Data Transformation & Visualization with Tidyverse

Data Science Boot Camp Batch 10: Sprint 04 - 05

Author: _gu_npe.tnnx__ , Date: 2024-07-29

เนื้อหา

Library Package

Tidyverse

```
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────────────────────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.0     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────────────────────
── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become
errors

#tinytex::install_tinytex(force = TRUE)
```

nycflights23

```
library(nycflights23)

# Inspecting the details about nycflights23 dataset


#?flights
#?airlines
#?airports
#?planes
#?weather
```

NYCFlights23 คือ ชุดข้อมูลเกี่ยวกับสายการบินต่างๆที่บินมาที่สนามบินในนิวยอร์ค

- flights dataset เป็นข้อมูลเที่ยวบินทั้งหมดในเมืองนิวยอร์คของปี 2013
- airlines dataset เป็นข้อมูลชื่อสายการบินแบ่งตาม carrier code
- airports dataset เป็นข้อมูลที่เกี่ยวกับสนามบิน อาทิเช่น สนามบิน, ตำแหน่งที่ตั้งของสนามบิน
- plans dataset เป็นข้อมูลเกี่ยวกับเครื่องบิน อาทิเช่น ปีที่ผลิตเครื่องบิน, รุ่น (Model) ของเครื่องบิน
- weather dataset เป็นข้อมูลสภาพอากาศรายชั่วโมง

Load Some Data

```
head(flights)
```

```
## # A tibble: 6 × 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  2023     1     1        1           2038       203      328              3
## 2  2023     1     1       18           2300        78      228            135
## 3  2023     1     1       31           2344        47      500            426
## 4  2023     1     1       33           2140       173      238           2352
## 5  2023     1     1       36           2048       228      223           2252
## 6  2023     1     1      503            500         3      808            815
## # ℹ 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 <dttm>
```

```
print(paste("No of data points:", as.character(nrow(flights))))
```

```
## [1] "No of data points: 435352"
```

```
print(paste("No of columns:", as.character(ncol(flights))))
```

```
## [1] "No of columns: 19"
```

- จากการสำรวจข้อมูลใน Flight dataset ทำให้รู้ว่ามีข้อมูลที่เก็บในคอลัมน์และแถวทั้งหมด 19 (Column) และ 435352 (Row)
  และจากข้อมูลตัวอย่างในส่วนบนและท้ายของข้อมูลทั้งหมดพบว่าเป็นข้อมูลเกี่ยวกับการขึ้นบินและลงจอดของสายการบินต่างๆในแต่ละวัน
  รวมถึงยังมีข้อมูลเกี่ยวกับความล่าช้าของการขึ้นบินและลงจอดของสายการบินแต่ละสายการบินด้วย

==========

Data Transformation

Query Data Best

Distance

```
unique(flights$carrier)
```

```
##  [1] "UA" "DL" "B6" "AA" "NK" "WN" "AS" "YX" "9E" "HA" "G4" "MQ" "OO" "F9"
```

```
length(unique(flights$carrier))
```

```
## [1] 14
```

```
airlines # Dataframe that contains full name of flights$carrier
```

```
## # A tibble: 14 × 2
##    carrier name
##    <chr>   <chr>
##  1 9E      Endeavor Air Inc.
##  2 AA      American Airlines Inc.
##  3 AS      Alaska Airlines Inc.
##  4 B6      JetBlue Airways
##  5 DL      Delta Air Lines Inc.
##  6 F9      Frontier Airlines Inc.
##  7 G4      Allegiant Air
##  8 HA      Hawaiian Airlines Inc.
##  9 MQ      Envoy Air
## 10 NK      Spirit Air Lines
## 11 OO      SkyWest Airlines Inc.
## 12 UA      United Air Lines Inc.
## 13 WN      Southwest Airlines Co.
## 14 YX      Republic Airline
```

```r
## Total carrier_distance for each
carrier_distance <-
  flights %>%
  group_by(carrier) %>%
  summarize(sum_distance = sum(distance)) %>%
  select(carrier, sum_distance) %>%
  left_join(airlines, by = "carrier")

carrier_distance
```

```
## # A tibble: 14 × 3
##    carrier sum_distance name
##    <chr>          <dbl> <chr>
##  1 9E          26330226 Endeavor Air Inc.
##  2 AA          46745602 American Airlines Inc.
##  3 AS          19458447 Alaska Airlines Inc.
##  4 B6          75240252 JetBlue Airways
##  5 DL          78649564 Delta Air Lines Inc.
##  6 F9           1244713 Frontier Airlines Inc.
##  7 G4            486266 Allegiant Air
##  8 HA           1823778 Hawaiian Airlines Inc.
##  9 MQ            258879 Envoy Air
## 10 NK          16468094 Spirit Air Lines
## 11 OO           4042350 SkyWest Airlines Inc.
## 12 UA          99071109 United Air Lines Inc.
## 13 WN          12675607 Southwest Airlines Co.
## 14 YX          43062991 Republic Airline
```

Descriptive of Range Delay

```
des_delay <-
   flights %>%
     filter(dep_delay > 0) %>%
     group_by(carrier) %>%
     summarise(count_delay = n(),
               avg_delay = mean(dep_delay),
               std_delay = sd(dep_delay)) %>%
     left_join(airlines) %>%
     select(name, count_delay, avg_delay, std_delay) %>%
     arrange(desc(count_delay))

## Joining with `by = join_by(carrier)`
```

What is the most number of flights in year 2023?

```
count_flights <-
  flights %>%
  group_by(month, carrier) %>%
  summarise(count = n()) %>%
  select(month,
         carrier,
         count
         ) %>%
  arrange(month, carrier)

## `summarise()` has grouped output by 'month'. You can override using the
## `.groups` argument.

count_flights

## # A tibble: 165 × 3
## # Groups:   month [12]
```

```
##    month carrier count
##    <int> <chr>   <int>
##  1     1 9E       3985
##  2     1 AA       3574
##  3     1 AS        542
##  4     1 B6       5917
##  5     1 DL       4836
##  6     1 F9         92
##  7     1 G4         42
##  8     1 HA         31
##  9     1 MQ          9
## 10     1 NK       1176
## # ℹ 155 more rows
```

สังเกตว่า ข้อมูลมีเพียง 165 แถวเท่านั้น แต่จาก flights ทั้งหมดมี 14 carrier และ 12 ปี มันควรจะมี 168 แถว หมายความว่ามี 3 แถวที่ไม่มีข้อมูล

```
count_flights %>%
  group_by(month) %>%
  summarise(count = n())

## # A tibble: 12 × 2
##    month count
##    <int> <int>
##  1     1    14
##  2     2    14
##  3     3    14
##  4     4    14
##  5     5    14
##  6     6    13
##  7     7    13
##  8     8    13
```

```
##  9    9   14
## 10   10   14
## 11   11   14
## 12   12   14
```

What is the flights that not available?

```
month = c(1:12)
carrier = unique(count_flights$carrier)

# จำนวน flights ที่มีเที่ยวบินทั้งหมด
flights_list <- c(paste(count_flights$month,
                        count_flights$carrier,
                        sep = "-")
                  ) # len = 165

# จำนวน flights ที่เป็นไปได้เอา month x carrier ได้ 168 ข้อมูล
ideal_list <- c(paste(rep(month, each = 14),
                      rep(carrier, 12),
                      sep = '-')
                ) # len = 168

answer <- ideal_list[!ideal_list %in% flights_list]

print(capture.output(
  cat(
    "3 Flights which not active in this year is ...\n",
    answer
    )
  ))
```

```
## [1] "3 Flights which not active in this year is ..."
## [2] " 6-MQ 7-MQ 8-MQ"
```

Month have most flight

```
most_flight_month <-
 flights %>%
 group_by(month) %>%
 summarize(avg_total_flight= mean(n())) %>%
 arrange(desc(avg_total_flight))


most_flight_month
```

```
## # A tibble: 12 × 2
##    month avg_total_flight
##    <int>            <dbl>
##  1     3            39514
##  2     5            38710
##  3     4            37476
##  4     8            36765
##  5    10            36586
##  6     7            36211
##  7     1            36020
##  8     6            35921
##  9     9            35505
## 10     2            34761
## 11    11            34521
## 12    12            33362
```

What is the month that is the most distance?

```
most_flight_month[1,] # return month and distance
```

```
## # A tibble: 1 × 2
##   month avg_total_flight
##   <int>            <dbl>
## 1     3            39514
```

Most_Airport

```
most_flights_airport <-
  flights%>%
  group_by(dest)%>%
  summarize(count_flights=n(),.groups='drop') %>%
  left_join(airports,by= c("dest"="faa")) %>%
  arrange(desc(count_flights)) %>%
  slice_head(n=10) %>%
  select(dest,name, count_flights)


most_flights_airport
```

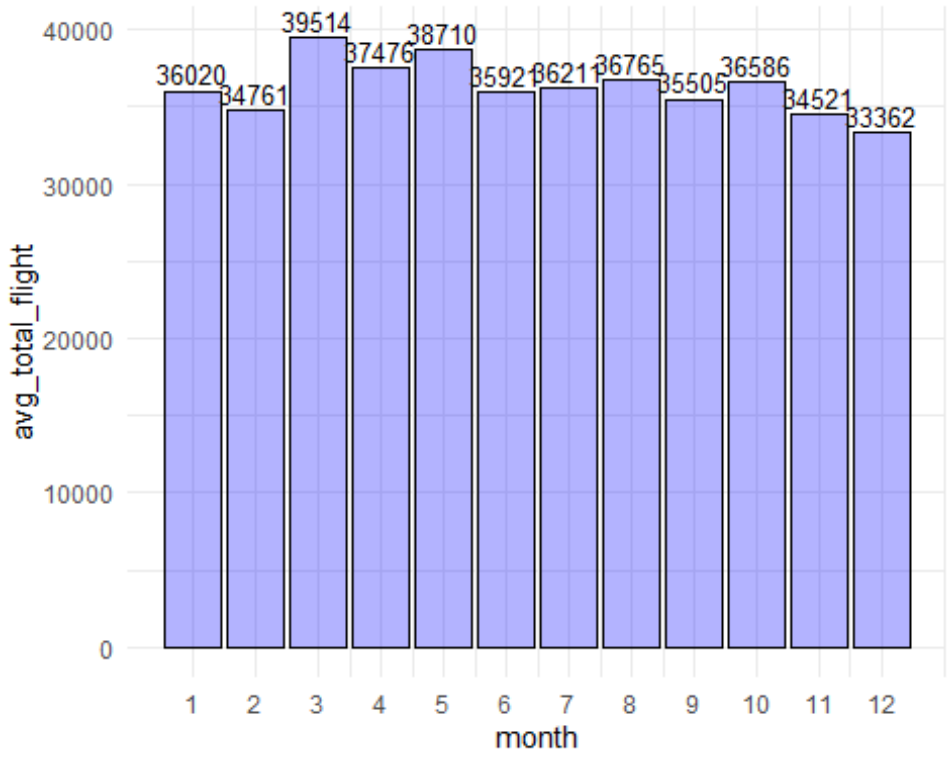
```
## # A tibble: 10 × 3
##    dest  name                                          count_flights
##    <chr> <chr>                                                 <int>
##  1 BOS   General Edward Lawrence Logan International Airport         19036
##  2 ORD   Chicago O'Hare International Airport                        18200
##  3 MCO   Orlando International Airport                               17756
##  4 ATL   Hartsfield Jackson Atlanta International Airport            17570
##  5 MIA   Miami International Airport                                 16076
##  6 LAX   Los Angeles International Airport                          15968
##  7 FLL   Fort Lauderdale Hollywood International Airport             14239
##  8 CLT   Charlotte Douglas International Airport                     12866
```

```
##  9 DFW   Dallas Fort Worth International Airport          11675
## 10 SFO   San Francisco International Airport             11651
```

Data Visualization

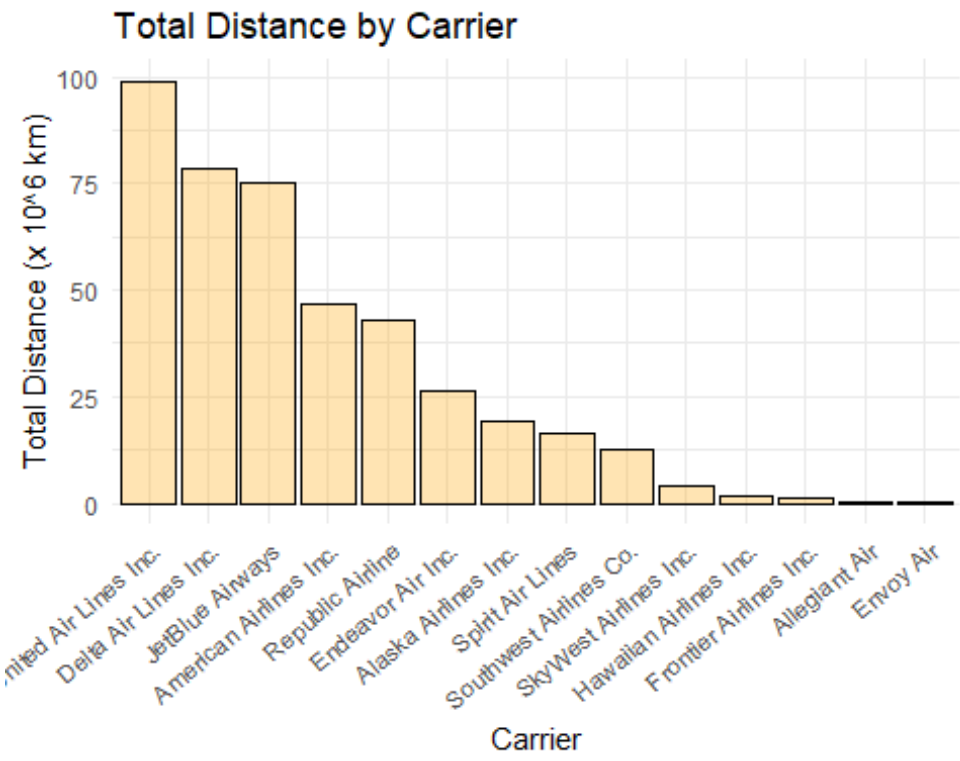
Plot Graph 1 : Histogram of month vs. avg_flights

```
ggplot(most_flight_month,
       aes(x=month,y=avg_total_flight)) +
 geom_col(fill="blue",
          alpha=0.3,
          color="black") +
 geom_text(aes(label=round(avg_total_flight,1))
           ,vjust=-0.3,size=3.5)+
 scale_x_continuous(
   breaks= round(unique(most_flight_month$month)))+
 theme_minimal()
```



Plot Graph 2 : Total Distance by Carrier

```
ggplot(carrier_distance,
       aes(x = reorder(name,-sum_distance),
           y = sum_distance/1000000)) +
  geom_col(fill = "orange",
           alpha = 0.3,
           color = "black") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 40, hjust = 1)) +
  labs(x = "Carrier",
       y = "Total Distance (x 10^6 km)",
       title = "Total Distance by Carrier")
```



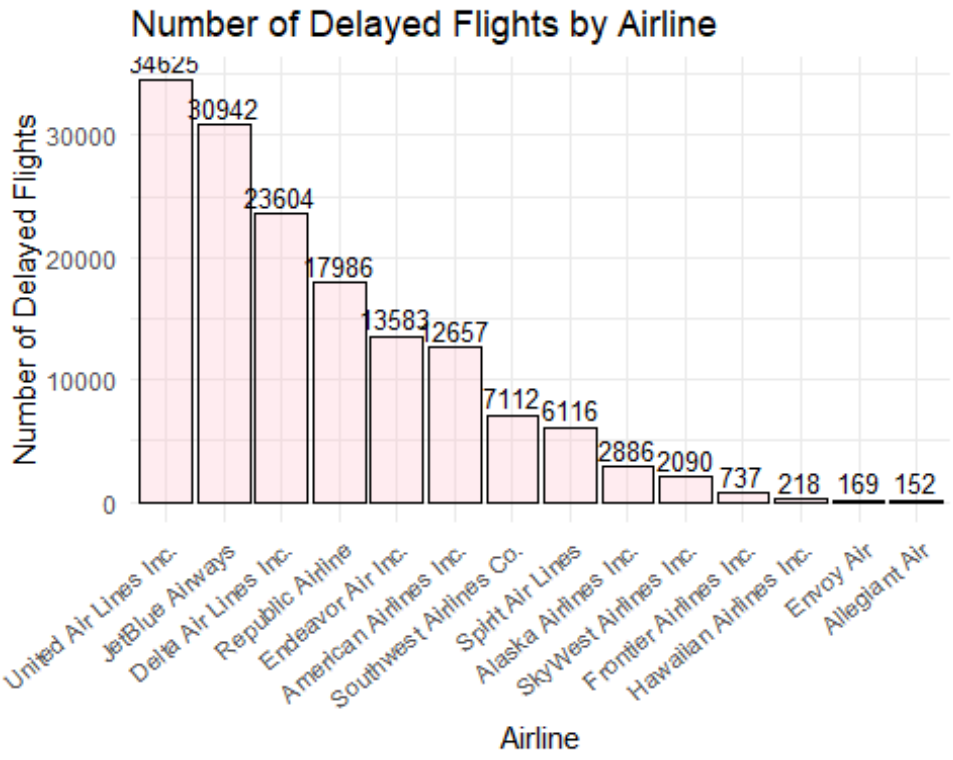
Plot Graph 3 : Number of Delayed Flights by Airline

```
delay_flights <-
  flights %>%
  filter(dep_delay > 0) %>%
  group_by(carrier) %>%
  summarize(delay_flight = n()) %>%
  select(carrier, delay_flight) %>%
  arrange(desc(delay_flight)) %>%
  left_join(airlines, by = "carrier")
```

PS. You can use des_delay dataframe, but I will create this dataframe ,because i want to practice to create new them.

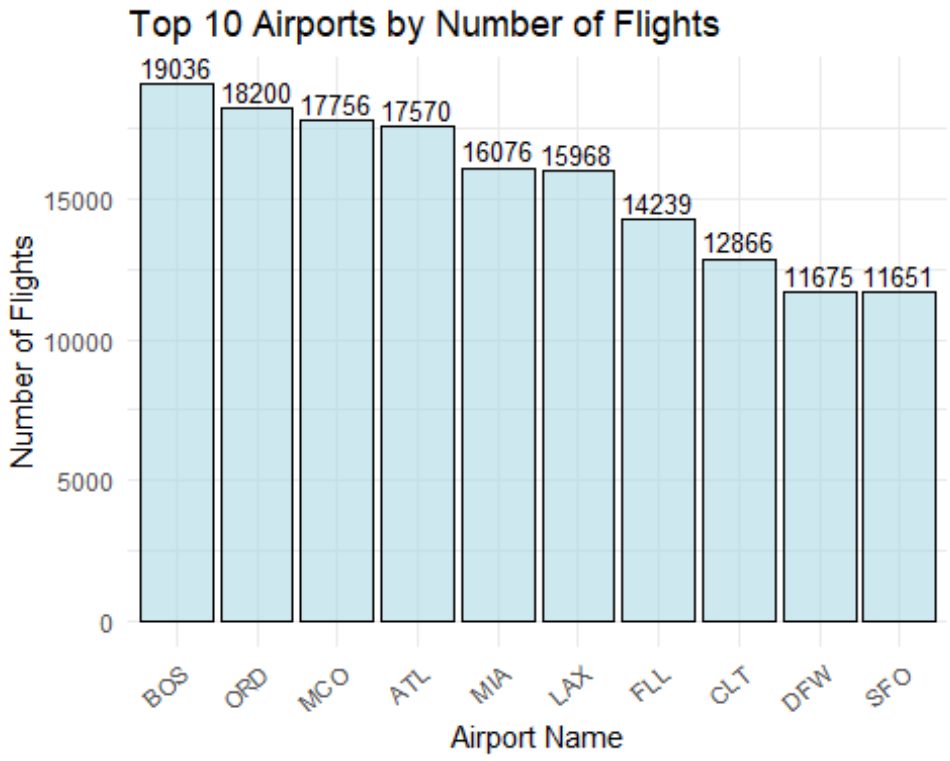
```
ggplot(delay_flights,
       aes(x = reorder(name,-delay_flight), y = delay_flight)) +
```

```
geom_col(fill = "pink",
         alpha = 0.3,
         color = "black") +
geom_text(aes(label = round(delay_flight, 1)),
          vjust = -0.3,
          size = 3.5) +
theme_minimal() +
theme(axis.text.x = element_text(angle = 40, hjust = 1)) +
labs(x = "Airline",
     y = "Number of Delayed Flights",
     title = "Number of Delayed Flights by Airline")
```



Plot Graph 4 : Top 10 Airports by Number of Flights

```
avg_delay_flights <-
  flights %>%
  filter(dep_delay > 0) %>%
  group_by(carrier) %>%
  summarize(delay_flight = mean(dep_delay,na.rm = TRUE)) %>%
  select(carrier, delay_flight) %>%
  arrange(desc(delay_flight)) %>%
  left_join(airlines, by = "carrier")

ggplot(most_flights_airport,
       aes(x = reorder(dest,-count_flights),
           y = count_flights)) +
geom_col(fill = "light blue",
         alpha = 0.6,
         color = "black") +
geom_text(aes(label = count_flights),
          vjust =-0.3,
          size = 3.5) +
theme_minimal() +
theme(axis.text.x = element_text(angle = 40, hjust = 1)) +
labs(x = "Airport Name",
     y = "Number of Flights",
     title = "Top 10 Airports by Number of Flights")
```

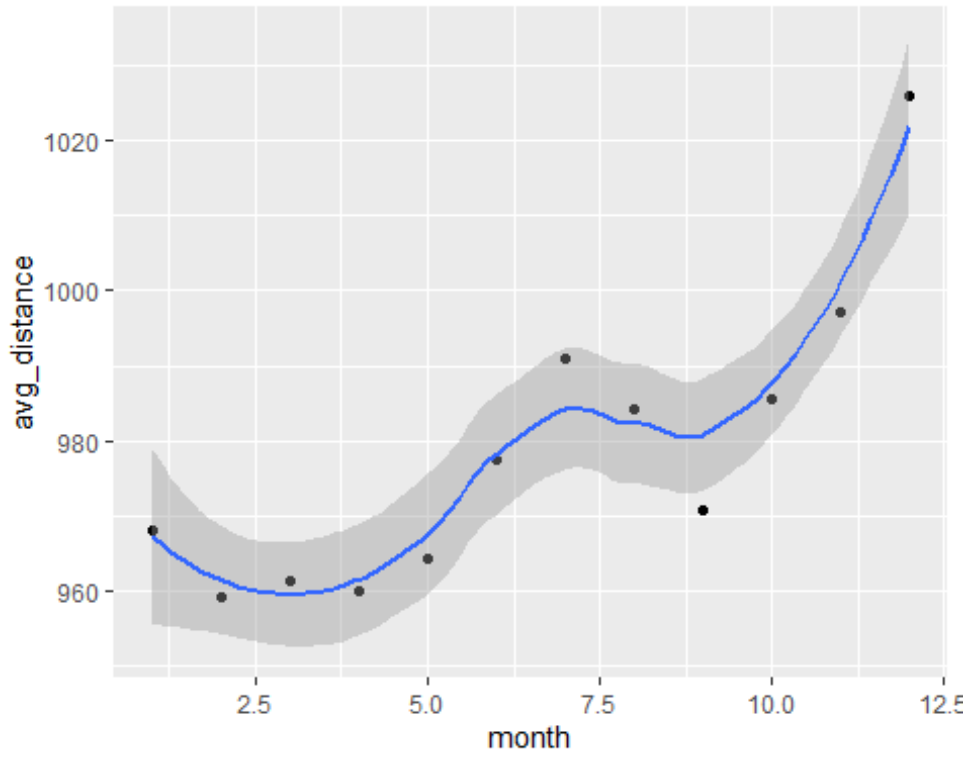

Top 10 Airports by Number of Flights
19036
18200
17756
17570
16076
15968
14239
12866
11675
11651
15000
10000
5000
0
Number of Flights
BOS
ORD
MCO
ATL
MIA
LAX
FLL
CLT
DFW
SFO
Airport Name

Plot Graph 5 : Scatter Plot Diagram of distance

```
set.seed(42)
n <- nrow(flights)
id <- sample(1:n, size = 0.5*n)
train_data <- flights[id,]
```

Scatter plot month vs. avg_distance

```
df1 <-
  train_data %>%
  group_by(month) %>%
  summarise(avg_distance = mean(distance))


ggplot(data = df1, aes(x = month, y = avg_distance)) +
  geom_point() +
  geom_smooth()

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

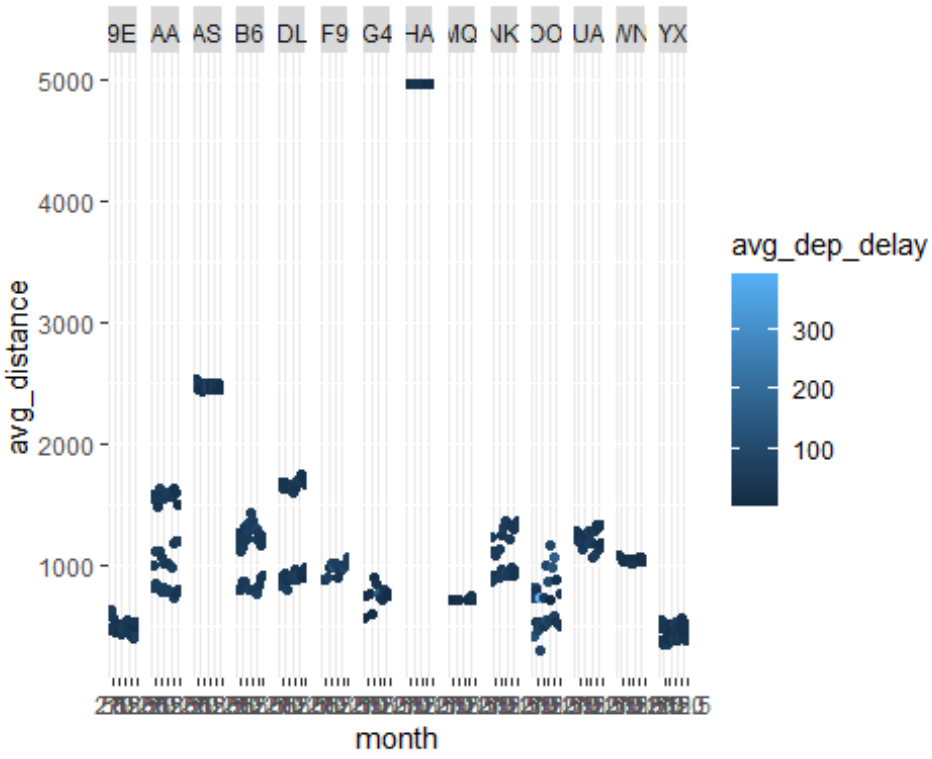


Scatter plot month vs. avg_distance vs. carrier

```
df2 <-
  train_data %>%
  filter(dep_delay > 0) %>%
  select(month, carrier, distance, dep_delay, origin) %>%
  group_by(month, carrier, origin) %>%
  summarise(avg_distance = mean(distance),
            avg_dep_delay = mean(dep_delay))

## `summarise()` has grouped output by 'month', 'carrier'. You can override using
## the `.groups` argument.

df2

## # A tibble: 337 × 5
## # Groups:   month, carrier [164]
##    month carrier origin avg_distance avg_dep_delay
```

```
##    <int> <chr>   <chr>        <dbl>        <dbl>
##  1     1 9E      EWR           647.         67.2
##  2     1 9E      JFK           491.         47.6
##  3     1 9E      LGA           478.         50.1
##  4     1 AA      EWR          1014.         50.4
##  5     1 AA      JFK          1583.         43.7
##  6     1 AA      LGA           824.         65.7
##  7     1 AS      EWR          2475.         44.4
##  8     1 AS      JFK          2542.         32.0
##  9     1 B6      EWR          1268.         38.7
## 10     1 B6      JFK          1159.         47.5
## # ℹ 327 more rows
```

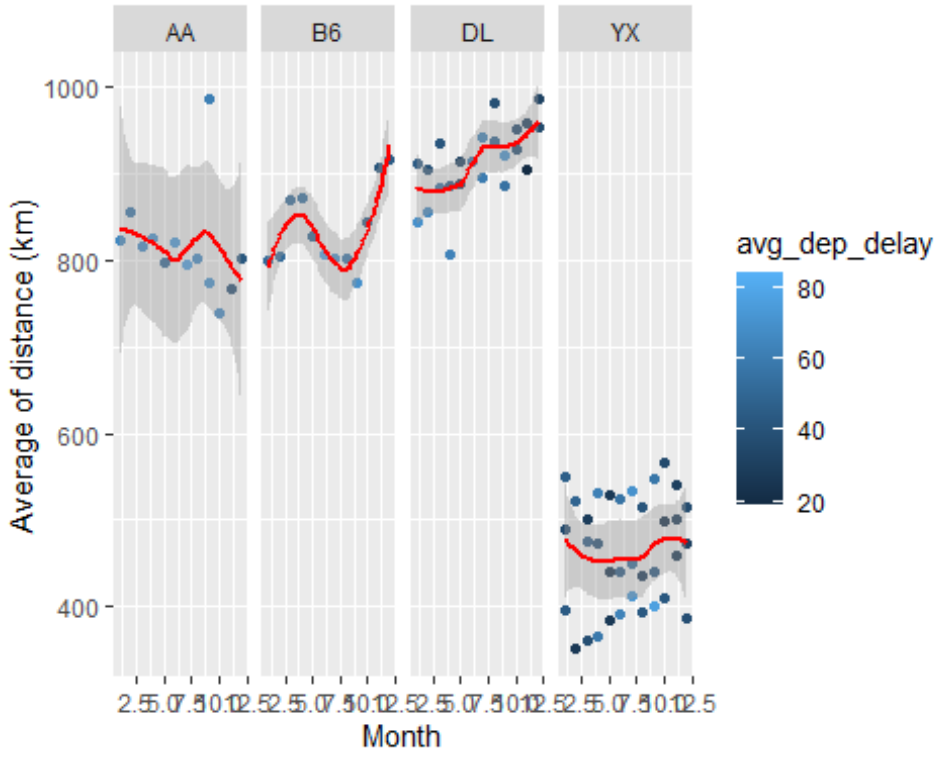
```
ggplot(data = df2, aes(x = month, y = avg_distance, color = avg_dep_delay)) +
  geom_point() +
  facet_grid(~ carrier)
```



สังเกตว่าช่วงระทาง 3K เป็นต้นไปค้นพบได้น้อยมาก ซึ่งมีเพียงแค่ Carrier HA เท่านั้นที่เจอ ฉะนั้นจึงเลือกพิจารณากรณีที่ต่ำกว่า 1K และบางเที่ยวบินพอ

```
ggplot(data = df2 %>% filter(avg_distance < 1000 &
                             carrier %in% c("AA", "B6", "DL", "YX")),
       aes(x = month, y = avg_distance, color = avg_dep_delay)) +
  geom_point() +
  geom_smooth(color = 'red') +
  facet_grid(~ carrier) +
  labs(x = "Month",
       y = "Average of distance (km)")
```

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



Plot Graph 6 : Bar chart of popular destination in 2023

Finding the popular destination

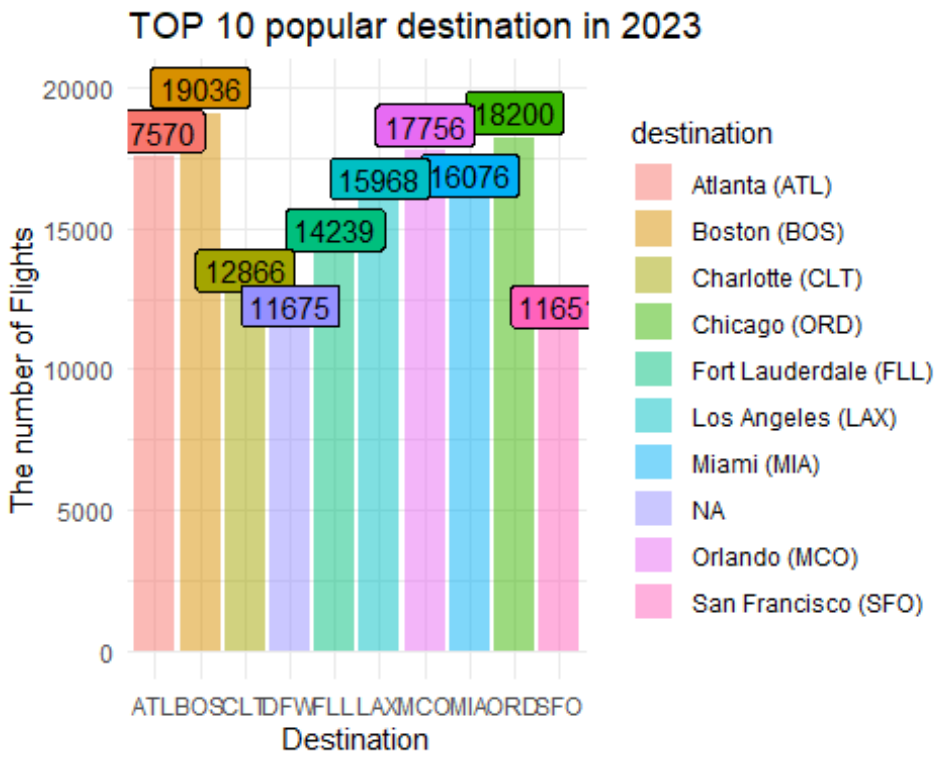
```
Pop_dest <- flights %>%
            count(dest) %>%
            arrange(-n) %>%
            head(10) %>%
            mutate(
   destination = ifelse(dest=="ATL","Atlanta (ATL)",
                 ifelse(dest=="BQN","Aguadilla (BQN)",
                 ifelse(dest=="FLL","Fort Lauderdale (FLL)",
                 ifelse(dest=="IAD","Washington (IAD)",
                 ifelse(dest=="IAH","Houston (IAH)",
                 ifelse(dest=="MCO","Orlando (MCO)",
                 ifelse(dest=="MIA","Miami (MIA)",
                 ifelse(dest=="ORD","Chicago (ORD)",
                 ifelse(dest=="PBI","West Palm Beach (PBI)",
                 ifelse(dest=="TPA","Tampa (TPA)",
                 ifelse(dest=="BOS","Boston (BOS)",
                 ifelse(dest=="CLT","Charlotte (CLT)",
                 ifelse(dest=="DCA","Washington (DCA)",
                 ifelse(dest=="LAX","Los Angeles (LAX)",
                 ifelse(dest=="SFO","San Francisco (SFO)","NA" ))))))))))))))
           )


Pop_dest

## # A tibble: 10 × 3
##    dest      n destination
##    <chr> <int> <chr>
```

```
##  1 BOS   19036 Boston (BOS)
##  2 ORD   18200 Chicago (ORD)
##  3 MCO   17756 Orlando (MCO)
##  4 ATL   17570 Atlanta (ATL)
##  5 MIA   16076 Miami (MIA)
##  6 LAX   15968 Los Angeles (LAX)
##  7 FLL   14239 Fort Lauderdale (FLL)
##  8 CLT   12866 Charlotte (CLT)
##  9 DFW   11675 NA
## 10 SFO   11651 San Francisco (SFO)
```

plotting bar chart of popular destination in 2023

```
ggplot(Pop_dest, aes(x = dest , y = n , fill= destination))+
  geom_col(size=5,alpha=0.5)+
  theme_minimal()+
  labs( title = "TOP 10 popular destination in 2023 ",
        x = "Destination",
        y = "The number of Flights" )+
  geom_label( aes(label= n),
              position = position_stack(vjust = 1.05),
              show.legend = FALSE)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



จากกราฟสรุปได้ว่าสถานที่ที่เป็นที่นิยม หรือมีเที่ยวบินลงจอดมากที่สุด 5 อันดับได้แก่ อันดับที่ 1 : Boston อันดับที่ 2 : Chicago อันดับที่ 3 : Orlando อันดับที่ 4 : Atlanta อันดับที่ 5 : Miami

---

Plot Graph 7 : bar chart of average arrival delay (minutes) by airline

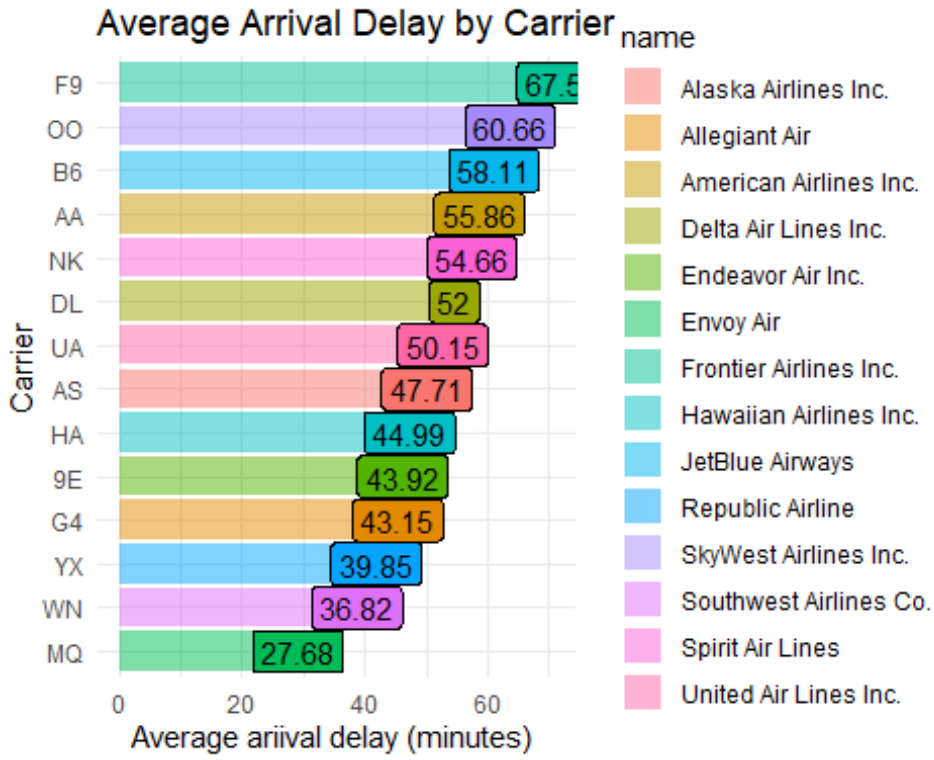
Finding Average of arrival delay for each airline

```
df3 <- flights %>%
  filter(arr_delay>0)%>%
  group_by(carrier) %>%
  summarise("Mean_Arr" = mean(arr_delay,na.rm = TRUE)) %>%
  left_join(airlines, by = "carrier")
```

Plotting bar chart of average arrival delay (minutes) by airline

```
ggplot(df3, aes(x = Mean_Arr, y =reorder(carrier,Mean_Arr)  , fill= name))+
  geom_col(size=5,alpha=0.5)+
  theme_minimal()+
```

```
labs( title = "Average Arrival Delay by Carrier",
      x = "Average ariival delay (minutes) ",
      y = "Carrier" )+
geom_label( aes(label= round(Mean_Arr,2)),
            position = position_stack(vjust = 1.05),
            show.legend = FALSE)
```



- จากกราฟด้านบนเป็นกราฟที่แสดงถึง ค่าเฉลี่ย delay time ของแต่ละสายการบิน โดยเฉลี่ยจากระยะเวลาของเที่ยวบินขาเข้าที่ล่าช้าและแยกตามสายการบินด้วย ซึ่งทำให้รู้ว่าสายการบิน F9 มีค่าเฉลี่ย delay time สูงที่สุด ซึ่งคิดเป็นเวลาเฉลี่ย 67.5 นาทีหรือ 1 ชั่วโมงกว่าเลยทีเดียว

Plot Graph 8 : Percent Delay by Carrier in 2023

Create Dataframe to contains delay and percent delay for each carrier

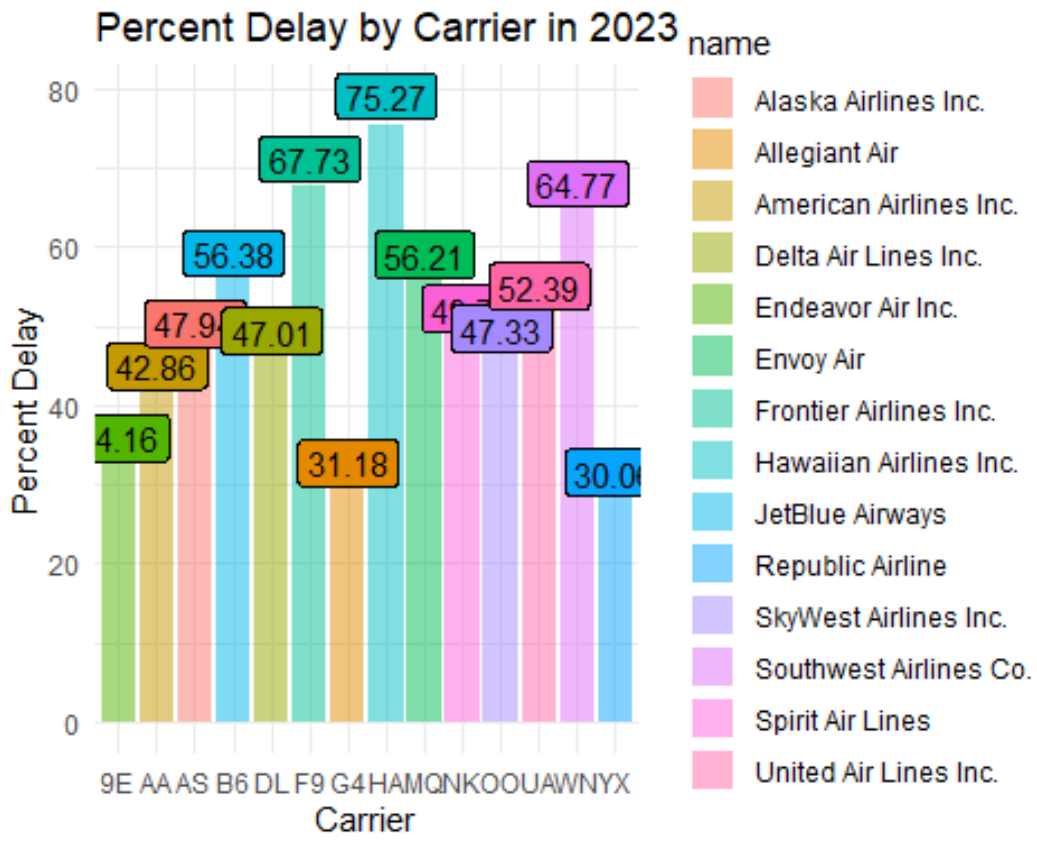
```
df4 <-  flights %>%
        filter(!is.na(dep_time)) %>%
        count(carrier) %>%
        rename( "count" = "n")


df4
```

```
## # A tibble: 14 × 2
##    carrier count
##    <chr>   <int>
##  1 9E      52380
##  2 AA      39940
##  3 AS       7774
##  4 B6      64622
##  5 DL      60616
##  6 F9       1218
##  7 G4        667
##  8 HA        364
##  9 MQ        354
## 10 NK      14827
## 11 OO       6214
## 12 UA      77810
## 13 WN      12105
## 14 YX      85723
```

```r
df5 <- flights %>%
       filter(dep_delay > 0 | arr_delay > 0) %>%
       count(carrier) %>%
       rename( "count_delay" = "n") %>%
       left_join(df4, by = "carrier") %>%
       left_join(airlines, by = "carrier") %>%
       mutate(
         percent_delay = round((count_delay/count)*100,2)
       )


df5
```

```
## # A tibble: 14 × 5
##    carrier count_delay count name                    percent_delay
##    <chr>         <int> <int> <chr>                           <dbl>
##  1 9E            17894 52380 Endeavor Air Inc.                34.2
##  2 AA            17117 39940 American Airlines Inc.           42.9
##  3 AS             3727  7774 Alaska Airlines Inc.             47.9
##  4 B6            36434 64622 JetBlue Airways                  56.4
##  5 DL            28494 60616 Delta Air Lines Inc.             47.0
##  6 F9              825  1218 Frontier Airlines Inc.           67.7
##  7 G4              208   667 Allegiant Air                    31.2
##  8 HA              274   364 Hawaiian Airlines Inc.           75.3
##  9 MQ              199   354 Envoy Air                        56.2
## 10 NK             7369 14827 Spirit Air Lines                 49.7
## 11 OO             2941  6214 SkyWest Airlines Inc.            47.3
## 12 UA            40767 77810 United Air Lines Inc.            52.4
## 13 WN             7840 12105 Southwest Airlines Co.           64.8
## 14 YX            25765 85723 Republic Airline                 30.1
```

```r
#plotting bar chart of percent delay by airline
ggplot(df5,
       aes(x = carrier, y =percent_delay, fill= name))+
  geom_col(size = 5,
           alpha = 0.5)+
  theme_minimal()+
  labs( title = "Percent Delay by Carrier in 2023",
        x = "Carrier",
        y = "Percent Delay" )+
  geom_label( aes(label=percent_delay),
              position = position_stack(vjust = 1.05),
              show.legend = FALSE)
```



- จากกราฟด้านบนเป็นกราฟที่แสดงถึง percent flights delay ในแต่ละสายการบิน โดยคิดจากจำนวนเที่ยวบินของแต่ละสายการบินที่เที่ยวบินล่าช้าเทียบกับจำนวนเที่ยวบินทั้งหมดของสาย

การบินนั้นๆ ซึ่งพบว่า สายการบินที่มี percent flights delay สูงสุด 3 อันดับ คือ HA, F9 และ WN ตามลำดับ

Thank you :D