

```
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)
library(Lahman)
library(babynames)
library(fueleconomy)
library(nasaweather)

##
## Attaching package: 'nasaweather'

## The following object is masked from 'package:dplyr':
##
##     storms
```

## Pregunta 1

Imagina que quieres dibujar la ruta de cada avión desde su origen a su destino. ¿Qué variables necesitarías y qué tablas deberías combinar?

```
flights %>%
  select(tailnum, origin, dest)

## # A tibble: 336,776 x 3
##   tailnum origin dest
##   <chr>    <chr> <chr>
## 1 N14228  EWR    IAH
## 2 N24211  LGA    IAH
## 3 N619AA  JFK    MIA
## 4 N804JB  JFK    BQN
## 5 N668DN  LGA    ATL
## 6 N39463  EWR    ORD
## 7 N516JB  EWR    FLL
## 8 N829AS  LGA    IAD
## 9 N593JB  JFK    MCO
## 10 N3ALAA LGA    ORD
## # ... with 336,766 more rows
```

## Pregunta 2

- ¿Qué relaciones existen entre las tablas weather y airports? ¿Qué claves son las que se relacionan entre ambas tablas?

```
# Se realacionan "faa" y "origin"
airports %>%
  left_join(weather, by = c("faa" = "origin"))
```

```
## # A tibble: 27,570 x 22
##   faa   name      lat   lon   alt   tz dst  tzone  year month   day  hour
##   <chr> <chr>    <dbl> <dbl> <dbl> <dbl> <chr> <chr> <int> <int> <int> <int>
## 1 04G   Lansdowne~ 41.1  -80.6 1044   -5 A    Amer~   NA    NA    NA    NA
## 2 06A   Moton Fie~ 32.5  -85.7 264    -6 A    Amer~   NA    NA    NA    NA
## 3 06C   Schaumbur~ 42.0  -88.1 801    -6 A    Amer~   NA    NA    NA    NA
## 4 06N   Randall A~ 41.4  -74.4 523    -5 A    Amer~   NA    NA    NA    NA
## 5 09J   Jekyll Is~ 31.1  -81.4 11     -5 A    Amer~   NA    NA    NA    NA
## 6 0A9   Elizabeth~ 36.4  -82.2 1593   -5 A    Amer~   NA    NA    NA    NA
## 7 0G6   Williams ~ 41.5  -84.5 730    -5 A    Amer~   NA    NA    NA    NA
## 8 0G7   Finger La~ 42.9  -76.8 492    -5 A    Amer~   NA    NA    NA    NA
## 9 0P2   Shoestrin~ 39.8  -76.6 1000   -5 U    Amer~   NA    NA    NA    NA
## 10 OS9  Jefferson~ 48.1 -123. 108    -8 A    Amer~   NA    NA    NA    NA
## # ... with 27,560 more rows, and 10 more variables: temp <dbl>, dewp <dbl>,
## #   humid <dbl>, wind_dir <dbl>, wind_speed <dbl>, wind_gust <dbl>,
## #   precip <dbl>, pressure <dbl>, visib <dbl>, time_hour <dtm>
```

- weather solo contiene información de los aeropuertos de origen de NYC. Si contuviera información meteorológica de todos los aeropuertos de Estados Unidos, qué relación adicional tendríamos que definir entre esta tabla y flights?

Resp: Tendríamos que realacionar la variable destino de la tabla flights.

- Algunos días del año son especiales (festivos como el 4 de Julio en América) y menos personal suele volar. ¿Cómo indicaríamos esta información como tabla en forma de data frame? ¿Cuáles serían sus claves primarias y cómo se relacionaría la nueva tabla con las ya existentes del dataset de vuelos?

```
flights %>%
  filter(month == 7, day == 4) -> july4

flights %>%
  semi_join(july4)
```

```
## Joining, by = c("year", "month", "day", "dep_time", "sched_dep_time",
## "dep_delay", "arr_time", "sched_arr_time", "arr_delay", "carrier", "flight",
## "tailnum", "origin", "dest", "air_time", "distance", "hour", "minute",
## "time_hour")
```

```
## # A tibble: 737 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     7     4     11           2359           12     400           340
```

```
## 2 2013 7 4 59 2359 60 501 350
## 3 2013 7 4 454 500 -6 635 640
## 4 2013 7 4 535 536 -1 802 806
## 5 2013 7 4 538 540 -2 835 840
## 6 2013 7 4 539 545 -6 918 921
## 7 2013 7 4 542 545 -3 814 813
## 8 2013 7 4 553 600 -7 659 712
## 9 2013 7 4 554 600 -6 822 815
## 10 2013 7 4 556 600 -4 705 711
## # ... with 727 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>
```

- Inventa una clave sustituta para el dataset de vuelos.

## Pregunta 3

Identifica las claves de los siguientes datasets de R (puede que te falten algunos paquetes así que instálalos antes):

- Lahman::Batting

*#Las claves primarias son:*

```
Batting %>%
  count(playerID, teamID, yearID, stint) %>%
  filter(n>1)
```

```
## [1] playerID teamID yearID stint n
## <0 rows> (or 0-length row.names)
```

- babynames::babynames

```
babynames %>%
  count(name, sex, year) %>%
  filter(n>1)
```

```
## # A tibble: 0 x 4
## # ... with 4 variables: name <chr>, sex <chr>, year <dbl>, n <int>
```

- fueleconomy::vehicles

*# La columna id es el identificador único de cada vehículo*

```
vehicles %>%
  count(id) %>%
  filter(n>1)
```

```
## # A tibble: 0 x 2
## # ... with 2 variables: id <dbl>, n <int>
```

- ggplot2::diamonds

```
diamonds %>%
  count(carat, cut, table, price, clarity, color, depth, x,y,z) %>%
  filter(n>2)
```

```
## # A tibble: 1 x 11
##   carat cut    table price clarity color depth      x      y      z      n
##   <dbl> <ord> <dbl> <int> <ord>   <ord> <dbl> <dbl> <dbl> <dbl> <int>
## 1  0.79 Ideal     57  2898 SI1     G     62.3  5.9  5.85  3.66     5
```

- nasaweather::atmos

```
atmos %>%
  count(lat, long, year, month) %>%
  filter(n>1)
```

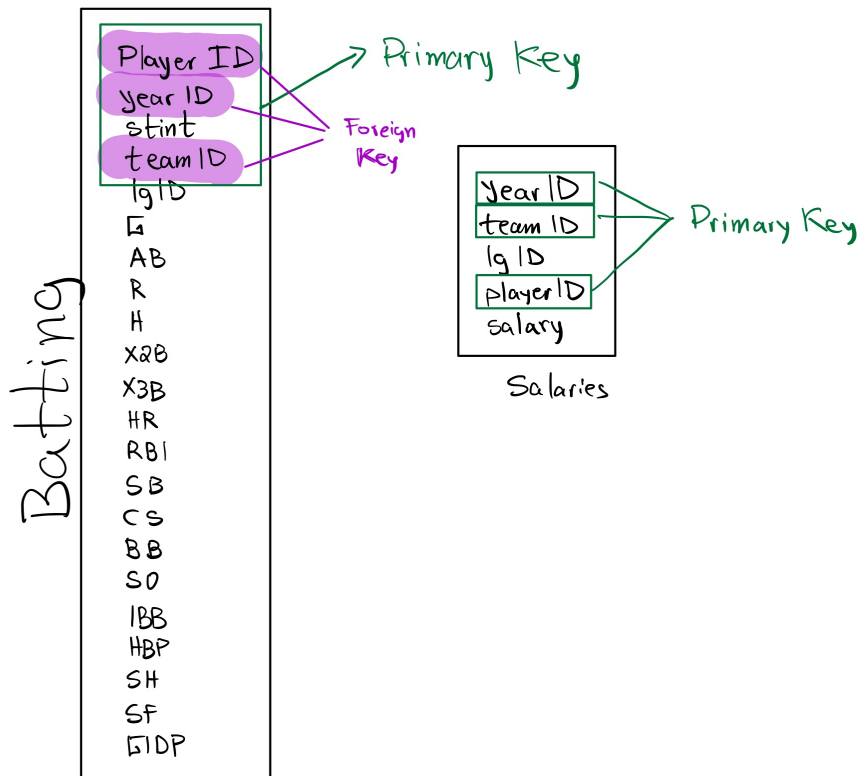
```
## # A tibble: 0 x 5
## # ... with 5 variables: lat <dbl>, long <dbl>, year <int>, month <int>, n <int>
```

## Pregunta 4

- Dibuja un diagrama que muestre las interrelaciones entre las tablas Batting, Master y Salaries del paquete Lahman.

```
Lahman::Salaries %>%
  count(yearID, teamID, playerID) %>%
  filter(n>1)
```

```
## [1] yearID teamID playerID n
## <0 rows> (or 0-length row.names)
```



- Dibuja otro diagrama que muestre las interrelaciones entre las tablas Master, Managers y AwardsManagers del mismo.

La tabla Master no se encuentra.

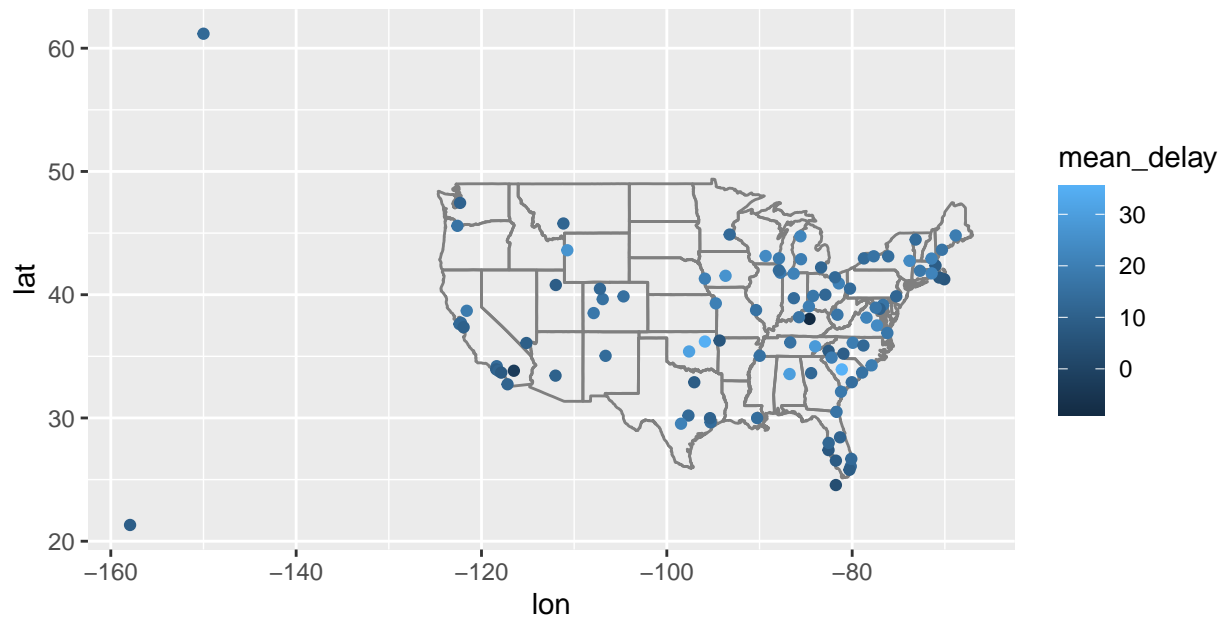
¿Cómo caracterizarías la relación existente entre las tablas Batting, Pitching y Fielding?

## Pregunta 5

Calcula el retraso promedio por destino. Luego haz un join con el dataset de airports para mostrar información espacial de los retrasos. Pinta un mapa con puntos proporcionales al retraso por cada destino (recuerda usar los parámetros size o colour para mostrar el retraso promedio de cada aeropuerto).

```
flights %>%
  group_by(dest) %>%
  summarise(mean_delay = mean(dep_delay, na.rm = TRUE)) %>%
  inner_join(airports, by = c(dest = "faa")) -> mean_dest_delay
```

```
mean_dest_delay %>%
  ggplot(aes(lon, lat))+
  borders("state")+
  geom_point(aes(col = mean_delay))+
  coord_quickmap()
```



## Pregunta 6

- Añade la localización del origen y destino (latitud y longitud) al dataset de vuelos.

```
locations <- airports %>%
  select(faa, lat, lon)

flights %>%
  select(year, month, day, hour, origin, dest) %>%
  left_join(locations, by = c("origin" = "faa")) %>%
  left_join(locations, by = c("dest" = "faa"), suffix = c(".origin", ".dest"))
```

```
## # A tibble: 336,776 x 10
##   year month   day hour origin dest lat.origin lon.origin lat.dest lon.dest
##   <int> <int> <int> <dbl> <chr> <chr>   <dbl>   <dbl>   <dbl>   <dbl>
## 1  2013     1     1     5 EWR   IAH     40.7    -74.2    30.0    -95.3
## 2  2013     1     1     5 LGA   IAH     40.8    -73.9    30.0    -95.3
## 3  2013     1     1     5 JFK   MIA     40.6    -73.8    25.8    -80.3
## 4  2013     1     1     5 JFK   BQN     40.6    -73.8    NA      NA
## 5  2013     1     1     6 LGA   ATL     40.8    -73.9    33.6    -84.4
## 6  2013     1     1     5 EWR   ORD     40.7    -74.2    42.0    -87.9
## 7  2013     1     1     6 EWR   FLL     40.7    -74.2    26.1    -80.2
## 8  2013     1     1     6 LGA   IAD     40.8    -73.9    38.9    -77.5
## 9  2013     1     1     6 JFK   MCO     40.6    -73.8    28.4    -81.3
## 10 2013     1     1     6 LGA   ORD     40.8    -73.9    42.0    -87.9
## # ... with 336,766 more rows
```

- Investiga si existe alguna relación entre la edad del avión y sus retrasos (utiliza algún gráfico de ggplot).

```
# Primero filtramos por edad y cola del avión.

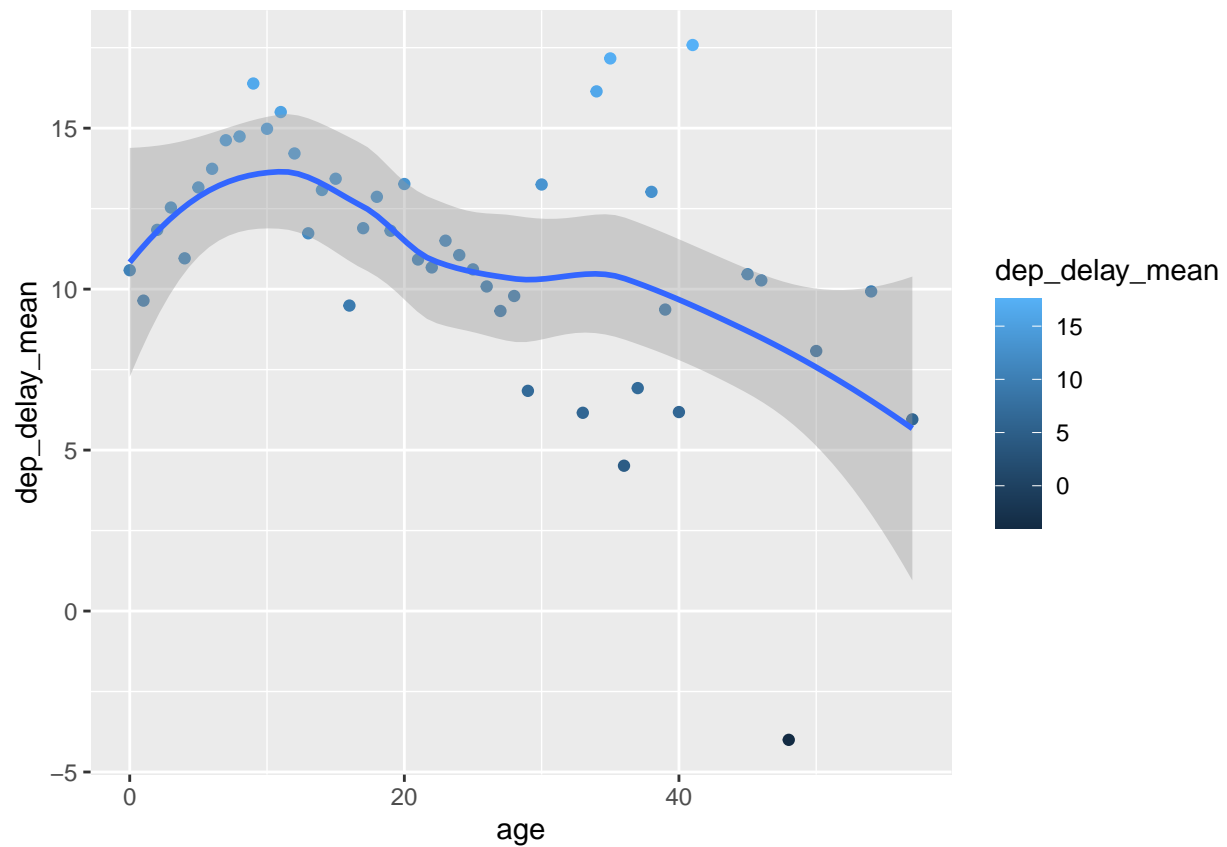
plane_age <- planes %>%
  select(tailnum, plane_year = year)

# Hacemos un inner_join de vuelos con plane_age filtrando por tailnum

plane_data <- flights %>%
  inner_join(plane_age, by = "tailnum") %>%
  # Hacemos un mutate para calcular la edad
  mutate(age = year - plane_year) %>%
  # Filtramos por los que no son NA
  filter(!is.na(age)) %>% # Nos cargamos 5306 Observaciones de age con NA
  group_by(age) %>%
  #Calculamos algunos estadísticos
  summarise(
    dep_delay_mean = mean(dep_delay, na.rm = TRUE),
    arr_delay_mean = mean(arr_delay, na.rm = TRUE)
  ) %>%
  arrange(desc(age))

plane_data %>%
  ggplot(aes(age, dep_delay_mean,
             col = dep_delay_mean)) +
  geom_point() +
  geom_smooth()

## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

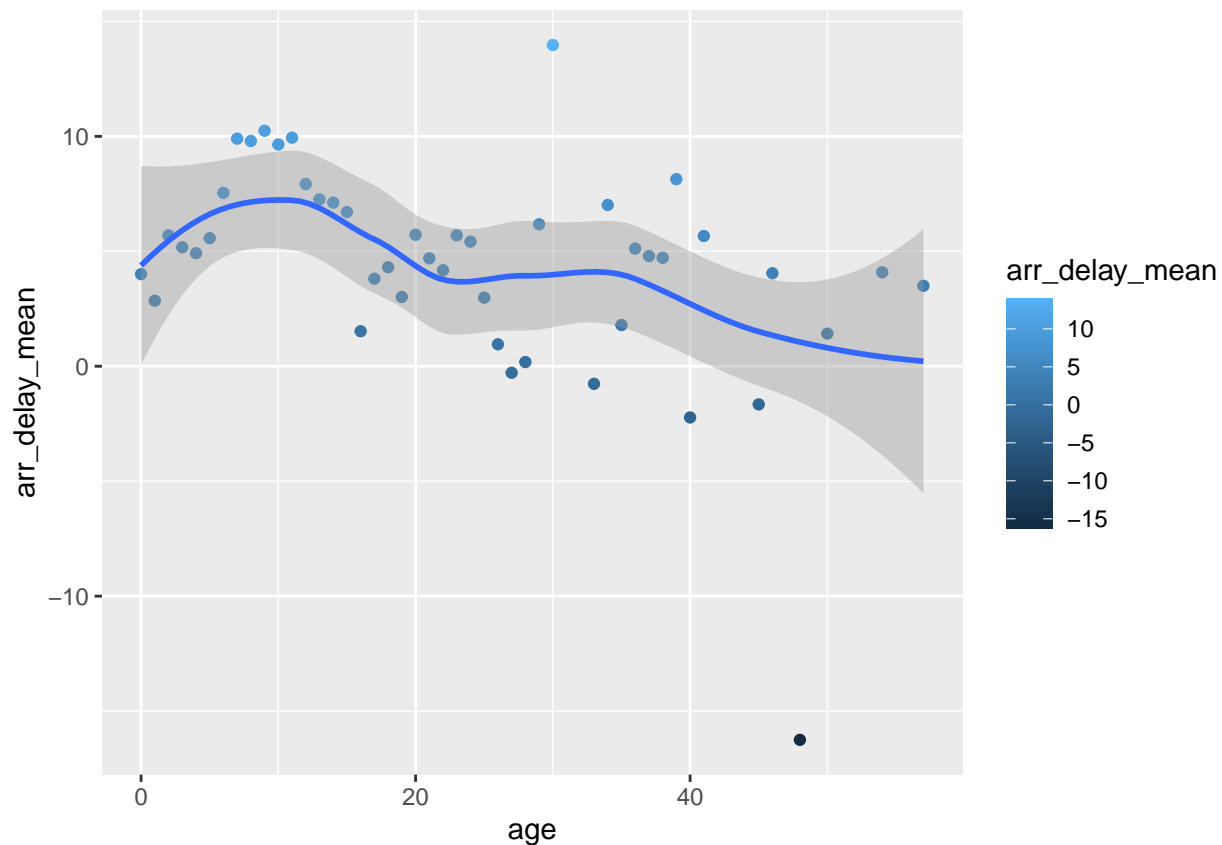


*# Según este gráfico la mayor cantidad de retrasos se concentran  
# cuando el avión llega a 10 años.*

```
plane_data %>%
  ggplot(aes(age, arr_delay_mean, col = arr_delay_mean)) +
  geom_point() +
  geom_smooth()
```

```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```





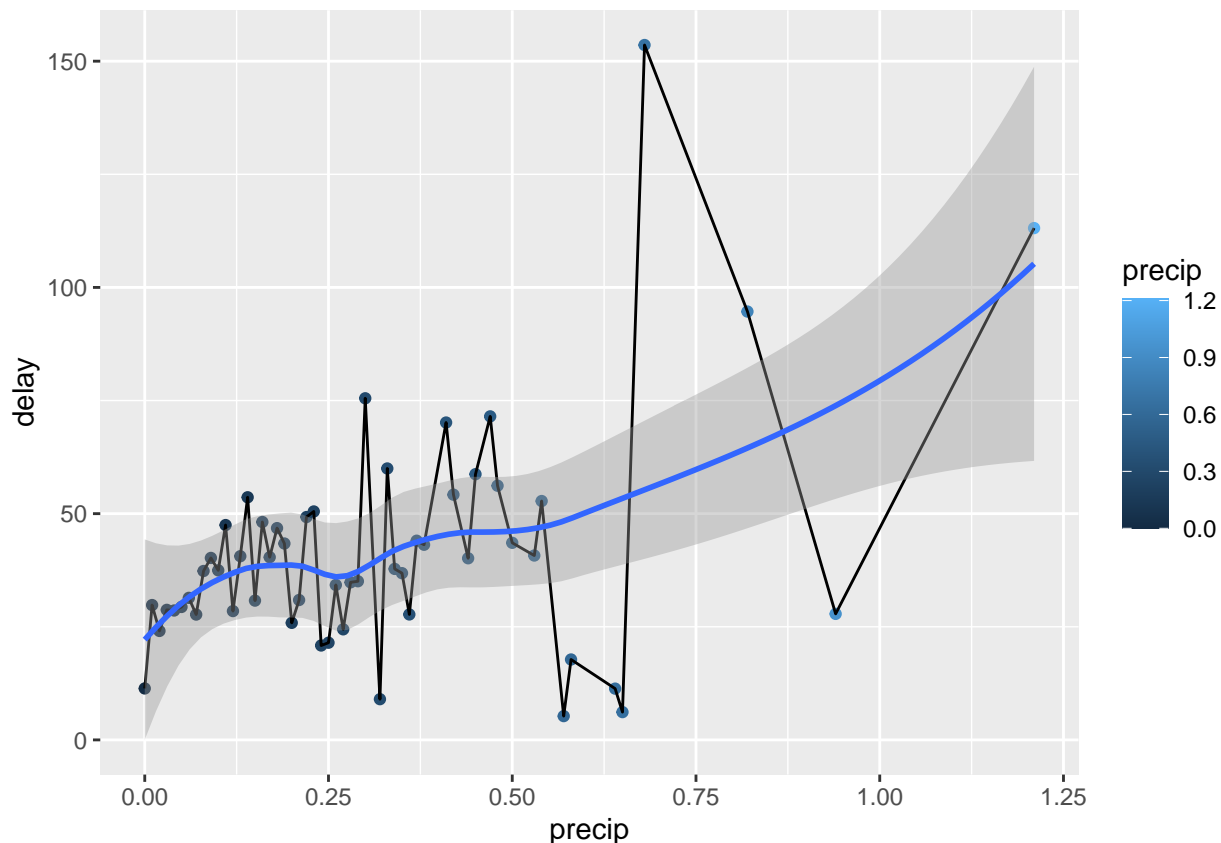
- ¿En qué condiciones meteorológicas es más probable encontrar un retraso de avión?

Por encima de 0.02 pulgadas de precipitación ya podemos notar una clara tendencia en el aumento del retraso.

```
# Hacemos un inner_join entre flight y weather
# Filtrando por origin, year, month, day y hour.

flights %>%
  inner_join(weather,
    by = c(
      "origin" = "origin",
      "year" = "year",
      "month" = "month",
      "day" = "day",
      "hour" = "hour"
    )) %>%
  # Agrupamos por precipitaciones
  group_by(precip) %>%
  summarise(delay = mean(dep_delay, na.rm = TRUE)) %>%
  # Graficamos
  ggplot(aes(precip, delay)) +
  geom_point(aes(col = precip)) +
  geom_line()+
  geom_smooth()
```

```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



- ¿Qué ocurrió el 13 de Junio de 2013? Representa la distribución de retrasos espaciales y luego usa Google para encontrar alguna referencia del tiempo ese día.

Hubo una gran serie de tormentas (derechos) en el sureste de los EE. UU. (consulte la serie derecho del 12 y 13 de junio de 2013).

```
flights %>%
  filter(year == 2013, month == 6, day == 13) %>%
  group_by(dest) %>%
  summarise(delay = mean(arr_delay, na.rm = TRUE)) %>%
  inner_join(airports, by = c("dest" = "faa"))
```

```
## # A tibble: 85 x 9
##   dest delay name          lat lon alt tz dst tzone
##   <chr> <dbl> <chr>          <dbl> <dbl> <dbl> <dbl> <chr> <chr>
## 1 ABQ    64 Albuquerque International S~ 35.0 -107. 5355 -7 A Amer~
## 2 ACK     8 Nantucket Mem          41.3 -70.1  48 -5 A Amer~
## 3 ALB   -19 Albany Intl          42.7 -73.8  285 -5 A Amer~
## 4 ATL  109. Hartsfield Jackson Atlanta ~ 33.6 -84.4 1026 -5 A Amer~
## 5 AUS  64.3 Austin Bergstrom Intl    30.2 -97.7  542 -6 A Amer~
## 6 AVL   90 Asheville Regional Airport  35.4 -82.5 2165 -5 A Amer~
## 7 BDL  18.5 Bradley Intl          41.9 -72.7  173 -5 A Amer~
## 8 BGR  102 Bangor Intl          44.8 -68.8  192 -5 A Amer~
## 9 BHM  158 Birmingham Intl          33.6 -86.8  644 -6 A Amer~
## 10 BNA  95.9 Nashville Intl          36.1 -86.7  599 -6 A Amer~
## # ... with 75 more rows
```

## Pregunta 7

- ¿Qué significa que un vuelo no tenga tailnum? ¿Qué significan los números de cola de los vuelos que no aparecen en la tabla planes? Con una variable o dos debería bastarte para responder.

Son vuelos que no tienen cola tampoco tienen hora de llegada, por lo tanto deben ser cancelados.

```
flights %>%
  select(tailnum, arr_time) %>%
  filter(is.na(tailnum)) %>%
  summary()
```

```
##      tailnum          arr_time
## Length:2512      Min.   : NA
## Class :character 1st Qu.: NA
## Mode  :character Median : NA
##                  Mean    :NaN
##                  3rd Qu.: NA
##                  Max.    : NA
##                  NA's    :2512
```

- Filtra los vuelos para mostrar solo los aviones que han volado más de 100 veces en el año.

```
flights %>%
  filter(!is.na(tailnum)) %>%
  group_by(tailnum) %>%
  count() %>%
  filter(n>=100)
```

```
## # A tibble: 1,217 x 2
## # Groups:   tailnum [1,217]
##   tailnum      n
##   <chr>   <int>
## 1 NOEGMQ    371
## 2 N10156    153
## 3 N10575    289
## 4 N11106    129
## 5 N11107    148
## 6 N11109    148
## 7 N11113    138
## 8 N11119    148
## 9 N11121    154
## 10 N11127   124
## # ... with 1,207 more rows
```

- Encuentra en el dataset de vuelos las 48 horas de todo el año que han tenido más retrasos. Cruza la información con la tabla del tiempo weather para explicar lo sucedido. ¿Observas algún patrón?

```
peores_horas <- flights %>%
  mutate(hora = sched_dep_time %/% 100) %>%
  group_by(origin, year, month, day, hour) %>%
```

```
summarise(atraso_salida = mean(dep_delay, na.rm = TRUE)) %>%
ungroup() %>%
arrange(desc(atraso_salida)) %>%
slice(1:48)
```

## 'summarise()' has grouped output by 'origin', 'year', 'month', 'day'. You can  
## override using the '.groups' argument.

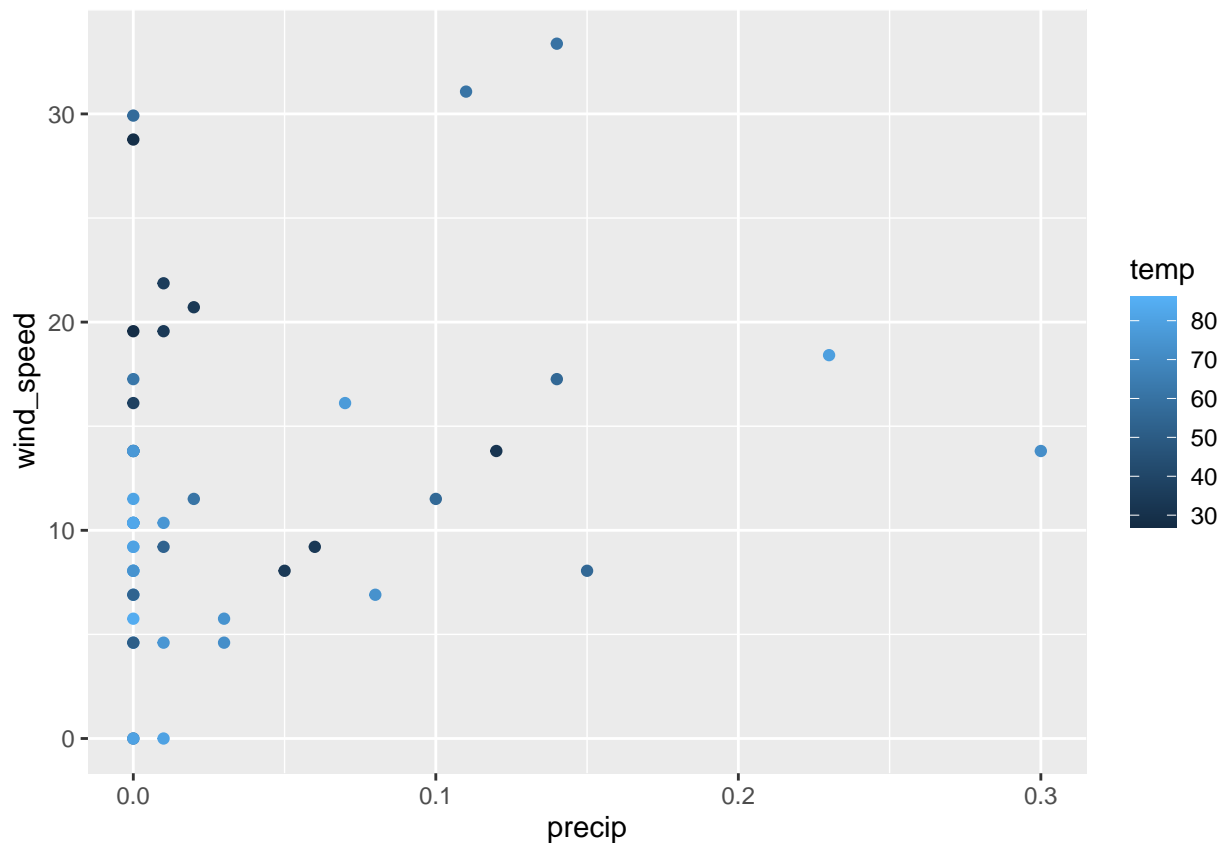
```
clima_peores_horas <- semi_join(weather, peores_horas,
                                by = c("origin", "year", "month", "day", "hour"))

select(clima_peores_horas, temp, wind_speed, precip) %>%
print(n = 48)
```

```
## # A tibble: 48 x 3
##   temp wind_speed precip
##   <dbl>      <dbl> <dbl>
## 1 27.0      13.8    0
## 2 28.0      19.6    0
## 3 28.9      28.8    0
## 4 33.8       9.21  0.06
## 5 34.0       8.06  0.05
## 6 80.1       8.06  0
## 7 86        13.8    0
## 8 73.4       6.90  0.08
## 9 84.0       5.75  0
## 10 78.8      18.4   0.23
## 11 53.6       0     0
## 12 60.8      31.1   0.11
## 13 55.4      17.3   0.14
## 14 53.1       9.21  0.01
## 15 55.9      11.5   0.1
## 16 55.4       8.06  0.15
## 17 57.0      29.9    0
## 18 33.8      20.7   0.02
## 19 34.0      19.6   0.01
## 20 36.0      21.9   0.01
## 21 37.9      16.1    0
## 22 32        13.8   0.12
## 23 60.1      33.4   0.14
## 24 60.8      11.5   0.02
## 25 62.1      17.3    0
## 26 66.9      10.4    0
## 27 66.9      13.8    0
## 28 79.0      10.4    0
## 29 77        16.1   0.07
## 30 75.9      13.8    0
## 31 82.4       8.06  0
## 32 86         9.21  0
## 33 80.1       9.21  0
## 34 80.6      11.5    0
## 35 78.1       6.90  0
## 36 75.2      10.4   0.01
```

```
## 37 73.9      5.75 0.03
## 38 73.9      8.06 0
## 39 75.0      4.60 0
## 40 75.0      4.60 0.01
## 41 80.1      0     0.01
## 42 80.1      0     0
## 43 77        10.4 0
## 44 82.0      10.4 0
## 45 72.0      13.8 0.3
## 46 72.0      4.60 0.03
## 47 51.1      4.60 0
## 48 54.0      6.90 0
```

```
ggplot(clima_peores_horas, aes(x = precip, y = wind_speed, color = temp)) +
  geom_point()
```



- ¿Qué nos indica la operación `anti_join(flights, airports, by = c("dest"="faa"))`?
- ¿Qué nos indica la operación `anti_join(airports, flights, by = c("faa"="dest"))`?
- ¿Crees que cada avión pertenece a una sola aerolínea? Esa es mi intuición. Confirma o rechaza esta hipótesis con las herramientas que te he enseñado en esta última sección.

## Pregunta 8

Combina las tablas de `fueleconomy::vehicles` con `fueleconomy::common` para encontrar los registros de los coches más comunes.