TabulaR - Limpieza de datos

00R Team

Marzo 2018

Índice

1. Primer contacto

Las bases de datos en el mundo real están desordenados y sin formatear.

Una buena práctica para formatear los datos es mantener los datos originales, usar un script para organizarlos, arreglar los errores y guardar los datos límpios en un archivo de texto plano (csv) y trabajar con él para el resto del análisis.

Pregunta: considerad los siguientes datos sobre heridos en un accidente. ¿Cuántas variables contiene el conjunto de datos?

hombres	mujeres
4	1
2	5

1.1. ¿Qué es un conjunto de datos *"limpio"*?

Se trata de aquellos conjuntos de datos que cumplen las siguientes características.

- Cada observación es una fila
- Cada variable es un columna
- Contenidos en un único conjunto de datos

Los conjuntos de datos que se han definido de manera limpia, facilitan llevar a cabo un análisis de datos.

2. Exploramos conjuntos de datos *"sucios"*

2.1. Ejemplo 1: distribución de la renta según la religión

Este primer conjunto de datos se basa en una encuesta realizada por *Pew Research Center*, que examina la relación existente la renta y la afiliación religiosa. **Fuente**

```
pew <- read.delim(
  file = "http://stat405.had.co.nz/data/pew.txt",
  header = TRUE,
  stringsAsFactors = FALSE,
  check.names = F
)
```

```
head( pew )
```

```
##              religion <$10k $10-20k $20-30k $30-40k $40-50k $50-75k
##  1           Agnostic    27      34      60      81      76     137
##  2            Atheist    12      27      37      52      35      70
##  3           Buddhist    27      21      30      34      33      58
##  4           Catholic   418     617     732     670     638    1116
##  5 Don't know/refused    15      14      15      11      10      35
##  6   Evangelical Prot   575     869    1064     982     881    1486
##    $75-100k $100-150k >150k Don't know/refused
##  1      122       109    84                 96
##  2       73        59    74                 76
##  3       62        39    53                 54
##  4      949       792   633               1489
##  5       21        17    18                116
##  6      949       723   414               1529
```

Este conjunto de datos tiene tres variables, `religion`, `renta` y `frecuencia`. Para "limpiarlo" lo que hacemos es fundir o juntar los datos. Es decir, tenemos que convertir las columnas en filas.

Usamos la función `melt()` del paquete `reshape2`.

```
# library( reshape2 )
pewl <- melt(
  data = pew,
  id = "religion",
  variable.name = "renta",
  value.name = "frecuencia"
)
```

```
head( pewl )
```

```
##              religion renta frecuencia
##  1           Agnostic <$10k         27
##  2            Atheist <$10k         12
##  3           Buddhist <$10k         27
##  4           Catholic <$10k        418
##  5 Don't know/refused <$10k         15
##  6   Evangelical Prot <$10k        575
```

Lo que hemos hecho es una tabla dinámica, donde hemos mezclado columnas, filas y valores.

2.2. Ejemplo 2: tiempo

Fuente

```
tiempo <- read.delim(
            file = "http://stat405.had.co.nz/data/weather.txt",
            stringsAsFactors = FALSE )
```

```
head( tiempo )
```

```
##             id year month element d1  d2  d3 d4  d5 d6 d7 d8 d9 d10 d11 d12
##  1 MX000017004 2010     1    TMAX NA  NA  NA NA  NA NA NA NA NA  NA  NA  NA
##  2 MX000017004 2010     1    TMIN NA  NA  NA NA  NA NA NA NA NA  NA  NA  NA
##  3 MX000017004 2010     2    TMAX NA 273 241 NA  NA NA NA NA NA  NA 297  NA
##  4 MX000017004 2010     2    TMIN NA 144 144 NA  NA NA NA NA NA  NA 134  NA
##  5 MX000017004 2010     3    TMAX NA  NA  NA NA 321 NA NA NA NA 345  NA  NA
##  6 MX000017004 2010     3    TMIN NA  NA  NA NA 142 NA NA NA NA 168  NA  NA
##    d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30
##  1  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA 278
##  2  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA 145
##  3  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA 299  NA  NA  NA  NA  NA  NA  NA
##  4  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA 107  NA  NA  NA  NA  NA  NA  NA
##  5  NA  NA  NA 311  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
##  6  NA  NA  NA 176  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
##    d31
##  1  NA
##  2  NA
##  3  NA
##  4  NA
##  5  NA
##  6  NA
```

Este conjunto de datos tiene dos problemas. Primero, las variables de las filas de la columna *elements*. Segundo, la variable *d* está está esparcida a través de multiples columnas.

```
# library( dplyr )
tmin <- filter(tiempo, element == "TMIN")
tmin <- tmin [, c(3, 5: ncol( tmin ) ) ]
```

```
head( tmin )
```

```
##    month d1  d2  d3 d4  d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18
##  1     1 NA  NA  NA NA  NA NA NA NA NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
##  2     2 NA 144 144 NA  NA NA NA NA NA  NA 134  NA  NA  NA  NA  NA  NA  NA
##  3     3 NA  NA  NA NA 142 NA NA NA NA 168  NA  NA  NA  NA  NA 176  NA  NA
##  4     4 NA  NA  NA NA  NA NA NA NA NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
##  5     5 NA  NA  NA NA  NA NA NA NA NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
##  6     6 NA  NA  NA NA  NA NA NA NA NA  NA  NA  NA  NA  NA  NA  NA 175  NA
##    d19 d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31
##  1  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA 145  NA
##  2  NA  NA  NA  NA 107  NA  NA  NA  NA  NA  NA  NA  NA
##  3  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
##  4  NA  NA  NA  NA  NA  NA  NA  NA 167  NA  NA  NA  NA
##  5  NA  NA  NA  NA  NA  NA  NA  NA 182  NA  NA  NA  NA
##  6  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA 180  NA  NA
```

```
# library( tidyr )
tminl <- gather( tmin, d1:d31,
            key = "day",
            value = "temparature", na.rm = TRUE )
```

```
head( tminl )
```

```
##     month day temparature
##  11    12  d1         138
##  13     2  d2         144
##  21    11  d2         163
##  24     2  d3         144
##  29     7  d3         175
##  43    11  d4         120
```

Lo hacemos con todo el fichero

Juntamos la variable `day` en una sola columna

```
tiempol <- gather( tiempo, d1:d31,
            key = "day",
            value = "temparature", na.rm = TRUE )
```

```
head( tiempol )
```

```
##              id year month element day temparature
##  21 MX000017004 2010    12    TMAX  d1         299
##  22 MX000017004 2010    12    TMIN  d1         138
##  25 MX000017004 2010     2    TMAX  d2         273
##  26 MX000017004 2010     2    TMIN  d2         144
##  41 MX000017004 2010    11    TMAX  d2         313
##  42 MX000017004 2010    11    TMIN  d2         163
```

Aún así seguimos teniendo la variable `element`

```
tiempol2 <- spread( tiempol, key = element, value = temparature )
```

```
head( tiempol2 )
```

```
##             id year month day TMAX TMIN
##  1 MX000017004 2010     1 d30  278  145
##  2 MX000017004 2010     2 d11  297  134
##  3 MX000017004 2010     2  d2  273  144
##  4 MX000017004 2010     2 d23  299  107
##  5 MX000017004 2010     2  d3  241  144
##  6 MX000017004 2010     3 d10  345  168
```

Un último detalle es que en la variable `day` no siga apareciendo `d1`, `d2`, `d3`, etc, ya sabemos que se refiere a los días del mes.

```
tiempol2 <- mutate( tiempol2,
                day = sub( "^d", "", day )
    )
```

```
head( tiempol2 )
```

```
##             id year month day TMAX TMIN
##  1 MX000017004 2010     1  30  278  145
##  2 MX000017004 2010     2  11  297  134
##  3 MX000017004 2010     2   2  273  144
##  4 MX000017004 2010     2  23  299  107
##  5 MX000017004 2010     2   3  241  144
##  6 MX000017004 2010     3  10  345  168
```