

# Tarea8

Yimmy Eman

2022-07-07

```
rm(list = ls())
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.6    v purrr  0.3.4
## v tibble  3.1.7    v dplyr  1.0.9
## v tidyr   1.2.0    v stringr 1.4.0
## v readr   2.1.2    v forcats 0.5.1
```

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

```
library(nycflights13)
```

```
df <- flights
```

## Pregunta 1

Encuentra todos los vuelos que llegaron más de una hora tarde de lo previsto.

```
#dep_delay, arr_delay
#Retrasos de salida y llegada, en minutos. Los tiempos negativos representan #salidas/llegadas anticipa
```

```
filter(df, arr_delay > 60)
```

```
## # A tibble: 27,789 x 19
```

```
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>    <int>         <int>
## 1  2013     1     1     811             630          101    1047             830
## 2  2013     1     1     848             1835         853    1001             1950
## 3  2013     1     1     957             733          144    1056             853
## 4  2013     1     1    1114             900          134    1447             1222
## 5  2013     1     1    1120             944           96    1331             1213
## 6  2013     1     1    1255             1200           55    1451             1330
## 7  2013     1     1    1301             1150           71    1518             1345
## 8  2013     1     1    1337             1220           77    1649             1531
```

```
## 9 2013 1 1 1342 1320 22 1617 1504
## 10 2013 1 1 1400 1250 70 1645 1502
## # ... with 27,779 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 2

Encuentra todos los vuelos que volaron hacia San Francisco (aeropuertos SFO y OAK)

```
filter(df, dest == "SFO" | dest == "OAK")
```

```
## # A tibble: 13,643 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>         <int>
## 1 2013     1     1     558           600          -2     923           937
## 2 2013     1     1     611           600          11     945           931
## 3 2013     1     1     655           700          -5    1037          1045
## 4 2013     1     1     729           730          -1    1049          1115
## 5 2013     1     1     734           737          -3    1047          1113
## 6 2013     1     1     745           745           0    1135          1125
## 7 2013     1     1     746           746           0    1119          1129
## 8 2013     1     1     803           800           3    1132          1144
## 9 2013     1     1     826           817           9    1145          1158
## 10 2013     1     1    1029          1030          -1    1427          1355
## # ... with 13,633 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 3

Encuentra todos los vuelos operados por United American ('UA') o por American Airlines (AA)

```
filter(df, carrier == "UA" | carrier == "AA")
```

```
## # A tibble: 91,394 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>         <int>
## 1 2013     1     1     517           515           2     830           819
## 2 2013     1     1     533           529           4     850           830
## 3 2013     1     1     542           540           2     923           850
## 4 2013     1     1     554           558          -4     740           728
## 5 2013     1     1     558           600          -2     753           745
## 6 2013     1     1     558           600          -2     924           917
## 7 2013     1     1     558           600          -2     923           937
## 8 2013     1     1     559           600          -1     941           910
## 9 2013     1     1     559           600          -1     854           902
## 10 2013     1     1     606           610          -4     858           910
## # ... with 91,384 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 4

Encuentra todos los vuelos que salieron los meses de primavera (Abril, Mayo y Junio)

```
filter(df, month == 4 | month == 5 | month == 6)
```

```
## # A tibble: 85,369 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>         <int>
## 1  2013     4     1     454             500          -6     636             640
## 2  2013     4     1     509             515          -6     743             814
## 3  2013     4     1     526             530          -4     812             827
## 4  2013     4     1     534             540          -6     833             850
## 5  2013     4     1     542             545          -3     914             920
## 6  2013     4     1     543             545          -2     921             927
## 7  2013     4     1     551             600          -9     748             659
## 8  2013     4     1     552             600          -8     641             701
## 9  2013     4     1     553             600          -7     725             735
##10  2013     4     1     554             600          -6     752             805
## # ... with 85,359 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 5

Encuentra todos los vuelos que llegaron más de una hora tarde, pero salieron con menos de una hora de retraso.

```
filter(df, arr_delay > 60 & dep_delay < 60)
```

```
## # A tibble: 4,956 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>         <int>
## 1  2013     1     1    1255             1200         55    1451             1330
## 2  2013     1     1    1342             1320         22    1617             1504
## 3  2013     1     1    1402             1323         39    1650             1526
## 4  2013     1     1    1411             1315         56    1717             1611
## 5  2013     1     1    1424             1349         35    1701             1556
## 6  2013     1     1    1428             1329         59    1803             1640
## 7  2013     1     1    1558             1534         24    1808             1703
## 8  2013     1     1    1604             1510         54    1817             1710
## 9  2013     1     1    1608             1535         33    2002             1850
##10  2013     1     1    1630             1548         42    1902             1755
## # ... with 4,946 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 6

Encuentra todos los vuelos que salieron con más de una hora de retraso, pero consiguieron llegar con menos de 30 minutos de retraso (el avión aceleró en el aire)

```
filter(df, dep_delay > 60, arr_delay < 30)
```

```
## # A tibble: 181 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>       <dbl>   <int>         <int>
## 1  2013     1     3    1850           1745         65    2148           2120
## 2  2013     1     3    1950           1845         65    2228           2227
## 3  2013     1     6    1019           900          79    1558           1530
## 4  2013     1     7    1543           1430         73    1758           1735
## 5  2013     1    12    1706           1600         66    1949           1927
## 6  2013     1    12    1953           1845         68    2154           2137
## 7  2013     1    19    1456           1355         61    1636           1615
## 8  2013     1    21    1531           1430         61    1843           1815
## 9  2013     1    21    1648           1545         63    1939           1910
## 10 2013    10    10    1938           1835         63    2158           2148
## # ... with 171 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 7

Encuentra todos los vuelos que salen entre medianoche y las 7 de la mañana (vuelos nocturnos).

```
filter(df, hour >= 0, hour < 7)
```

```
## # A tibble: 27,905 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>       <dbl>   <int>         <int>
## 1  2013     1     1     517           515          2     830           819
## 2  2013     1     1     533           529          4     850           830
## 3  2013     1     1     542           540          2     923           850
## 4  2013     1     1     544           545         -1    1004          1022
## 5  2013     1     1     554           600         -6     812           837
## 6  2013     1     1     554           558         -4     740           728
## 7  2013     1     1     555           600         -5     913           854
## 8  2013     1     1     557           600         -3     709           723
## 9  2013     1     1     557           600         -3     838           846
## 10 2013     1     1     558           600         -2     753           745
## # ... with 27,895 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 8

Investiga el uso de la función `between()` de `dplyr`. ¿Qué hace? ¿Puedes usarlo para resolver la sintaxis necesaria para responder alguna de las preguntas anteriores?

Este es un atajo para `x >= izquierda` y `x <= derecha`, implementado eficientemente en C++ para valores locales y traducido al SQL apropiado para tablas remotas.

## Pregunta 9

¿Cuántos vuelos tienen un valor desconocido de dep\_time?

```
filter(df, is.na(dep_time))
```

```
## # A tibble: 8,255 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>         <int>
## 1  2013     1     1     NA             1630           NA       NA             1815
## 2  2013     1     1     NA             1935           NA       NA             2240
## 3  2013     1     1     NA             1500           NA       NA             1825
## 4  2013     1     1     NA              600           NA       NA              901
## 5  2013     1     2     NA             1540           NA       NA             1747
## 6  2013     1     2     NA             1620           NA       NA             1746
## 7  2013     1     2     NA             1355           NA       NA             1459
## 8  2013     1     2     NA             1420           NA       NA             1644
## 9  2013     1     2     NA             1321           NA       NA             1536
##10  2013     1     2     NA             1545           NA       NA             1910
## # ... with 8,245 more rows, and 11 more variables: arr_delay <dbl>,
## #   carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

## Pregunta 10

¿Qué variables del dataset contienen valores desconocidos? ¿Qué representan esas filas donde faltan los datos?

Los campos desconocidos son fechas de salida/llegada así como retraso de salida/llegada. Puede que se trate de vuelos cancelados, sobre todo por las cifras que manejamos (unos 8000 vuelos anuales)<sup>2</sup>

## Pregunta 11

Ahora vas a sorprenderte con la magia oscura. . . Contesta qué dan las siguientes condiciones booleanas  
-  $NA^0$  -  $NA|TRUE$  -  $FALSE \& NA$  Intenta establecer la regla general para saber cuando es o no es NA (cuidado con  $NA^0$ )

```
NA ^ 0
```

```
## [1] 1
```

```
NA | TRUE
```

```
## [1] TRUE
```

```
FALSE & NA
```

```
## [1] FALSE
```

TRUE es un absorbente de la DISYUNCION y FALSE de la CONJUNCION.