\.*\\d*")
```

```
## [[1]]
## [1] "9"       "0"       "174.532"
##
## [[2]]
## [1] "8"       "9"       "139.291"
##
## [[3]]
## [1] "8"      "7"      "70.716"
##
## [[4]]
## [1] "8"       "6"       "174.827"
##
## [[5]]
## [1] "8"      "6"      "44.868"
##
## [[6]]
## [1] "8"      "6"      "32.927"
##
## [[7]]
## [1] "8"       "6"       "169.030"
##
## [[8]]
## [1] "8"      "6"      "63.396"
##
## [[9]]
## [1] "8"      "6"      "85.861"
##
## [[10]]
## [1] "8"       "6"       "202.371"
##
## [[11]]
## [1] "8"       "5"       "100.828"
##
## [[12]]
## [1] "8"      "5"      "35.574"
##
```

```
## [[13]]
## [1] "8"      "5"      "13.401"
##
## [[14]]
## [1] "8"      "5"      "47.378"
##
## [[15]]
## [1] "8"       "5"       "191.511"
##
## [[16]]
## [1] "8"      "4"      "35.671"
##
## [[17]]
## [1] "8"      "4"      "56.300"
##
## [[18]]
## [1] "8"     "4"     "9.951"
##
## [[19]]
## [1] "8"      "4"      "42.917"
##
## [[20]]
## [1] "8"      "4"      "14.548"
##
## [[21]]
## [1] "8"      "4"      "34.102"
##
## [[22]]
## [1] "8"      "4"      "81.569"
##
## [[23]]
## [1] "8"       "4"       "128.091"
##
## [[24]]
## [1] "8"      "4"      "33.218"
##
## [[25]]
## [1] "8"      "4"      "12.993"
##
## [[26]]
## [1] "8"       "4"       "109.196"
##
## [[27]]
## [1] "8"      "4"      "28.141"
##
## [[28]]
## [1] "8"       "4"       "113.389"
##
## [[29]]
## [1] "8"      "4"      "98.189"
##
## [[30]]
## [1] "8"       "3"       "110.966"
```

Nos separa el puntaje, pero extrae bien la cantidad de usuarios. Nos sirve para ese caso. Guardamos la información:

```
data_movies$USUARIOS = str_extract_all(data_movies$POINTS,pattern = "\\d+\\.*\\d*", simplify = T) [,3]
```

Póngamonos creativos con el otro valor:

```
data_movies$PUNTAJE1 = str_extract_all(data_movies$POINTS,pattern = "\\d+\\.*\\d*", simplify = T) [,1]
data_movies$PUNTAJE2 = str_extract_all(data_movies$POINTS,pattern = "\\d+\\.*\\d*", simplify = T) [,2]
```

```
data_movies$PUNTAJE = paste(data_movies$PUNTAJE1, data_movies$PUNTAJE2, sep = ",")
```

Limpiamos:

```
data_movies = data_movies[,-c(3,4,16,17)]
```

Limpio! A practicar!

EJERCICIO.

1. Abrir la base de datos de ataques terroristas para la década de 1990, la cual puedes encontrar en el siguiente enlace: https://github.com/Alexanderbenit7/EAP2_2023-2/tree/main/data. Luego, separa la variable "fecha" y crear una columna que contenga solo el año.
2. Abre la base de datos llamada "cia" desde el repositorio de github: https://github.com/WendyAdrianzenRossi/Statistics. Esta es una base de datos sobre los ingresos obtenidos por recursos forestales. La columna "col1" presenta la fecha de recolección de la información y el % del GDP. Intenta limpiar la base de datos, de tal manera que, tengas cada variable en una columna diferente.