

Tarea 2. Transformación de datos

Dalia Camacho

August 28, 2018

Transformación de datos Entrega: Lunes 27 de agosto.

Utiliza los datos de vuelos (flights) para responder la siguientes preguntas.

```
##  
## 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  
## Parsed with column specification:  
## cols(  
##   date = col_datetime(format = ""),  
##   hour = col_integer(),  
##   minute = col_integer(),  
##   dep = col_integer(),  
##   arr = col_integer(),  
##   dep_delay = col_integer(),  
##   arr_delay = col_integer(),  
##   carrier = col_character(),  
##   flight = col_integer(),  
##   dest = col_character(),  
##   plane = col_character(),  
##   cancelled = col_integer(),  
##   time = col_integer(),  
##   dist = col_integer()  
## )
```

¿A qué hora del día debo volar para evitar, lo más posible, retrasos de salida?

```
flights %>%  
  group_by(hour)%>%  
  summarise(mean=mean(dep_delay)) %>%  
  arrange(mean)
```

```
## # A tibble: 25 x 2  
##       hour     mean  
##     <int>    <dbl>  
## 1      4    -10  
## 2      5   -3.28  
## 3      6   -0.0143  
## 4      7    0.0467  
## 5      8    2.62
```

```

## 6   10  3.62
## 7    9  3.73
## 8   11  7.16
## 9   13  7.73
## 10  12  8.03
## # ... with 15 more rows

```

Para evitar retrasos de salida, la mejor hora del día son las 4 de la mañana.

Para cada destino calcula el total de minutos de retraso de salida.

```

dest_delay <- flights %>%
  group_by(dest) %>%
  summarise(tot_delay = sum(dep_delay, na.rm = T))

glimpse(dest_delay)

## # Observations: 116
## # Variables: 2
## # $ dest      <chr> "ABQ", "AEX", "AGS", "AMA", "ANC", "ASE", "ATL", "AU...
## # $ tot_delay <int> 23630, 4542, 10, 8480, 3119, 1918, 79750, 42063, 308...

```

Para cada vuelo calcula su proporción del total de retrasos de su destino.

```

flt_delay <- flights %>%
  filter(dep_delay > 0) %>%
  group_by(dest) %>%
  mutate(tot_dest_delay = sum(dep_delay, na.rm = TRUE)) %>%
  group_by(dest, hour, carrier) %>%
  summarise(prop_delay = (mean(dep_delay / tot_dest_delay)))

glimpse(flt_delay %>%
  arrange(desc(prop_delay)))

## # Observations: 2,942
## # Variables: 4
## # $ dest      <chr> "AGS", "GRK", "BPT", "BKG", "BKG", "EGE", "MTJ", "P...
## # $ hour      <int> 20, 20, 9, 14, 16, 16, 13, 12, 1, 13, 13, 20, 11, 1...
## # $ carrier   <chr> "CO", "XE", "XE", "FL", "FL", "CO", "CO", "OO", "CO...
## # $ prop_delay <dbl> 1.00000000, 0.57446809, 0.50000000, 0.28461538, 0.1...

```

Los retrasos suelen estar correlacionados temporalmente, incluso cuando se ha resuelto el problema que ocasionó los retrasos iniciales, vuelos posteriores suelen mantener algo de retraso. Usando la función lag() explora como el retraso de salida de un vuelo se relaciona con el retraso de salida del vuelo anterior. Realiza una gráfica para visualizar tus hallazgos.

```

# ?lag
flights <- flights %>%
  arrange(date, hour) %>%

```

```

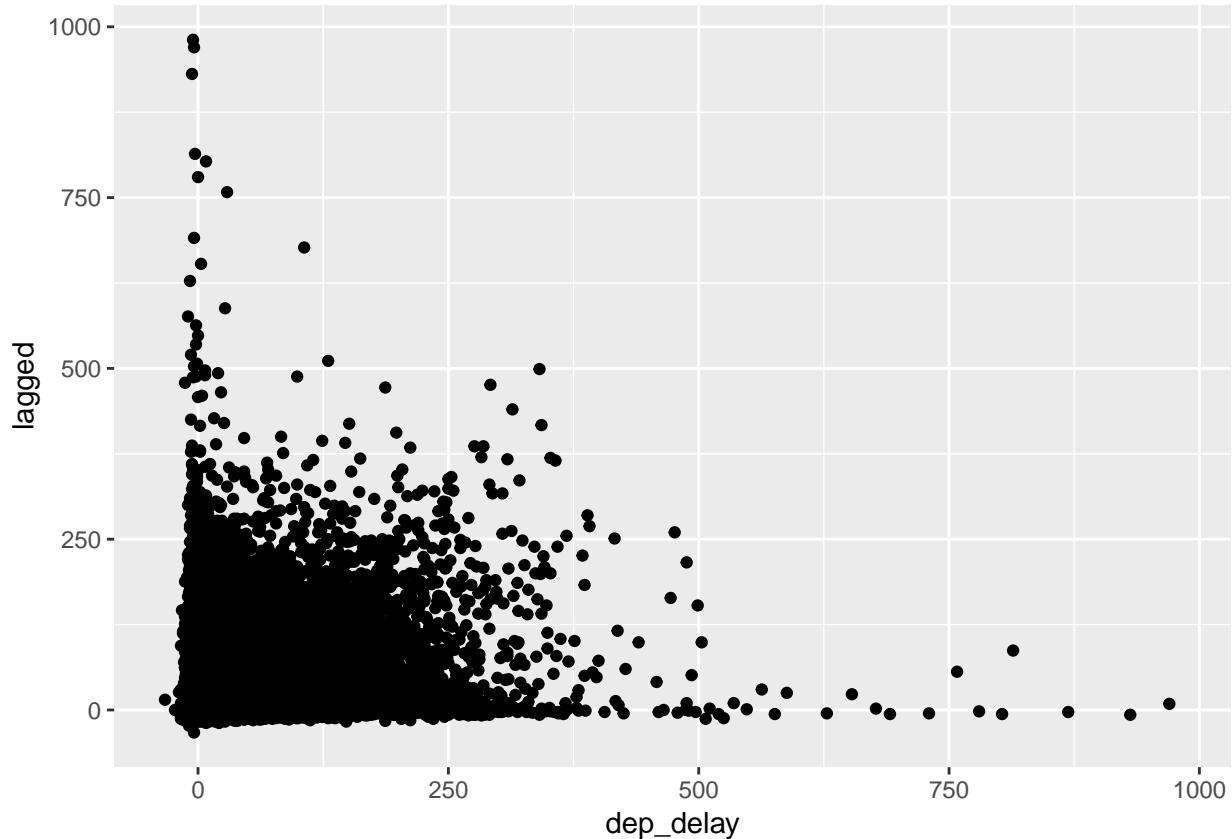
  mutate(lagged=lag(dep_delay))

ggplot(flights)+  

  geom_point(aes(dep_delay,lagged))

## Warning: Removed 3220 rows containing missing values (geom_point).

```



Los vuelos con mayor retraso de salida tienen un lag cercano a cero, por lo que vuelos anteriores influyen en el retraso de vuelos posteriores

Para cada destino, puedes encontrar vuelos sospechosamente rápidos o lentos? (quizá debido a problemas en la captura de los datos).

```

# Ordenar vuelos por destino y tiempo de retraso en el aire del más letno al más rápido
glimpse(flights %>%
  mutate(flt_delay=arr_delay-dep_delay) %>%
  arrange(desc(dest,flt_delay)))

```

```

## Observations: 227,496
## Variables: 16
## $ date      <dttm> 2011-01-01 12:00:00, 2011-01-01 12:00:00, 2011-01-0...
## $ hour       <int> 13, 16, 13, 16, 20, 13, 17, 20, 13, 16, 19, 13, 15, ...
## $ minute     <int> 17, 24, 15, 24, 16, 14, 35, 57, 17, 4, 57, 17, 58, 2...
## $ dep        <int> 1317, 1624, 1315, 1624, 2016, 1314, 1735, 2057, 1317...
## $ arr        <int> 1444, 1748, 1437, 1745, 2139, 1433, 1859, 2213, 1443...
## $ dep_delay  <int> -3, -1, -5, -1, 1, -6, 70, 52, -3, -1, -8, -3, -7, -...

```

```

## $ arr_delay <int> 1, 0, -11, -8, -7, -15, 66, 37, -5, -9, -5, 2, -12, ...
## $ carrier   <chr> "XE", "XE", "XE", "XE", "XE", "XE", "XE", "XE"...
## $ flight    <int> 2213, 2920, 2213, 2920, 2800, 2213, 2920, 2800, 2213...
## $ dest      <chr> "XNA", "XNA", "XNA", "XNA", "XNA", "XNA", "XNA", "XN...
## $ plane     <chr> "N14925", "N11192", "N14940", "N16963", "N13994", "N...
## $ cancelled <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ time      <int> 60, 58, 67, 65, 65, 61, 60, 59, 63, 60, 59, 68, 67, ...
## $ dist      <int> 438, 438, 438, 438, 438, 438, 438, 438, 438, 438, 438, 43...
## $ lagged    <int> 31, 4, 0, 36, 0, 25, 1, 37, 8, 4, 12, 26, 2, 7, -3, ...
## $ flt_delay <int> 4, 1, -6, -7, -8, -9, -4, -15, -2, -8, 3, 5, -5, -11...

```

Calcula el tiempo de vuelo relativo a la mediana de tiempo de vuelo a su destino. Qué vuelos se retrasaron más en el aire?

```

glimpse(flights %>%
  group_by(dest) %>%
  mutate(dif=time-median(time, na.rm = TRUE)) %>%
  arrange(desc(dif)))

## # Observations: 227,496
## # Variables: 16
## $ date      <dttm> 2011-08-18 12:00:00, 2011-08-20 12:00:00, 2011-04-0...
## $ hour       <int> 7, 8, 20, 10, 18, 10, 12, 13, 15, 13, 9, 20, 18, 20, ...
## $ minute     <int> 10, 33, 17, 58, 25, 10, 54, 21, 29, 20, 42, 5, 39, 5...
## $ dep        <int> 710, 833, 2017, 1058, 1825, 1010, 1254, 1321, 1529, ...
## $ arr        <int> 1028, 1225, 43, 1441, 2337, 1316, 1818, 1906, 2124, ...
## $ dep_delay  <int> -10, 98, 152, 258, 30, 60, -1, 11, -2, 10, 17, 30, 1...
## $ arr_delay  <int> 94, 175, 243, 331, 93, 149, 79, 76, 83, 71, 87, 78, ...
## $ carrier    <chr> "XE", "MQ", "FL", "MQ", "CO", "XE", "CO", "CO", "B6"...
## $ flight     <int> 2454, 3859, 292, 3859, 1625, 2799, 1561, 810, 624, 8...
## $ dest       <chr> "MEM", "ORD", "ATL", "ORD", "DCA", "ICT", "DCA", "EW...
## $ plane      <chr> "N27962", "N512MQ", "N176AT", "N526MQ", "N33294", "N...
## $ cancelled  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ time       <int> 182, 215, 186, 205, 233, 159, 232, 257, 258, 252, 54...
## $ dist       <int> 468, 925, 696, 925, 1208, 542, 1208, 1400, 1428, 140...
## $ lagged    <int> 0, 0, 34, -1, 6, 1, 0, -5, 0, 2, -2, 9, 15, 2, 18, 4...
## $ dif        <dbl> 113, 93, 92, 83, 82, 81, 81, 80, 76, 75, 73, 73, 72, ...

```

Encuentra los destinos que se vuelan por al menos dos compañías (carriers).

```

flights %>%
  group_by(dest) %>%
  summarise(N=length(unique(carrier))) %>%
  filter(N>=2)

## # A tibble: 60 x 2
##   dest      N
##   <chr> <int>
## 1 ABQ      4
## 2 ATL      6

```

```
## 3 AUS      4
## 4 BHM      3
## 5 BNA      2
## 6 BWI      2
## 7 CHS      2
## 8 CLT      5
## 9 CMH      3
## 10 COS     2
## # ... with 50 more rows
```