Lab04

Vivian Yeh

Part I: Understanding the Context of the Data

```
library(stat20data)
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.3.6      v purrr   0.3.4
v tibble  3.1.8      v dplyr   1.0.10
v tidyr   1.2.0      v stringr 1.4.1
v readr   2.1.2      v forcats 0.5.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(dplyr)
data(flights)
```

1. **What is the unit of observation in the data frame on the handout?**

   One flight from SFO or OAK

2. **Which variables are categorical?**

   Carrier, flight, tailnum, origin, dest,

3. **Which variables are numerical?**

   year, month, day, dep_time, sched_dep_time, dep_delay, arr_time, sched_arr_time, arr_delay, air_time, distance, hour, minute, time_hour

4. **Do any of the variable have ambiguous data types to you?**

   year, month, day, dep_time, sched_dep_time, dep_delay, arr_time, sched_arr_time, arr_delay, air_time, distance, hour, minute, time_hour

5. **Is there any discernible pattern to the manner in which the rows are ordered?**

   The departure time(dep_time)

6. **What is your guess for the units/format used to record the departure time? Said another way, what would a value of 1517 represent?**

   In a 24 hours time frame representation

7. **What filter would you use to extract the flights that left in the springtime?**

   Months from March to May

```
spring <- filter(flights, month >= 3 & month <= 5)
```

Part II: Computing on the Data

8. **filter()**: Filter the data set to contain only the flights that went to Portland, Oregon and print the first few rows of the data frame. How many were there in 2020?

```
data_8 <- filter(flights, dest == "PDX" | dest == "ORE")
data_8
```

```
# A tibble: 3,882 x 19
    year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
   <dbl> <dbl> <dbl>    <dbl>      <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>
 1  2020     1     1      613        600      13     801     754       7 AS
 2  2020     1     1      656        700      -4     842     854     -12 AS
 3  2020     1     1      657        700      -3     836     845      -9 WN
 4  2020     1     1      825        830      -5    1024    1024       0 UA
 5  2020     1     1      900        900       0    1116    1050      26 OO
 6  2020     1     1     1055       1055       0    1240    1240       0 WN
 7  2020     1     1     1110       1115      -5    1327    1319       8 UA
 8  2020     1     1     1129       1135      -6    1354    1329      25 AS
 9  2020     1     1     1246       1248      -2    1443    1443       0 UA
10  2020     1     1     1304       1305      -1    1518    1459      19 AS
# ... with 3,872 more rows, 9 more variables: flight <dbl>, tailnum <chr>,
#   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
#   minute <dbl>, time_hour <dttm>, and abbreviated variable names
```

```
#   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
#   5: arr_delay
```

```
nrow(data_8)
```

```
[1] 3882
```

There are 3882 flights in 2020.

9. **mutate()**: Mutate a new variable called `avg_speed` that is the average speed of the plane during the flight, measured in miles per hour. (Look through the column names or the help file to find variables that can be used to calculate this.)

```
avg_speed <- mutate(flights, avg_speed = (distance / (air_time / 60)))
select(avg_speed, avg_speed)
```

```
# A tibble: 120,605 x 1
   avg_speed
       <dbl>
 1      516.
 2      464.
 3      540
 4      524.
 5      558.
 6      518.
 7      519.
 8      501.
 9      446.
10      560
# ... with 120,595 more rows
```

10. **arrange()**: Arrange the data set to figure out: which flight holds the record for longest departure delay (in hrs) and what was its destination? What was the destination and delay time (in hrs) for the flight that was least delayed, i.e. that left the most ahead of schedule?

```
longest_delay <- arrange(flights, desc(dep_delay))
longest_delay
```

```
# A tibble: 120,605 x 19
    year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
   <dbl> <dbl> <dbl>    <dbl>      <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>
```

```
 1  2020     3     6     1407        907    1740    1722    1213    1749 AA
 2  2020     2    20     1604       1245    1639    1900    1538    1642 AA
 3  2020     3     2     1247       1140    1507    2049    1958    1491 AA
 4  2020     2    12      955        907    1488    1246    1215    1471 AA
 5  2020     1    24     1005        952    1453    1303    1245    1458 AA
 6  2020     3     6      816       1111    1265    1611    1914    1257 AA
 7  2020     2    29     1046       1403    1243    1221    1529    1252 OO
 8  2020     2    12      828       1253    1175    1647    2124    1163 AA
 9  2020     2    14      655       1125    1170    1015    1435    1180 AA
10  2020     1    13     1238       1800    1118    1423    1933    1130 OO
# ... with 120,595 more rows, 9 more variables: flight <dbl>, tailnum <chr>,
#   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
#   minute <dbl>, time_hour <dttm>, and abbreviated variable names
#   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
#   5: arr_delay
```

```r
least_delay <- arrange(flights, dep_delay)
least_delay
```

```
# A tibble: 120,605 x 19
    year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
   <dbl> <dbl> <dbl>    <dbl>      <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>
 1  2020     3    31     1930       2010     -40    2119    2222     -63 OO
 2  2020     3    28     1540       1615     -35    2329      48     -79 UA
 3  2020    11    19     2341         16     -35     509     558     -49 UA
 4  2020     3    20     1303       1334     -31    1424    1446     -22 OO
 5  2020     5    24      804        834     -30    1115    1144     -29 G4
 6  2020     3    30     1851       1920     -29    2033    2100     -27 OO
 7  2020     9    10     1834       1903     -29    1959    2030     -31 G4
 8  2020     4     3     2057       2125     -28    2223    2301     -38 AS
 9  2020     6    26     2045       2113     -28    2243    2312     -29 G4
10  2020     3    17     1708       1735     -27    1840    1909     -29 AS
# ... with 120,595 more rows, 9 more variables: flight <dbl>, tailnum <chr>,
#   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
#   minute <dbl>, time_hour <dttm>, and abbreviated variable names
#   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
#   5: arr_delay
```

The flight 576 holds the record for longest departure delay and its destination is PHX.

The flight that was least delayed has the destination to GEG, and its delay time is 0.667 hours

11. **summarize()**: Confirm the records for departure delay from the question above by summarizing that variable by its maximum and its minimum value.

```
flights %>%
  summarize(dep_longest = max(dep_delay,na.rm = TRUE), dep_least = min(dep_delay, na.
```

```
# A tibble: 1 x 2
  dep_longest dep_least
        <dbl>     <dbl>
1        1740       -40
```

12. How many flights left SFO during March 2020?

```
data_12 <- filter(flights, month == 3 & origin == "SFO")
nrow(data_12)
```

```
[1] 14165
```

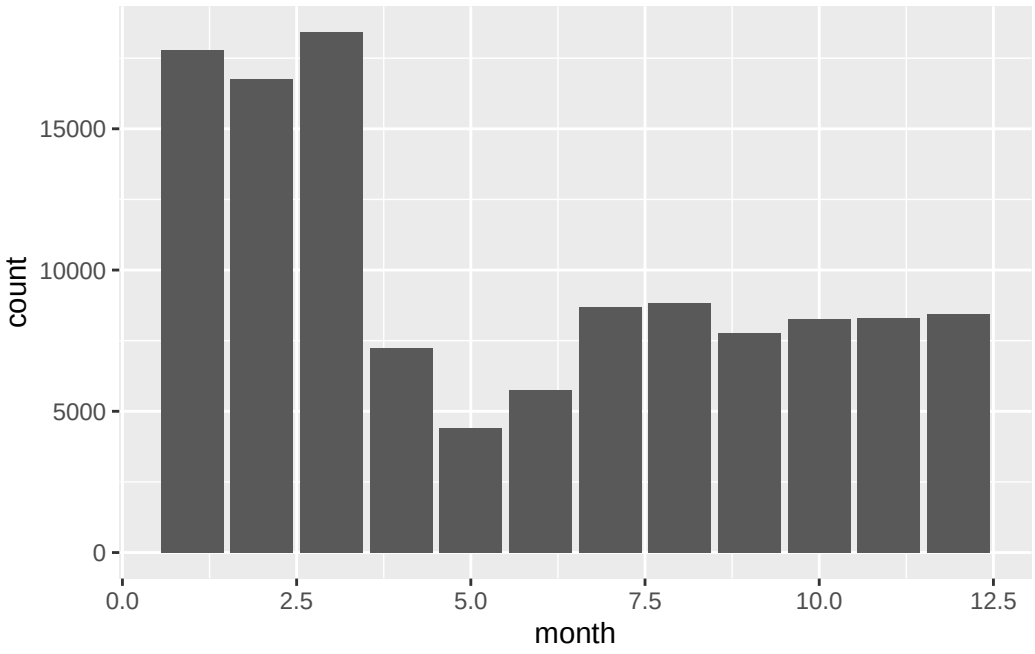
13. How many flights left SFO during April 2020?

```
data_13 <- filter(flights, month == 4 & origin == "SFO")
nrow(data_13)
```

```
[1] 4517
```

14. Create a bar chart that shows the distribution by month of all the flights leaving the Bay Area (SFO and OAK). Do you any sign of an effect of the pandemic?

```
flights %>%
  filter(origin == "SFO" | origin == "OAK") %>%
  ggplot(aes(x = month))+
  geom_bar()
```

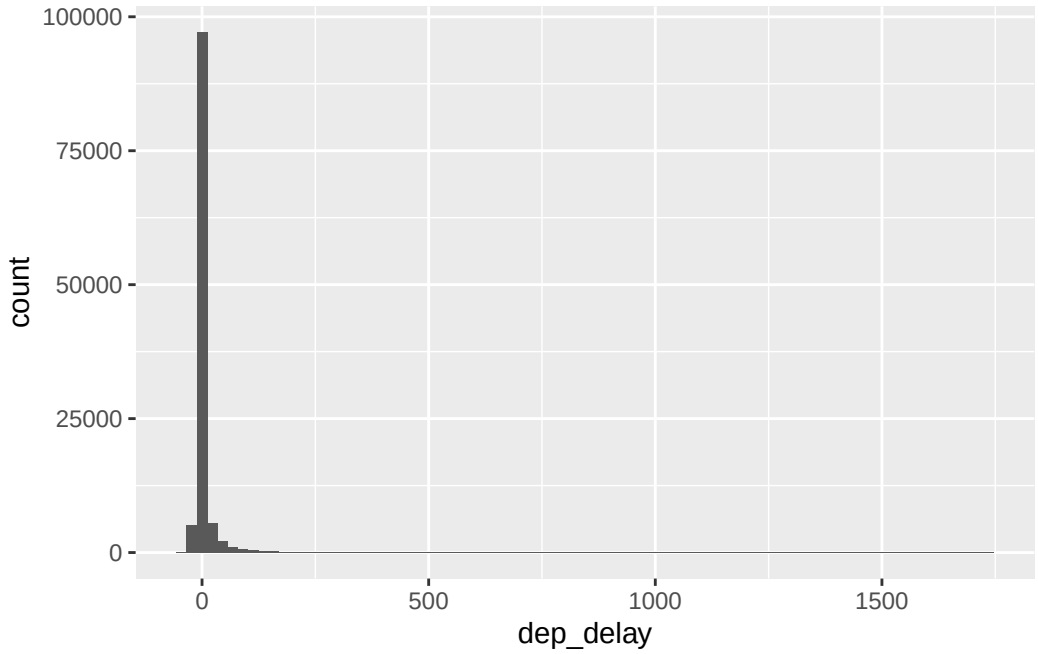


The graph shows that there is a huge decrease between March and April, and this results from the pandemic. We can see that the counts of flights decreased and stayed at a low number after March.

15. Create a histogram showing the distribution of departure delays for all flights. Describe in words the shape and modality of the distribution and, using numerical summaries, (i.e. summary statistics) its center and spread. Be sure to use measures of center and spread that are most appropriate for this type of distribution. Also set the limits of the x-axis to focus on where most of the data lie.

```
data_15 <- ggplot(flights, aes(x = dep_delay))+
  geom_histogram(bins = 80)
data_15
```

```
Warning: Removed 7369 rows containing non-finite values (stat_bin).
```




```
median_delay <- median(flights$dep_delay, na.rm = TRUE)
median_delay
```

```
[1] -4
```

```
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}

mode_delay <- getmode(flights$dep_delay)
mode_delay
```

```
[1] -5
```

```
min_delay <- min(flights$dep_delay, na.rm = TRUE)
min_delay
```

```
[1] -40
```

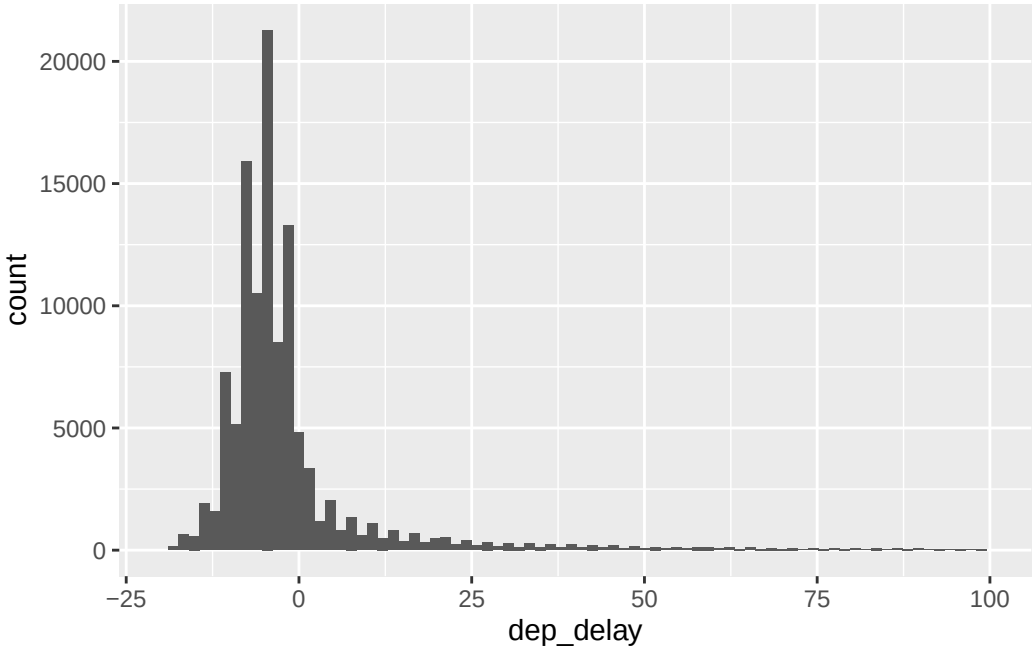
```r
max_delay <- max(flights$dep_delay, na.rm = TRUE)
max_delay
```

```
[1] 1740
```

```r
data_15_xset <- data_15 + xlim(-20, 100)
data_15_xset
```

```
Warning: Removed 9098 rows containing non-finite values (stat_bin).
```

```
Warning: Removed 2 rows containing missing values (geom_bar).
```



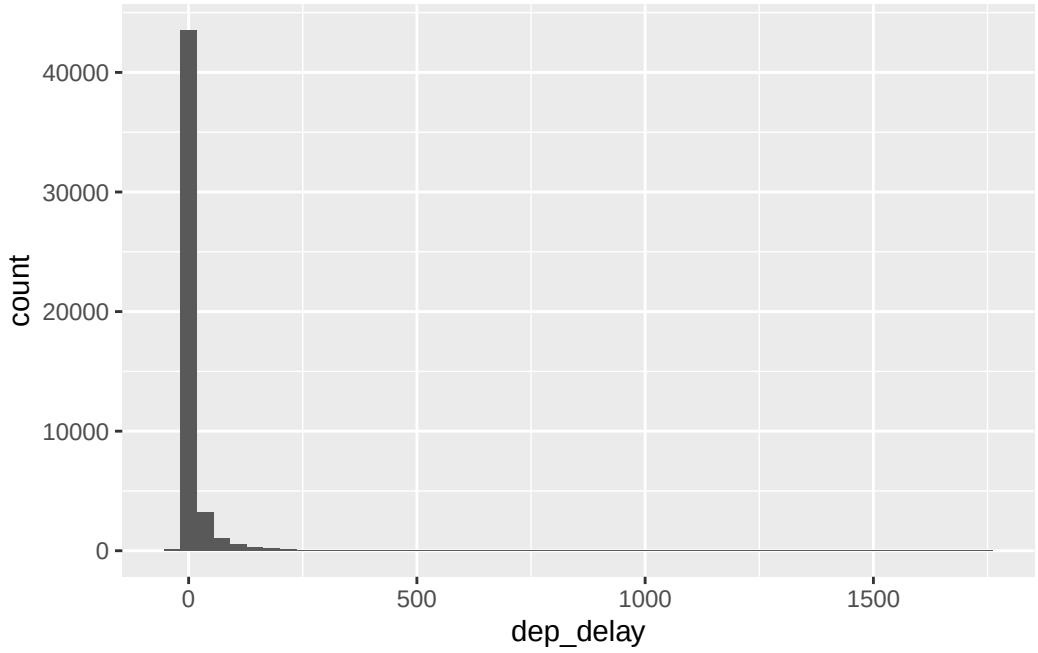
The distribution is skewed to right. Because there are outliers, using median to calculate the center of dep_delay is the most appropriate way. For the median of dep_delay is -4, and the mode of dep_delay is -5, which is the most of data lie.

16. Add a new column to your data frame called `before_times` that takes values of `TRUE` and `FALSE` indicating whether the flight took place up through the end of March or after April 1st, respectively. Remake the histograms above, but now separated into two subplots: one with the departure delays from the before times, the other with the flights from afterwards.

    Can you visually detect any difference in the distribution of departure delays?

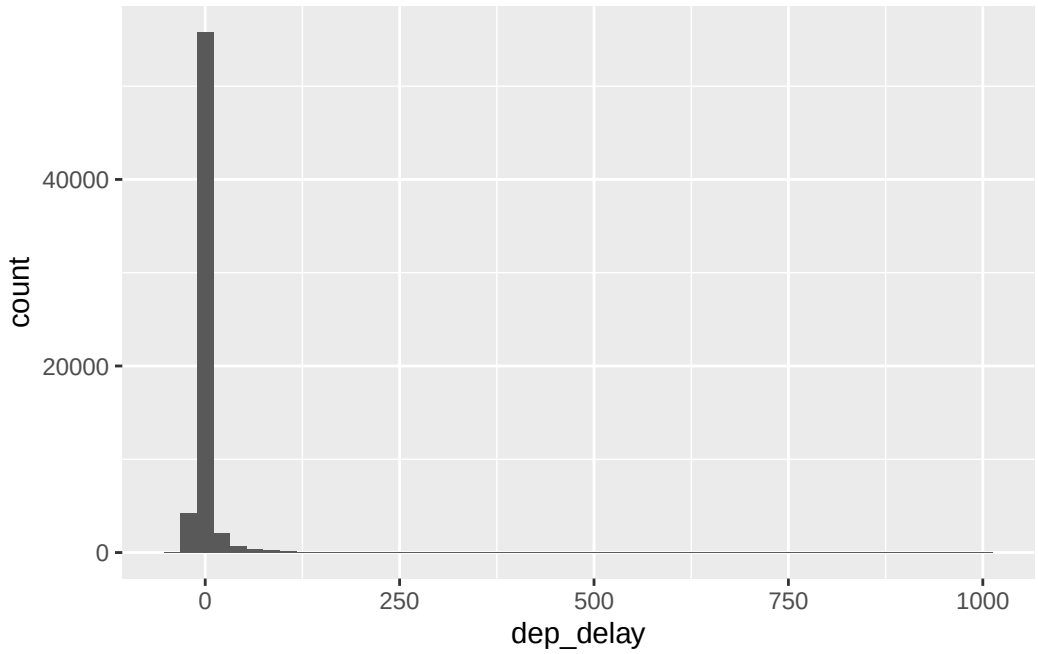
```r
flights <- mutate(flights, before_times = ifelse(month <= 3, TRUE, FALSE))

flights %>%
  filter(before_times == TRUE) %>%
  ggplot(aes(x = dep_delay))+
  geom_histogram(bins = 50)
```

```
Warning: Removed 3723 rows containing non-finite values (stat_bin).
```




```r
flights %>%
  filter(before_times == FALSE) %>%
  ggplot(aes(x = dep_delay))+
  geom_histogram(bins = 50)
```

```
Warning: Removed 3646 rows containing non-finite values (stat_bin).
```



Yes, the difference between the distribution of departure delays is visible but not too different. The flights took place after April 1st tend to have less delays, and more flights arrived before the schedule time.

17. If you flew out of OAK or SFO during this time period, what is the tail number of the plane that you were on? If you did not fly in this period, find the tail number of the plane that flew JetBlue flight 40 to New York's JFK Airport from SFO on May 1st.

```
data_17 <- filter(flights, origin == "OAK" | origin == "SFO")
data_17 <- filter(data_17, before_times == TRUE)
tail_number <- select(data_17, tailnum)

data_17
```

```
# A tibble: 52,943 x 20
    year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
   <dbl> <dbl> <dbl>    <dbl>      <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>
 1  2020     1     1        8       2359       9     528     532      -4 UA
 2  2020     1     1       29         39     -10     356     420     -24 F9
 3  2020     1     1       37         40      -3     846     856     -10 UA
 4  2020     1     1       41         45      -4     908     913      -5 AA
 5  2020     1     1       44       2300     104     834     709      85 AA
 6  2020     1     1       48         56      -8     641     658     -17 UA
```

```
 7  2020     1     1       49         56      -7     614     634     -20 UA
 8  2020     1     1      506        515      -9    1050    1101     -11 UA
 9  2020     1     1      528        530      -2     812     820      -8 WN
10  2020     1     1      540        536       4    1303    1332     -29 AA
# ... with 52,933 more rows, 10 more variables: flight <dbl>, tailnum <chr>,
#   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
#   minute <dbl>, time_hour <dttm>, before_times <lgl>, and abbreviated
#   variable names 1: sched_dep_time, 2: dep_delay, 3: arr_time,
#   4: sched_arr_time, 5: arr_delay
```

```
tail_number
```

```
# A tibble: 52,943 x 1
   tailnum
   <chr>
 1 N76522
 2 N342FR
 3 N17126
 4 N907AA
 5 N165US
 6 N77865
 7 N79521
 8 N36472
 9 N567WN
10 N549UW
# ... with 52,933 more rows
```

```
data_not <- filter(flights, before_times == FALSE, carrier == "B6", flight == 40, ori
tail_number_15 <- select(data_not, tailnum)
tail_number_15
```

```
# A tibble: 1 x 1
  tailnum
  <chr>
1 N982JB
```

The tail number of the plane that flew JetBlue flight 40 to New York's JFK Airport from SFO on May 1st is N982JB.

18. What proportion of the flights left on or ahead of schedule?

```
data_18_left <- filter(flights, dep_delay > 0)
num_left <- nrow(data_18_left)
num_total <- nrow(flights)

data_18_ahead <- filter(flights, dep_delay < 0)
num_ahead <- nrow(data_18_ahead)

prop_left <- num_left / num_total
prop_ahead <- num_ahead / num_total

prop_left
```

```
[1] 0.177132
```

```
prop_ahead
```

```
[1] 0.7219518
```

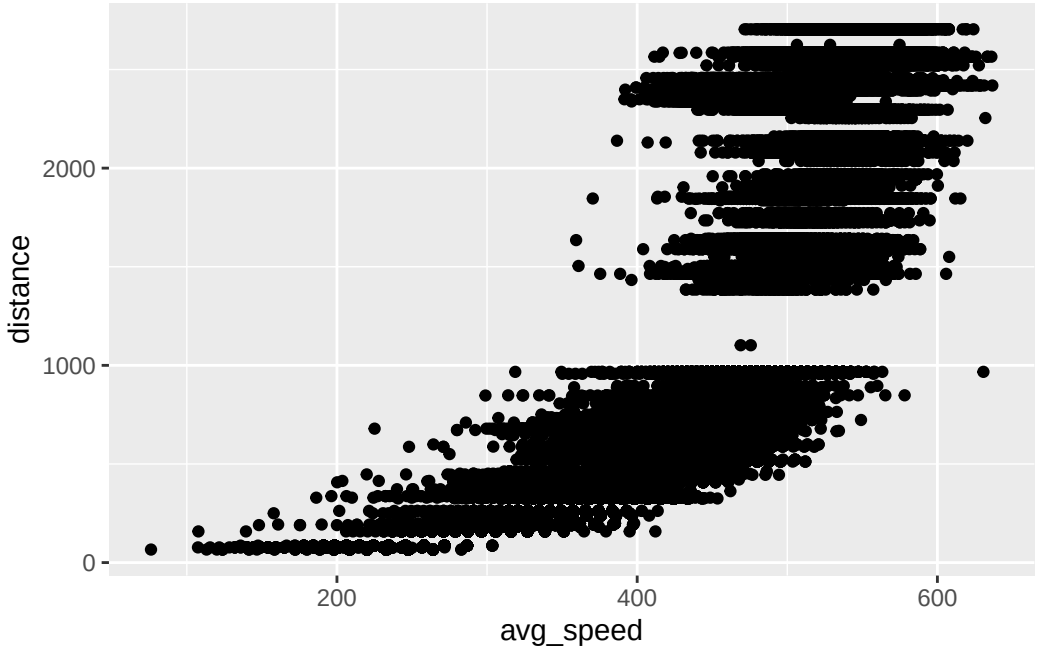
The proportion of the flights left on schedule is 0.177.

The proportion of the flights ahead of schedule is 0.722.

19. Create a plot that captures the relationship of average speed vs. distance and describe the shape and structure that you see. What phenomena related to taking flights from the Bay Area might explain this structure?

```
ggplot(avg_speed, aes(x = avg_speed, y = distance))+
  geom_point()
```

```
Warning: Removed 7592 rows containing missing values (geom_point).
```



describe the shape and structure that you see. What phenomena related to taking flights from the Bay Area might explain this structure?

The shape of this plot is like a upward line, and every distance has its fixed range of speed. Overall, the longer distances, the higher average speed. There seems to have a upper limitation of speed because of the current technology of the air crafts. Also, the air planes tend to fly in the different layers of the air such as mesosphere, stratosphere.

20. For OAK and SFO separately, what proportion of the flights left on or ahead of schedule?

```
data_oak <- filter(flights, origin == "OAK")
data_oak_left <- filter(data_oak, dep_delay > 0)
data_oak_ahead <- filter(data_oak, dep_delay < 0)
num_oak_left <- nrow(data_oak_left)
num_oak_ahead <- nrow(data_oak_ahead)
num_oak_total <- nrow(data_oak)
prop_oak_left <- num_oak_left / num_oak_total
prop_oak_ahead <- num_oak_ahead / num_oak_total

prop_oak_left
```

```
[1] 0.1910865
```

```
prop_oak_ahead
```

```
[1] 0.6766204
```

```
data_sfo <- filter(flights, origin == "SFO")
data_sfo_left <- filter(data_sfo, dep_delay > 0)
data_sfo_ahead <- filter(data_sfo, dep_delay < 0)
num_sfo_left <- nrow(data_sfo_left)
num_sfo_ahead <- nrow(data_sfo_ahead)
num_sfo_total <- nrow(data_sfo)
prop_sfo_left <- num_sfo_left / num_sfo_total
prop_sfo_ahead <- num_sfo_ahead / num_sfo_total

prop_sfo_left
```

```
[1] 0.1722599
```

```
prop_sfo_ahead
```

```
[1] 0.7377788
```

For OAK, the proportion of the flights left on schedule is 0.191, and the proportion of the flights ahead of schedule is 0.677.

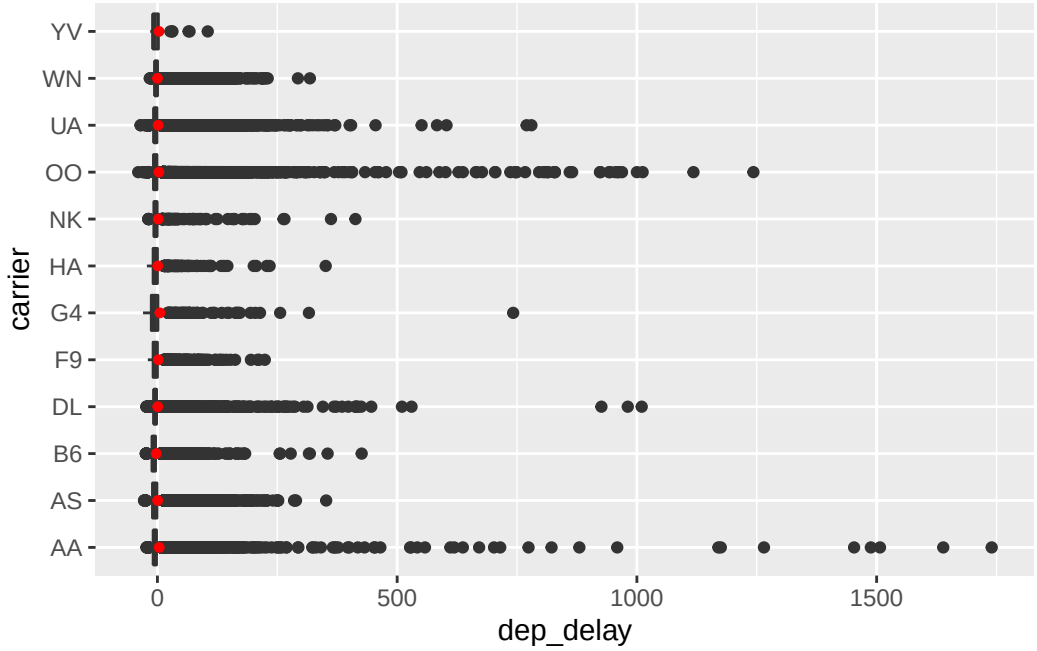
For SFO, the proportion of the flights left on schedule is 0.172, and the proportion of the flights ahead of schedule is 0.738.

21. Create a data frame that contains the median and interquartile range for departure delays, grouped by carrier. Which carrier has the lowest typical departure delay? Which one has the least variable departure delays?

```
ggplot(flights, aes(x = dep_delay, y = carrier))+
  geom_boxplot() +
  stat_summary(fun=mean, geom="point", shape=20, size=2, color="red", fill="red")
```

```
Warning: Removed 7369 rows containing non-finite values (stat_boxplot).
```

```
Warning: Removed 7369 rows containing non-finite values (stat_summary).
```




```
flights %>%
  group_by(carrier) %>%
  summarize(mean(dep_delay, na.rm = TRUE), IQR(dep_delay, na.rm = TRUE))
```

```
# A tibble: 12 x 3
   carrier `mean(dep_delay, na.rm = TRUE)` `IQR(dep_delay, na.rm = TRUE)`
   <chr>                             <dbl>                          <dbl>
 1 AA                                3.79                               6
 2 AS                                0.564                              9
 3 B6                               -2.44                               7
 4 DL                                0.867                              5
 5 F9                                1.98                               9
 6 G4                                5.23                              13.8
 7 HA                                0.931                              9
 8 NK                                1.86                               6
 9 OO                                3.14                               7
10 UA                                1.62                               6
11 WN                                0.237                              5
12 YV                                3.41                              11
```

```
data_21 <- flights
data_21 %>%
```

```
    summarize(min_delay = min(dep_delay, na.rm = TRUE))
```

```
# A tibble: 1 x 1
  min_delay
      <dbl>
1       -40
```

```
  data_21 %>%
    filter(dep_delay == min_delay) %>%
    select(carrier)
```

```
# A tibble: 1 x 1
  carrier
  <chr>
1 OO
```

```
  data_21 <- group_by(flights, carrier)
  data_21 <- mutate(data_21, var_delay = var(dep_delay, na.rm = TRUE))
  min_var_dep = min(data_21$var_delay)
  data_21 %>%
    filter(var_delay == min_var_dep) %>%
    select(carrier)
```

```
# A tibble: 35,267 x 1
# Groups:   carrier [1]
   carrier
   <chr>
 1 WN
 2 WN
 3 WN
 4 WN
 5 WN
 6 WN
 7 WN
 8 WN
 9 WN
10 WN
# ... with 35,257 more rows
```

```
#flights <- mutate(flights, var_dep_delay = var(dep_delay, na.rm = TRUE))

  #summarize(min_var = min(var_dep_delay)) %>%
  #filter(var_dep_delay == min_var) %>%
  #select(carrier)

#data_21 %>%
  #filter(var_dep_delay == min_var) %>%
#  select(carrier)

#flights %>%
#  group_by(carrier) %>%
#  summarize(min(dep_delay, na.rm = TRUE))
```

The carrier OO has the lowest typical departure delay.

The carrier WN has the least variable departure delays.

Part III: Extensions

22. For flights leaving SFO, which month has the highest average departure delay? What about the highest median departure delay? Which of these measures is more useful to know when deciding which month(s) to avoid flying if you particularly dislike flights that are severely delayed?

```
data_22 <- filter(flights, origin == "SFO")
data_22 %>%
  group_by(month) %>%
  summarize(avg_dep_delay = mean(dep_delay, na.rm = TRUE))
```

```
# A tibble: 12 x 2
   month avg_dep_delay
   <dbl>         <dbl>
 1     1          9.05
 2     2          8.06
 3     3          1.15
 4     4         -3.81
 5     5         -2.68
 6     6         -1.86
 7     7         -1.56
 8     8         -1.80
```

```
 9     9        -1.75
10    10        -0.155
11    11        -1.91
12    12        -0.970
```

```
data_22 %>%
  group_by(month) %>%
  summarize(med_dep_delay = median(dep_delay, na.rm = TRUE))
```

```
# A tibble: 12 x 2
   month med_dep_delay
   <dbl>         <dbl>
 1     1            -3
 2     2            -3
 3     3            -5
 4     4            -7
 5     5            -6
 6     6            -5
 7     7            -6
 8     8            -5
 9     9            -5
10    10            -5
11    11            -6
12    12            -5
```

For flights leaving SFO, January has the highest average departure delay.

January and Feburary have the highest median departure delay.

Evaluating the median departure delay is more useful to know when deciding which month(s) to avoid flying because it is not sensitive to the outliers.

23. Each individual airplane can be uniquely identified by its tailnumber in the same way that US citizens can be by their social security numbers. Which airplane flew the farthest during this year for which we have data? How many times around the planet does that translate to?

```
flights %>%
  filter(tailnum != "NA") %>%
  group_by(tailnum) %>%
  summarize(total_dist = sum(distance, na.rm = TRUE)) %>%
  filter(total_dist == max(total_dist))
```

```
# A tibble: 1 x 2
  tailnum total_dist
  <chr>        <dbl>
1 N705TW      245670
```

```
flights %>%
  filter(tailnum != "NA") %>%
  group_by(tailnum) %>%
  summarize(total_dist = sum(distance, na.rm = TRUE)) %>%
  summarize(max_total_dist = max(total_dist)) %>%
  summarize(round = max_total_dist / 24901.461)
```

```
# A tibble: 1 x 1
  round
  <dbl>
1  9.87
```

N705TW flew the farthest during this year for which we have data.

And, that is 9.865686 times around the planet.

24. What is the tailnumber of the fastest plane in the data set? What type of plane is it (google it!)? Be sure to be clear how you're defining fastest.

```
flights %>%
  filter(tailnum != "NA", air_time != 0) %>%
  group_by(tailnum) %>%
  summarize(sum_dist = sum(distance, na.rm = TRUE), sum_time = sum(air_time, na.rm =
  summarize(tailnum = tailnum, avg_speed = sum_dist / sum_time) %>%
  filter(avg_speed == max(avg_speed))
```

```
# A tibble: 1 x 2
  tailnum avg_speed
  <chr>       <dbl>
1 N823NN       10.1
```

N823NN is the tailnumber of the fastest plane in the data set. It is Boeing 737-823 which type code B738 and mode S AB3D39.

25. Using the airport nearest your hometown, which day of the week and which airline seems best for flying there from San Francisco (if you're from near SFO or OAK or from abroad, use Chicago as your hometown)? Be clear on how you're defining *best*. (note that there is no explicit weekday column in this data set, but there is sufficient information to piece it together. The following line of code can be added to your pipeline to create that new

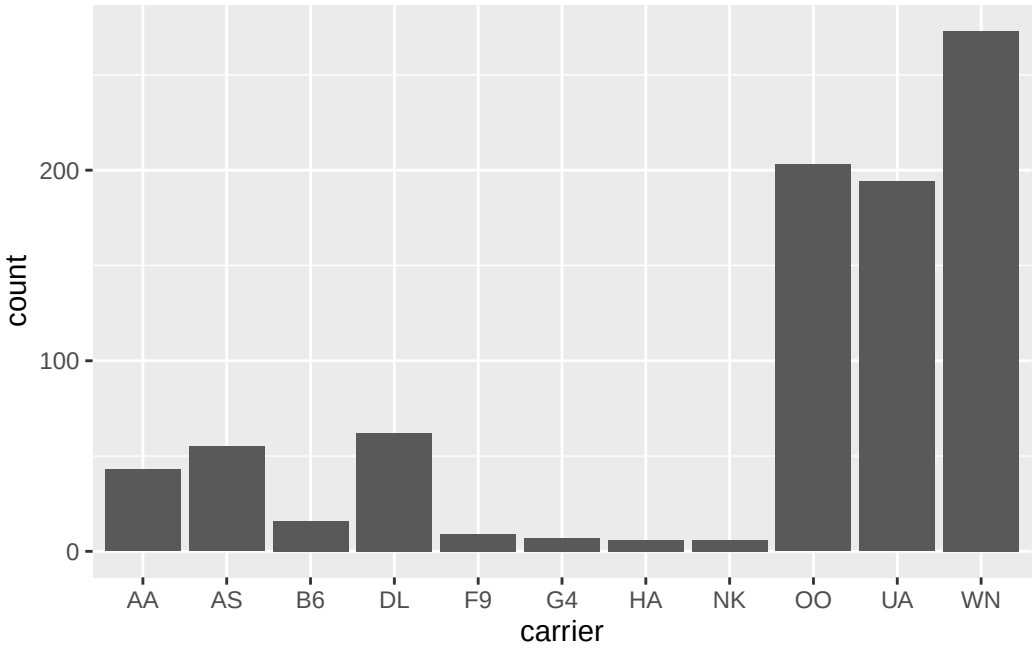
column. It uses functions in the `lubridate` package, so be sure to load it in at the start of this exercise).

```
library(lubridate)
```

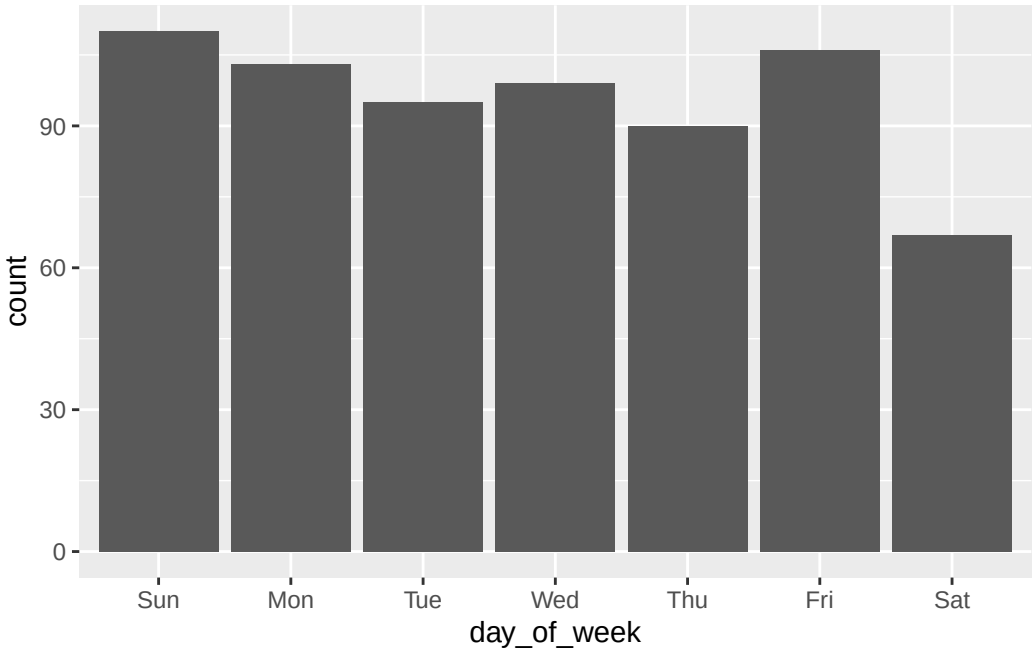
```
Attaching package: 'lubridate'
```

```
The following objects are masked from 'package:base':

    date, intersect, setdiff, union
```

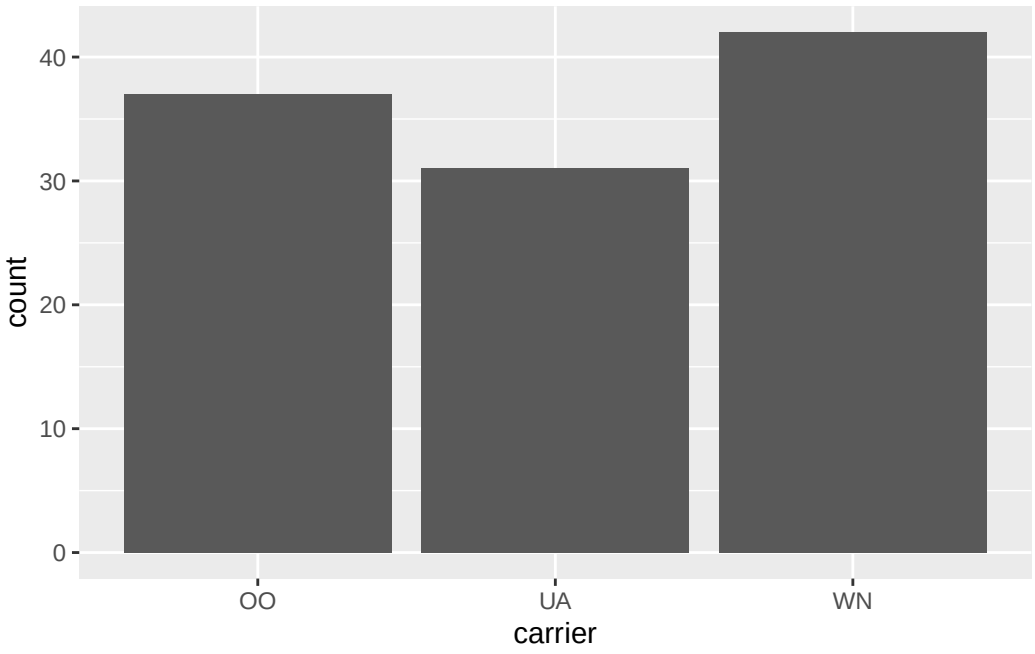
```
data_25 <- filter(flights, dest == "SNA", tailnum != "NA", )
data_25 <- mutate(flights, day_of_week = wday(ymd(paste(year, month, day, set = "-"))
data_25 <- mutate(data_25, delay_time = (dep_delay + arr_delay) / 2)
data_25 <- filter(data_25, delay_time == 0)
data_25 <- arrange(data_25, carrier, day_of_week)

data_25 <- group_by(data_25, carrier, day_of_week)
ggplot(data_25, aes(x = carrier)) +
  geom_bar()
```



```
data_25 <- filter(data_25, carrier == "OO" |carrier == "UA"| carrier == "WN")
ggplot(data_25, aes(x = day_of_week)) +
  geom_bar()
```

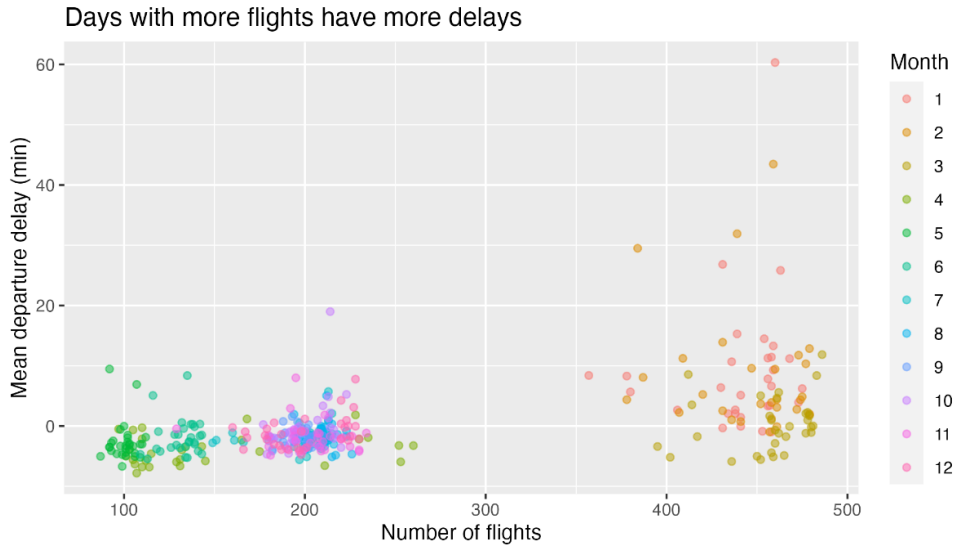



```
data_25 <- filter(data_25, day_of_week == "Sun")
ggplot(data_25, aes(x = carrier)) +
  geom_bar()
```

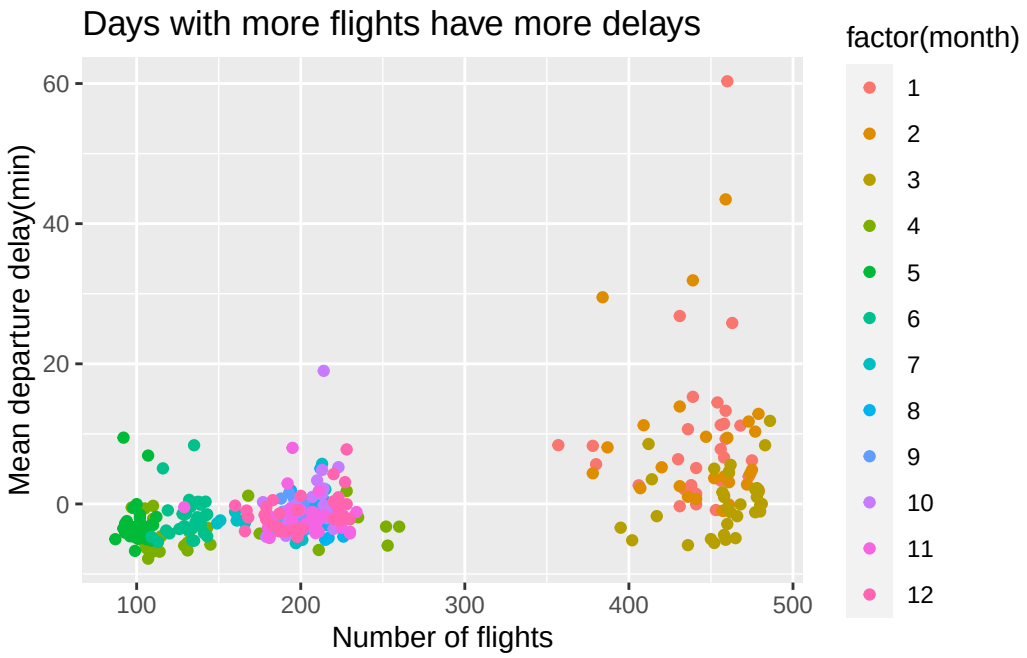


Taking airline WN on Sunday seems best for flying Santa Ana from San Francisco because in average, it has the least delay time.

26. The plot below shows a relationship between the number of flights going out of SFO and the average departure delay. It illustrates the hypothesis that more flights on a given day would lead to a more congested airport which would lead to greater delays on average (mean is used here specifically to capture the impact of the outliers - very long delays). Each point represents single day in 2020; there are 366 of them on the plot. Please form a single chain that will create this plot, starting with the raw data set.




```
library("dplyr")
options(dplyr.summarise.inform = FALSE)
library(dplyr, warn.conflicts = FALSE)
flights %>%
  filter(origin == "SFO") %>%
  select(dep_delay, month, day) %>%
  group_by(month, day) %>%
  summarize(avg_dep_delay = mean(dep_delay, na.rm = TRUE), num_flights = n()) %>%
  ggplot(aes(x = num_flights, y = avg_dep_delay, color = factor(month)))+
  geom_point() +
  ggtitle("Days with more flights have more delays") +
  xlab("Number of flights") +
  ylab("Mean departure delay(min)")
```


Days with more flights have more delays
Mean departure delay(min)
60
40
20
0
100
200
300
400
500
Number of flights
factor(month)
1
2
3
4
5
6
7
8
9
10
11
12