TutorialDplyr

Ricardo Aler

1 Agosto 2019

Introducción a dplyr

dplyr es una librería de manipulado de data.frames. La mayoría de estas operaciones se pueden hacer ya con R-base, pero con *dplyr* son normalmente más rápidas y más claras. *dplyr* manipula data.frames usando "verbos":

- select() selecciona columnas
- mutate() crea nuevas columnas
- filter() selecciona filas
- summarise() resume el data.frame (calcula medias, sumas, ...)
- arrange() ordena filas
- slice() selecciona filas por número de fila
- group_by() agrupa filas que compartan valores. Se pueden calcular después medias de grupos, etc.

Usaremos los datos de supervivencia del Titanic: 891 pasajeros, con sus características: sobrevivió?, clase, nombre, sexo, edad

```
library(titanic)
data("titanic_train")
head(titanic_train)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                  Name    Sex Age SibSp
## 1                             Braund, Mr. Owen Harris   male  22     1
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
## 3                              Heikkinen, Miss. Laina female  26     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
## 5                            Allen, Mr. William Henry   male  35     0
## 6                                    Moran, Mr. James   male  NA     0
##   Parch           Ticket    Fare Cabin Embarked
## 1     0        A/5 21171  7.2500              S
## 2     0         PC 17599 71.2833   C85        C
## 3     0 STON/O2. 3101282  7.9250              S
## 4     0           113803 53.1000  C123        S
## 5     0           373450  8.0500              S
## 6     0           330877  8.4583              Q
```

```
dim(titanic_train)
```

```
## [1] 891  12
```

Los verbos de dplyr

Nos vamos a quedar primero con las columnas que nos interesen con *select*. Podríamos usar también *-Sex* para por ejemplo, quitar la columna *Sex* y dejar las demás.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
t_df <- select(titanic_train, Survived, Pclass, Sex, Age, Fare)

head(t_df)
```

```
##   Survived Pclass    Sex Age    Fare
## 1        0      3   male  22  7.2500
## 2        1      1 female  38 71.2833
## 3        1      3 female  26  7.9250
## 4        1      1 female  35 53.1000
## 5        0      3   male  35  8.0500
## 6        0      3   male  NA  8.4583
```

Vamos a convertir los dolares del billete a euros con *mutate*. *mutate* modifica columnas existentes, o crea columnas nuevas. En este caso, modificamos la columna ya existente *Fare*)

```
t_df <- mutate(t_df, Fare = Fare * 0.86 )
head(t_df)
```

```
##   Survived Pclass    Sex Age      Fare
## 1        0      3   male  22  6.235000
## 2        1      1 female  38 61.303638
## 3        1      3 female  26  6.815500
## 4        1      1 female  35 45.666000
## 5        0      3   male  35  6.923000
## 6        0      3   male  NA  7.274138
```

Podemos quedarnos con una parte del data.frame que cumpla cierta condición. Por ejemplo, vamos a seleccionar los pasajeros mujeres que sobrevivieron con *filter*

```
t_mujeres_supervivientes_df <- filter(t_df, Survived==1 & Sex=="female")
head(t_mujeres_supervivientes_df)
```

```
##   Survived Pclass    Sex Age      Fare
## 1        1      1 female  38 61.303638
## 2        1      3 female  26  6.815500
## 3        1      1 female  35 45.666000
## 4        1      3 female  27  9.574638
## 5        1      2 female  14 25.860888
## 6        1      3 female   4 14.362000
```

Podemos usar *summarise* para calcular por ejemplo, la edad media de los pasajeros (y desviación típica). También creamos una columna. También calculamos lo que se gastaron los pasajeros en total en el pasaje.

```
summarise(t_df, edad_media = mean(Age, na.rm=TRUE), desviacion = sd(Age, na.rm=TRUE), euros = sum(Fare))
```

```
##   edad_media desviacion   euros
## 1   29.69912    14.5265 24676.8
```

Por último *group_by*, permite agrupar filas según algún criterio, y después hacer operaciones a cada grupo. Vamos a calcular la edad media de las personas que viajaban en el Titanic, pero descomponiendo por hombres y mujeres. Parece que los hombres eran algo más mayores que las mujeres y que gastaron algo más en el pasaje.

```
t_agrupado_df <- group_by(t_df, Sex)
summarise(t_agrupado_df, edad_media = mean(Age, na.rm=TRUE), desviacion = sd(Age, na.rm=TRUE), euros = s
```

```
## # A tibble: 2 x 4
##   Sex    edad_media desviacion  euros
##   <chr>       <dbl>      <dbl>  <dbl>
## 1 female       27.9       14.1 12011.
## 2 male         30.7       14.7 12665.
```

Una característica interesante de *dplyr* es el uso de la *pipe* o "entubado", mediante el cual podemos concatenar operaciones. Vamos a repetir la operación anterior con *pipes*, representadas por %>%. Nótese que cuando usamos *pipes*, no hace falta poner el primer argumento (el data.frame que se procesa, *t_df* o *t_agrupado_df*). El data.frame digamos que va circulando por la tubería.

```
resultado <- t_df %>%
  group_by(Sex) %>%
  summarise(edad_media = mean(Age, na.rm=TRUE), desviacion = sd(Age, na.rm=TRUE), euros = sum(Fare))

resultado
```

```
## # A tibble: 2 x 4
##   Sex    edad_media desviacion  euros
##   <chr>       <dbl>      <dbl>  <dbl>
## 1 female       27.9       14.1 12011.
## 2 male         30.7       14.7 12665.
```

Podemos por último ordenar el data.frame con *arrange*, por Fare y por Age. Siempre ordena de menor a mayor. Si queremos que vaya al contrario, ponemos - delante.

```
ordenado <- arrange(t_df, -Fare, -Age)
head(ordenado)
```

```
##   Survived Pclass    Sex Age     Fare
## 1        1      1   male  36 440.6031
## 2        1      1 female  35 440.6031
## 3        1      1   male  35 440.6031
## 4        0      1   male  64 226.1800
## 5        1      1 female  24 226.1800
## 6        1      1 female  23 226.1800
```

Algunos ejercicios

Vamos a responder a algunas preguntas usando *dplyr*. Por ejemplo, ¿cuántos hombres y mujeres sobrevivieron? Vemos que sobrevivieron más mujeres que hombres.

```
resultado <- t_df %>%
  group_by(Sex) %>%
  filter(Survived == 1) %>%
  summarise(total = n())

resultado
```

```
## # A tibble: 2 x 2
##   Sex    total
##   <chr>  <int>
## 1 female   233
## 2 male     109
```

Pero, ¿qué proporción de hombres y mujeres sobrevivió? Como *Survived* vale 0 o 1, según se sobreviviera, basta con calcular la media de dicho atributo. Vemos que proporcionalmente sobrevivieron más las mujeres.

```
resultado <- t_df %>%
  group_by(Sex) %>%
  summarise(media = mean(Survived))

resultado
```

```
## # A tibble: 2 x 2
##   Sex    media
##   <chr>  <dbl>
## 1 female 0.742
## 2 male   0.189
```

Si queremos desglosar por sexo y edad (suponemos que alguien es un niño si tiene menos de 10 años). Desgraciadamente, *Age* tiene muchos NA's, así que los filtramos antes. Vemos que tanto niños como niñas sobrevieron alrededor del 60%.

```
resultado <- t_df %>%
  filter(!is.na(Age)) %>%
  group_by(Sex, Age<=10) %>%
  summarise(media = mean(Survived))

resultado
```

```
## # A tibble: 4 x 3
## # Groups:   Sex [?]
##   Sex    `Age <= 10` media
##   <chr>  <lgl>       <dbl>
## 1 female FALSE       0.774
## 2 female TRUE        0.613
## 3 male   FALSE       0.176
## 4 male   TRUE        0.576
```

Vamos a ver si hay algún patrón de supervivencia por clase. Parece que los de primera clase tienen la tasa de supervivencia más elevada, mientras que de la tercera clase sólo sobrevivió un 24%.

```
resultado <- t_df %>%
  group_by(Pclass) %>%
  summarise(media = mean(Survived))

resultado
```

```
## # A tibble: 3 x 2
```

```
##   Pclass media
##    <int> <dbl>
## 1      1 0.630
## 2      2 0.473
## 3      3 0.242
```

En la misma línea, podemos calcular el pasaje medio de los que sobrevivieron y los que no. Vemos que es bastante más elevado para los primeros.

```
resultado <- t_df %>%
  group_by(Survived) %>%
  summarise(media = mean(Fare))

resultado
```

```
## # A tibble: 2 x 2
##   Survived media
##      <int> <dbl>
## 1        0  19.0
## 2        1  41.6
```