

# Actividad Individual 8

Martín Olivera - CI 4845488-3

17/5/2021

Esta actividad tiene que quedar disponible en su repositorio de GitHub con el resto de las actividades y tareas del curso. Asegurate que tanto Federico como yo seamos colaboradoras de tu proyecto Tareas\_STAT\_NT creado hace dos semanas. Recordar seleccionar en las opciones de proyecto, codificación de código UTF-8. Recordar que para que tengas la última versión de tu repositorio en GitHub debes hacer pull a tu repositorio para tener la copia localmente en tu computadora y no generar inconsistencias y antes de terminar subir tus cambios con `commit` y `push`.

## Ejercicio 1

Cargar la librería `tidyverse` y con los datos `mpg` usar las funciones que vimos de `dplyr` para transformar los datos de la siguiente forma.

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.0 --
## v ggplot2 3.3.3      v purrr 0.3.4
## v tibble 3.0.6       v dplyr 1.0.4
## v tidyr 1.1.2        v stringr 1.4.0
## v readr 1.4.0        v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

library(dplyr)
```

1. Seleccionar el conjunto de autos del 2008 que tienen `hwy` mayor a 31.

¿Cuántos autos son?

```
filtro1 <- filter(mpg, year == 2008 & hwy > 31)
filtro1

## # A tibble: 6 x 11
##   manufacturer model displ year  cyl trans  drv    cty   hwy fl    class
##   <chr>          <chr> <dbl> <int> <int> <chr>  <chr> <int> <int> <chr> <chr>
## 1 honda         civic   1.8  2008    4 manual(~ f      26    34 r      subcom~
## 2 honda         civic   1.8  2008    4 auto(15) f      25    36 r      subcom~
## 3 honda         civic   1.8  2008    4 auto(15) f      24    36 c      subcom~
## 4 nissan         altima  2.5  2008    4 manual(~ f      23    32 r      midsize
## 5 toyota         corol~  1.8  2008    4 manual(~ f      28    37 r      compact
## 6 toyota         corol~  1.8  2008    4 auto(14) f      26    35 r      compact
```

Hay 6 autos luego de filtrada la base.

2. Seleccionar el conjunto de autos que tienen `cty` igual a 9 ó `hwy` es mayor a 90. ¿Cuántos autos son?

```
filtro2 <- filter(mpg, cty == 9 | hwy > 90)
filtro2
```

```
## # A tibble: 5 x 11
##   manufacturer model      displ  year  cyl trans  drv    cty   hwy fl    class
##   <chr>          <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
## 1 dodge        dakota pi~   4.7  2008     8 auto(~ 4     9    12 e    pick~
## 2 dodge        durango 4~   4.7  2008     8 auto(~ 4     9    12 e    suv
## 3 dodge        ram 1500 ~   4.7  2008     8 auto(~ 4     9    12 e    pick~
## 4 dodge        ram 1500 ~   4.7  2008     8 manua~ 4     9    12 e    pick~
## 5 jeep         grand che~   4.7  2008     8 auto(~ 4     9    12 e    suv
```

Hay 5 autos bajo este filtro.

3. Seleccioná la variable `displ` y `hwy`.

```
filtro3 <- select(mpg, c(displ, hwy))
filtro3
```

```
## # A tibble: 234 x 2
##   displ  hwy
##   <dbl> <int>
## 1  1.8    29
## 2  1.8    29
## 3  2      31
## 4  2      30
## 5  2.8    26
## 6  2.8    26
## 7  3.1    27
## 8  1.8    26
## 9  1.8    25
## 10 2      28
## # ... with 224 more rows
```

Comentario: No es necesario concatenar.

4. Seleccioná desde la variable `trans` hasta la variable `hwy`.

```
filtro4 <- select(mpg, trans:hwy)
filtro4
```

```
## # A tibble: 234 x 4
##   trans      drv    cty   hwy
##   <chr>    <chr> <int> <int>
## 1 auto(l5)  f      18    29
## 2 manual(m5) f      21    29
## 3 manual(m6) f      20    31
## 4 auto(av)  f      21    30
## 5 auto(l5)  f      16    26
## 6 manual(m5) f      18    26
## 7 auto(av)  f      18    27
## 8 manual(m5) 4      18    26
## 9 auto(l5)  4      16    25
## 10 manual(m6) 4      20    28
## # ... with 224 more rows
```

5. Produce un dataframe: marca, modelo, año, cantidad de cilindros y rendimiento en ciudad. Únicamente

para los autos, toyota, camry.

```
filtro5 <- filter(mpg, manufacturer == "camry" | manufacturer == "toyota")
filtro5b <- select(filtro5, c(manufacturer, model, year, cyl, cty))
filtro5b <- as.data.frame(filtro5b)
```

Comentario: Correcto pero NO hacerlo de esta forma. No es necesario crear objetos intermedios.

6. Calcula el rendimiento promedio en ciudad para cada marca (manufacturer). ¿Cuál es el mejor rendimiento en ciudad? y el peor ?

```
mpg %>%
  group_by(manufacturer) %>%
  summarise(
    mean_cty = mean(cty, na.rm = TRUE)
  )
```

```
## # A tibble: 15 x 2
##   manufacturer mean_cty
##   * <chr>         <dbl>
## 1 audi           17.6
## 2 chevrolet      15
## 3 dodge          13.1
## 4 ford           14
## 5 honda          24.4
## 6 hyundai        18.6
## 7 jeep           13.5
## 8 land rover     11.5
## 9 lincoln        11.3
## 10 mercury       13.2
## 11 nissan         18.1
## 12 pontiac       17
## 13 subaru        19.3
## 14 toyota        18.5
## 15 volkswagen    20.9
```

El mejor rendimiento en ciudad es el de los autos marca Hyundai, mientras que el de peor rendimiento es el de Lincoln

Comentario: Usar min y max para obtener el mejor y el peor.

7. Para cada marca: calcula el rendimiento promedio en ciudad, el error estandard de la media, y el rango

```
mpg %>%
  group_by(manufacturer) %>%
  summarise(
    mean_cty = mean(cty, na.rm = TRUE),
    sd_cty = sd(cty, na.rm=TRUE),
    rg_cty = max(cty, na.rm=TRUE) - min(cty, na.rm=TRUE)
  )
```

```
## # A tibble: 15 x 4
##   manufacturer mean_cty sd_cty rg_cty
##   * <chr>         <dbl> <dbl> <int>
## 1 audi           17.6  1.97     6
```

```
## 2 chevrolet      15    2.92    11
## 3 dodge          13.1  2.49     9
## 4 ford           14    1.91     7
## 5 honda          24.4  1.94     7
## 6 hyundai        18.6  1.50     5
## 7 jeep           13.5  2.51     8
## 8 land rover     11.5  0.577    1
## 9 lincoln        11.3  0.577    1
## 10 mercury       13.2  0.5     1
## 11 nissan         18.1  3.43    11
## 12 pontiac       17    1       2
## 13 subaru        19.3  0.914   3
## 14 toyota        18.5  4.05    17
## 15 volkswagen    20.9  4.56    19
```

8. Mejora en el rendimiento: calcular el rendimiento promedio para cada marca, distinguiendo antes y después de 2004

```
mpg %>%
  group_by(manufacturer) %>%
  filter(year >= 2004) %>%
  summarise(
    mean_cty = mean(cty, na.rm = TRUE)
  )
```

```
## # A tibble: 15 x 2
##   manufacturer mean_cty
##   <chr>         <dbl>
## 1 audi          18.1
## 2 chevrolet     14.9
## 3 dodge         13.0
## 4 ford          14.1
## 5 honda         24
## 6 hyundai       18.9
## 7 jeep          13.2
## 8 land rover    12
## 9 lincoln       12
## 10 mercury      13
## 11 nissan        18.4
## 12 pontiac      17
## 13 subaru       19.5
## 14 toyota       19.1
## 15 volkswagen   20.5
```

```
mpg %>%
  group_by(manufacturer) %>%
  filter(year < 2004) %>%
  summarise(
    mean_cty = mean(cty, na.rm = TRUE)
  )
```

```
## # A tibble: 15 x 2
##   manufacturer mean_cty
##   <chr>         <dbl>
## 1 audi          17.1
## 2 chevrolet     15.1
```

```
## 3 dodge          13.4
## 4 ford            13.9
## 5 honda           24.8
## 6 hyundai         18.3
## 7 jeep            14.5
## 8 land rover      11
## 9 lincoln         11
## 10 mercury        13.5
## 11 nissan          17.7
## 12 pontiac        17
## 13 subaru         19
## 14 toyota         18.2
## 15 volkswagen     21.2
```

Comentario: Correcto el resultado pero esto debe ser hecho en un solo data.frame no en dos. Ver solución.

9. Calcular el rendimiento promedio en carretera (hwy), para 3 marcas seleccionadas aleatoriamente y ordena el resultado de menor a mayor

```
filter(mpg, manufacturer == sample(manufacturer, 3)) %>%
  group_by(manufacturer) %>%
  summarise(
    mean_hwy = mean(hwy, na.rm = TRUE)
  )
```

```
## # A tibble: 3 x 2
##   manufacturer mean_hwy
## * <chr>         <dbl>
## 1 audi           26.3
## 2 chevrolet      22.1
## 3 nissan          25.5
```

Comentario: Agregar set.seed

10. Crear una nueva variable que transforme hwy (millas por galón) en litros/100 km. Primero tenés que saber la conversión de galones a litros y de millas a kilómetros.

1 gallón = 3.78541 litros

1 milla = 1.60934 km

```
mutate(mpg, hwy = (3.78541*100)/(1.60934*hwy))
```

```
## # A tibble: 234 x 11
##   manufacturer model  displ  year  cyl trans  drv    cty   hwy fl    class
##   <chr>         <chr>    <dbl> <int> <int> <chr>  <chr> <int> <dbl> <chr> <chr>
## 1 audi         a4        1.8  1999    4 auto(l~ f     18  8.11 p    comp~
## 2 audi         a4        1.8  1999    4 manual~ f     21  8.11 p    comp~
## 3 audi         a4        2    2008    4 manual~ f     20  7.59 p    comp~
## 4 audi         a4        2    2008    4 auto(a~ f     21  7.84 p    comp~
## 5 audi         a4        2.8  1999    6 auto(l~ f     16  9.05 p    comp~
## 6 audi         a4        2.8  1999    6 manual~ f     18  9.05 p    comp~
## 7 audi         a4        3.1  2008    6 auto(a~ f     18  8.71 p    comp~
## 8 audi         a4 quat~ 1.8  1999    4 manual~ 4     18  9.05 p    comp~
## 9 audi         a4 quat~ 1.8  1999    4 auto(l~ 4     16  9.41 p    comp~
## 10 audi        a4 quat~ 2    2008    4 manual~ 4     20  8.40 p    comp~
```

```
## # ... with 224 more rows
```

Comentario: Muy buen trabajo Martín. A mejorar: El código debe ser más ordenado en especial estar indentado para ser más legible.