IMT 573: Problem Set 2 - Working with Data

Xinyi Yang

Due: Tuesday, October 20, 2020 by 9am PT

Collaborators:

Instructions: Before beginning this assignment, please ensure you have access to R and RStudio.

1. Download the `problemset2.Rmd` file from Canvas. Open `problemset2.Rmd` in RStudio and supply your solutions to the assignment by editing `problemset2.Rmd`.
2. Replace the "Insert Your Name Here" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. Be sure to include well-documented (e.g. commented) code chucks, figures, and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text. Be sure that each visualization adds value to your written explanation; avoid redundancy – you do not need four different visualizations of the same pattern.
4. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment. In particular, note that Stack Overflow is licenses as Creative Commons (CC-BY-SA). This means you have to attribute any code you refer from SO.
5. Partial credit will be awarded for each question for which a serious attempt at finding an answer has been shown. But please **DO NOT** submit pages and pages of hard-to-read code and attempts that is impossible to grade. That is, avoid redundancy. Remember that one of the key goals of a data scientist is to produce coherent reports that others can easily follow. Students are *strongly* encouraged to attempt each question and to document their reasoning process even if they cannot find the correct answer. If you would like to include R code to show this process, but it does not run without errors you can do so with the `eval=FALSE` option as follows:

```
a + b # these object dont' exist
# if you run this on its own it with give an error
```

6. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit PDF`, rename the knitted PDF file to `ps2_ourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.
7. Collaboration is often fun and useful, but each student must turn in an individual write-up in their own words as well as code/work that is their own. Regardless of whether you work with others, what you turn in must be your own work; this includes code and interpretation of results. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.

Setup In this problem set you will need, at minimum, the following R packages.

```
# Load standard libraries
library(tidyverse)
library(nycflights13)
```

Problem 1: Describing the NYC Flights Data In this problem set we will continue to use the data on all flights that departed NYC (i.e. JFK, LGA or EWR) in 2013. Recall, you can find this data in the `nycflights13` R package. Load the data in R and ensure you know the variables in the data. Keep the documentation of the dataset (e.g. the help file) nearby.

In Problem Set 1 you started to explore this data. Now we will perform a more thorough description and summarization of the data, making use of our new data manipulation skills to answer a specific set of questions. When answering these questions be sure to include the code you used in computing empirical responses, this code should include code comments. Your response should also be accompanied by a written explanation, code alone is not a sufficient response.

(a) Describe and Summarize Answer the following qeustions in order to describe and summarize the `flights` data.

1. How many flights out of NYC are there in the data?
2. How many NYC airports are included in this data? Which airports are these?
3. Into how many airports did the airlines fly from NYC in 2013?
4. How many flights were there from NYC to Seattle (airport code `SEA`)?
5. Were the any flights from NYC to Spokane `(GAG)`?
6. What about missing destination codes? Are there any destinations that do not look like valid airport codes (i.e. three-letter-all-upper case)?

Hint:check the function `grepl` to do regular expression matching. You may use `"^[[:upper:]]{3}$"` for a regular expression that matches three upper case letters. See an example below:

```
grepl("^[[:upper:]]{3}$", c("12AB", "SEA", "ABCD", "ATL"))
```

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

```
#load data and make it a table
data(flights)
flights <- tibble::as_tibble(flights)

#check the origins of the data to make sure all flights are out of NYC
unique(flights$origin)
```

```
## [1] "EWR" "LGA" "JFK"
```

```
#print the number of flights out of NYC
paste("There are" , dim(flights)[1] , "flights out of NYC in the data.")
```

```
## [1] "There are 336776 flights out of NYC in the data."
```

```
a <- c(unique(flights$origin))
paste("There are",length(a),"NYC airports included in this data. They are",
      #print elements of a vector in one line uses collapse
      paste(a,collapse = ","))
```

```
## [1] "There are 3 NYC airports included in this data. They are EWR,LGA,JFK"
```

```
b <- unique(flights$dest)
paste("There are", length(b), "airports the airlines fly from NYC in 2013.")
```

```
## [1] "There are 105 airports the airlines fly from NYC in 2013."
```

```
paste("There are",nrow(flights %>% filter(dest=="SEA")),"flights from NYC to Seattle.")
```

```
## [1] "There are 3923 flights from NYC to Seattle."
```

```
nrow(flights %>% filter(dest=="GAG")) >0
```

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

RESPONSE: There is no flights from NYC to Spokane.

```
#use all function to check if all values in a vector are TRUE
all(grepl("^[[:upper:]]{3}$", b))
```

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

RESPONSE: All the destinations are valid airport codes.

(b) Reflect and Question Comment on the questions (and answers) so far. Were you able to answer all of these questions? Are all questions well defined? Is the data good enough to answer all these?

RESPONSE: I didn't feel any confused when answering the questions above. But we cannot make sure the destinations with three-letter-all-upper case are valid airport code. If we can validate them with a code sheet of these airports would be more convincing.

Problem 2: NYC Flight Delays Flights are often delayed. Let's look closer at this topic using the NYC Flight dataset. Answer the following questions about flight delays using the `dplyr` data manipulation verbs we talked about in class.

(1) Typical Delays What is the typical delay of flights in this data?
RESPONSE: Here we have departure delays and arrival delays in minutes in this data.

(2) Defining Flight Delays What definition of flight delay did you use to answer part (a)? Did you do any specific exploration and description of this variable prior to using it? If no, please do so now. Is there any missing data? Are there any implausible or invalid entries?

```
#look up description of the dataset
# ?flights

#check the data types of the delays
class(flights$dep_delay)
```

```
## [1] "numeric"
```

```
class(flights$arr_delay)
```

```
## [1] "numeric"
```

```
# get the summary of the delays
summary(flights$dep_delay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -43.00   -5.00   -2.00   12.64   11.00 1301.00    8255
```

```
summary(flights$arr_delay)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
##  -86.000  -17.000   -5.000    6.895   14.000 1272.000     9430
```

RESPONSE: There are 8255 missing data in the departure delay variable and 9430 missing data in the arrival delay variable. Except missing data, all entries are numeric. No implausible or invalid entries detected so far.

(3) Delays by Destination Now compute flight delay by destinations. Which ones are the worst three destinations from NYC if you don't like flight delays? Be sure to justify your delay variable choice.

```
#clean the data first to get calculate average data
flights_clean <- na.omit(flights)
worst_arrival <- flights_clean %>%
                      select(dest,dep_delay,arr_delay) %>%
                      group_by(dest) %>%
                      summarize(avg_arr_delay=mean(arr_delay)) %>%
                      # arrange(-avg_arr_delay) %>%
                      slice_max(avg_arr_delay,n=3)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
worst_departure <- flights_clean %>%
                      select(dest,dep_delay,arr_delay) %>%
                      group_by(dest) %>%
                      summarize(avg_dep_delay=mean(dep_delay)) %>%
                      # arrange(-avg_dep_delay) %>%
                      slice_max(avg_dep_delay,n=3)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
paste(worst_arrival$dest,collapse = ",")
```

```
## [1] "CAE,TUL,OKC"
```

```
paste(worst_departure$dest,collapse = ",")
```

```
## [1] "TUL,CAE,OKC"
```

```
knitr::kable(worst_arrival, caption = 'highest average arrival delays top 3 destinations')
```

Table 1: highest average arrival delays top 3 destinations

dest	avg_arr_delay
CAE	41.76415
TUL	33.65986
OKC	30.61905

```
knitr::kable(worst_departure, caption = 'highest average departure delays top 3 destinations')
```

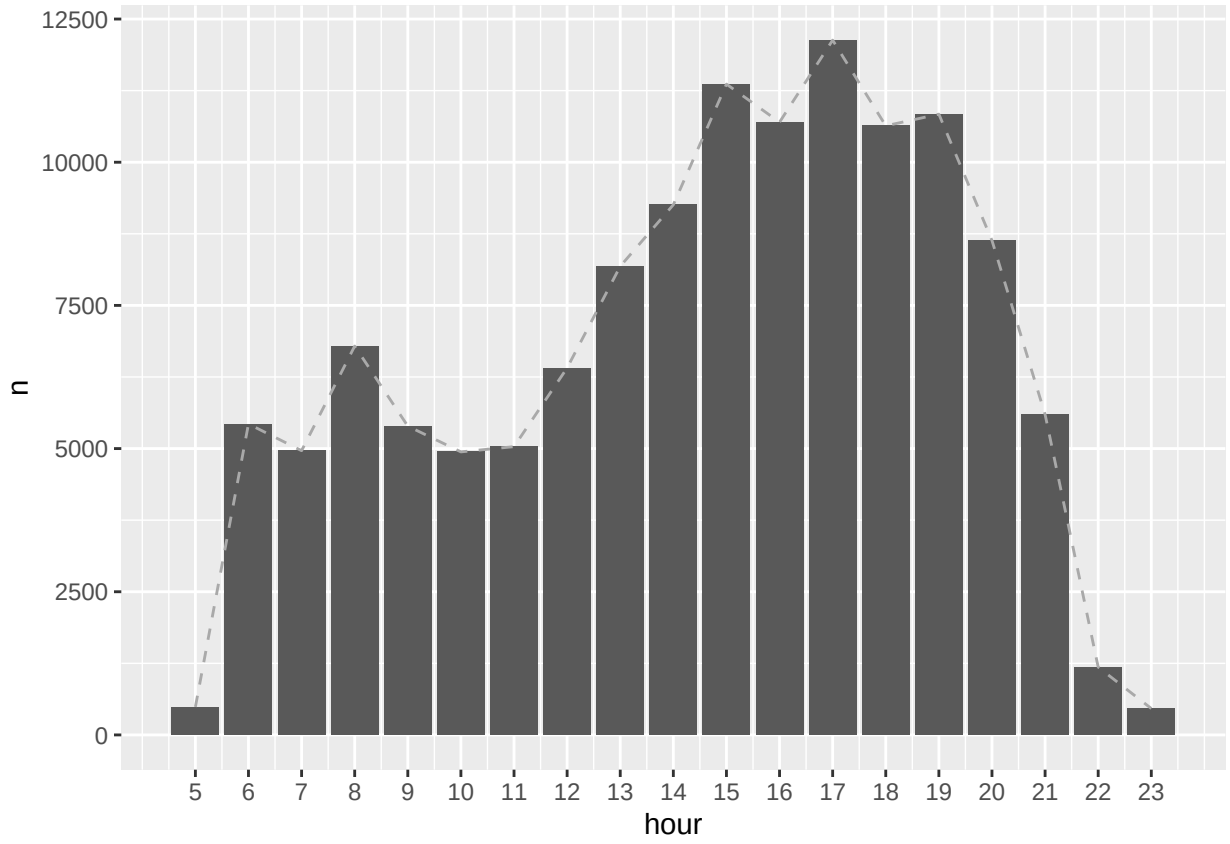
Table 2: highest average departure delays top 3 destinations

dest	avg_dep_delay
TUL	34.88776
CAE	33.81132
OKC	29.18095

The three destinations with the highest average departure delay and average arrival delay are all CAE, TUL and OKC.

(4) Delays by time of day We'd like to know how much do delays depend on the time of day. Are there more delays in the mornings? Late night when all the daily delays may accumulate? Create a visualization (graph or table) to illustrate your findings.

```r
f <- flights %>% filter(dep_delay>0) %>% count(hour)

ggplot(f, aes(x=hour,y=n))+
  geom_bar(stat = "identity")+
  geom_line(lty = 2, color = "darkgrey")+
  scale_x_continuous(breaks=c(unique(f$hour)))
```



First, I filtered the flights that delays at departure and group them by the scheduled hour in a day. Then, I made a bar plot to count the departure-delayed flights by hour to find the pattern. The plot shows there are more delays in the afternoon.

(5) Reflect and Challenge Your Results After completing the exploratory analyses from Problem 2, do you have any concerns about these questions and your findings? How well defined were the questions? If you feel a question is not defined well enough, re-formulate it in a more specific way so you can actually answer this question. And state clearly what is your more precise question.
Can you formulate any additional questions regarding flight delays?

RESPONSE: The definition of delay is not well-defined. For quesiton 2-2, I don't know what kind of entries are implausible or invalid entries the question refers to. Maybe the question could be, are there any blank entries or entries that are not numeric. For question 2-3, I calculated both delays to evaluate the worst destination. Maybe the question could be more specific on either departure delay or arrival delay. For question 2-4, I use departure delay to count delays by time of day. Because if there's a condition in a specific day, the most related delay is the departure delay. We can also count delays by different carriers to see which carriers are more possible to be on-time.

Problem 3: Let's Fly Across the Country!

(a) Describe and Summarize Answer the following questions in order to describe and summarize the `flights` data, focusing on flights from New York to Portland, OR (airport code `PDX`).

1. How many flights were there from NYC airports to Portland in 2013?
2. How many airlines fly from NYC to Portland?
3. Which are these airlines (find the 2-letter abbreviations)? How many times did each of these go to Portland?
4. How many unique airplanes fly from NYC to PDX?
   Hint: airplane tail number is a unique identifier of an airplane.
5. How many different airplanes arrived from each of the three NYC airports to Portland?
6. What percentage of flights to Portland were delayed at departure by more than 15 minutes?
7. Is one of the New York airports noticeably worse in terms of departure delays for flights to Portland, OR than others?

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.0.3
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following objects are masked from 'package:psych':
##
##     alpha, rescale
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

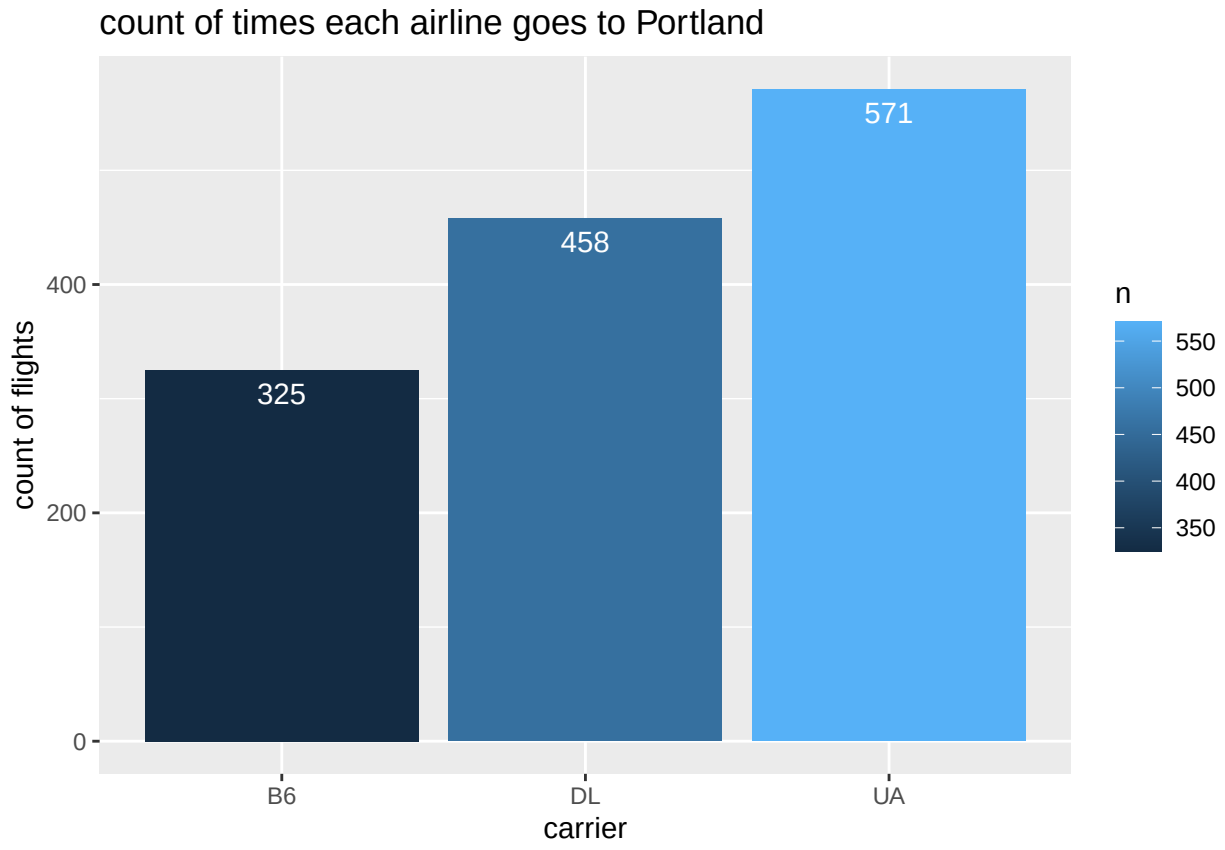
```
# make a shortcut of flights from NYC to PDX
port<-flights %>% filter(dest=='PDX')

# How many flights were there from NYC airports to Portland in 2013?
q3<-port %>% count(carrier)
paste('There are', nrow(q3), 'flights from NYC airports to Portland in 2013')
```

```
## [1] "There are 3 flights from NYC airports to Portland in 2013"
```

```
# How many airlines fly from NYC to Portland?
paste('There are',length(unique(q3$carrier)), 'different airlines fly from NYC to Portland. They are', p
```

```
## [1] "There are 3 different airlines fly from NYC to Portland. They are B6,DL,UA"
```

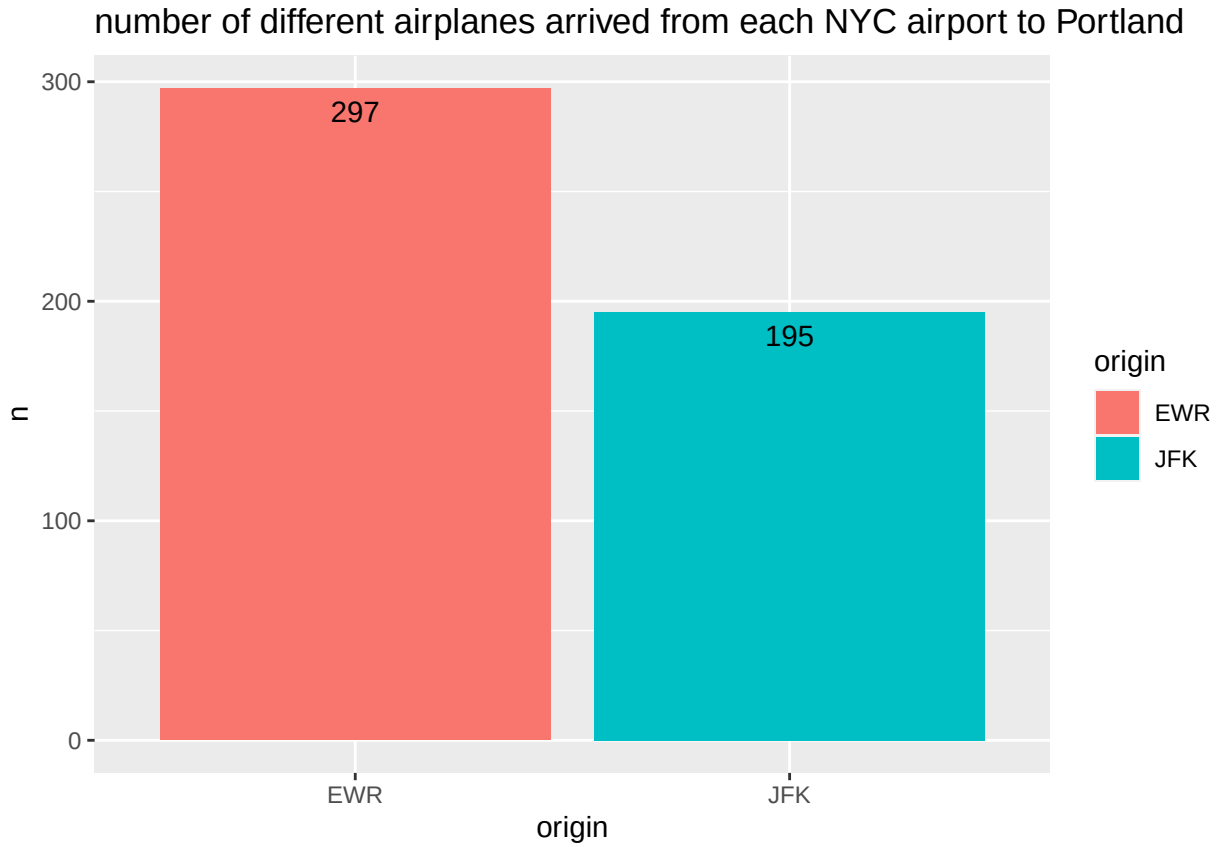
```
# Which are these airlines (find the 2-letter abbreviations)?  How many times did each of these go to P
ggplot(data=q3,aes(x=carrier,y=n))+
  geom_bar(stat = "identity",aes(fill=n))+
  geom_text(aes(label=n),vjust=1.6,color="white")+
  ggtitle("count of times each airline goes to Portland")+
  ylab("count of flights")
```




```
# How many unique airplanes fly from NYC to PDX?
paste('There are',length(unique(port$tailnum)),'unique airplanes fly from NYC to PDX.')
```

```
## [1] "There are 492 unique airplanes fly from NYC to PDX."
```

```
# How many different airplanes arrived from each of the three NYC airports to Portland?
ct_origin<-port %>% count(origin,tailnum) %>% count(origin)

ggplot(data=ct_origin,aes(x=origin,y=n)) +
  geom_bar(stat = "identity",aes(fill=origin))+
  geom_text(aes(label=n),vjust=1.6)+
  ggtitle("number of different airplanes arrived from each NYC airport to Portland")
```




```
# What percentage of flights to Portland were delayed at departure by more than 15 minutes?

# reference:https://stackoverflow.com/questions/7145826/how-to-format-a-number-as-percentage-in-r

per <- nrow(port %>% filter(dep_delay>15))/nrow(port)
paste('There are',percent(per),'flights to Portland were delayed at departure by more than 15 minutes.')
```

```
## [1] "There are 27% flights to Portland were delayed at departure by more than 15 minutes."
```

```
# Is one of the New York airports noticeably worse in terms of departure delays for flights to Portland

#statistical numbers of departure delays
summary(port$dep_delay)
```

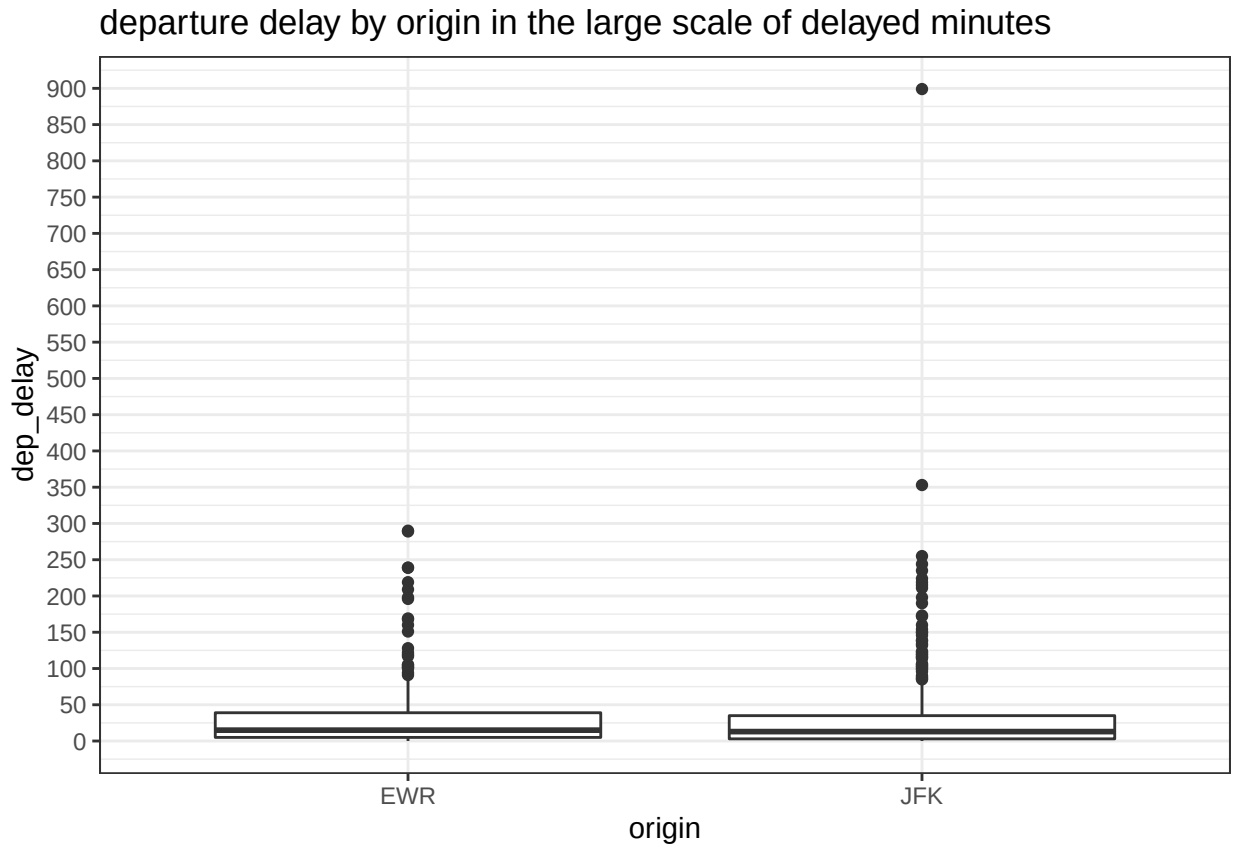
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -12.00   -3.00    1.00   16.26   18.00  899.00       6
```

```
#filter the dataset to focus on departure delays
port_delay<-port %>% filter(dep_delay>=0) %>% select(origin,dep_delay)
# different results statistical numbers of departure delays
describeBy(port_delay, port_delay$origin)
```

```
##
##  Descriptive statistics by group
## group: EWR
##           vars   n mean    sd median trimmed   mad min max range skew kurtosis
## origin*      1 340  1.0  0.00      1    1.00  0.00   1   1     0  NaN      NaN
## dep_delay    2 340 30.4 44.03     15   20.98 19.27   0 290   290 3.05    11.46
```

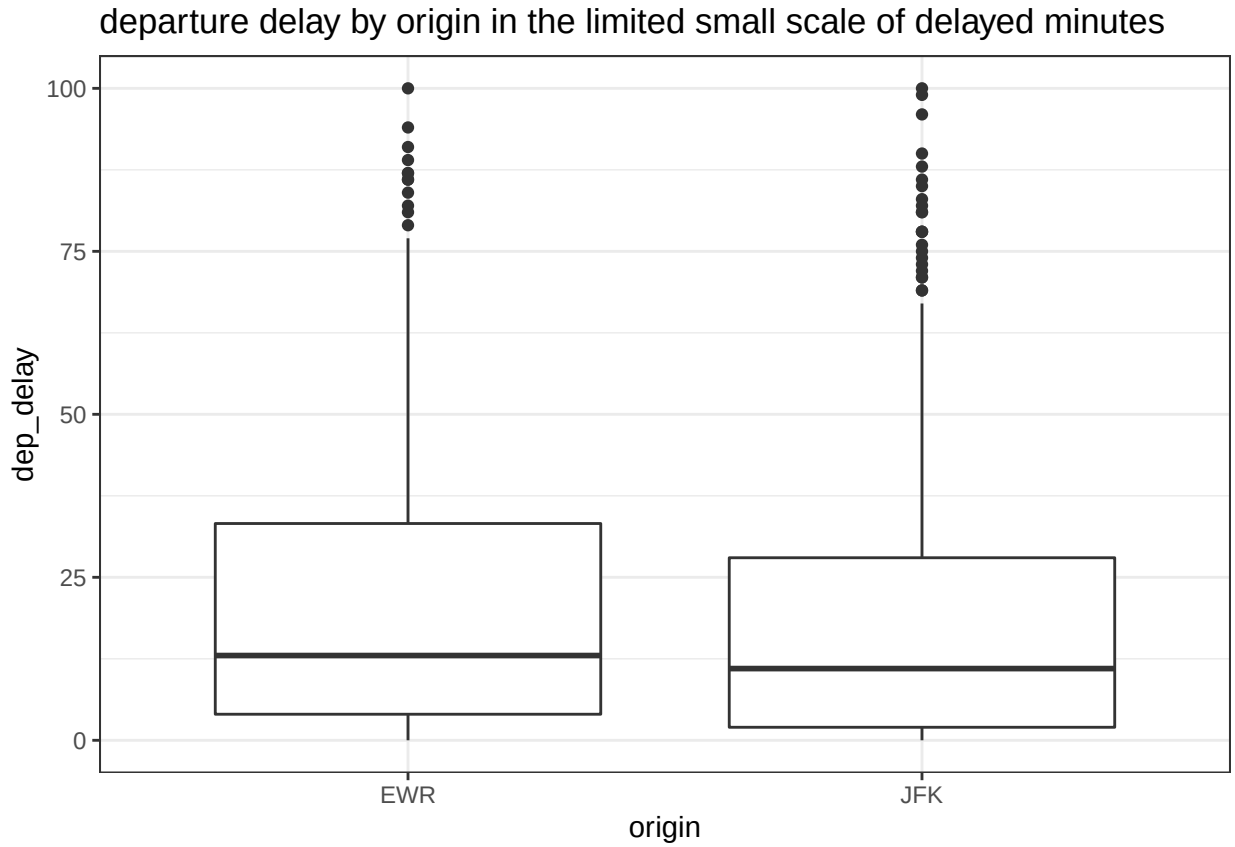
```
##             se
## origin*   0.00
## dep_delay 2.39
## ------------------------------------------------------------
## group: JFK
##           vars   n  mean    sd median trimmed   mad min max range skew kurtosis
## origin*      1 429  1.00  0.00      1     1.0  0.00   1   1     0  NaN      NaN
## dep_delay    2 429 32.04 62.42     13    19.7 17.79   0 899   899 7.39    87.74
##             se
## origin*   0.00
## dep_delay 3.01
```

```
ggplot(port_delay,aes(origin,dep_delay))+
  theme_bw()+
  geom_boxplot()+
  scale_y_continuous(breaks = seq(0,900,by=50))+
  ggtitle("departure delay by origin in the large scale of delayed minutes")
```




```
ggplot(port_delay,aes(origin,dep_delay))+
  geom_boxplot()+
  scale_y_continuous(limits = c(0,100))+
  theme_bw()+
  ggtitle("departure delay by origin in the limited small scale of delayed minutes")
```

```
## Warning: Removed 53 rows containing non-finite values (stat_boxplot).
```



RESPONSE: JFK has more departure delays especially long departure delays than EWR. While in terms of departure delays shorter than 50 minutes, EWR has longer average departure delays than JFK. In conclusion, for long delays JFK is worse while for short delays, EWR is worse.

(b) Reflect and Question Comment on the questions (and answers) in this analysis. Were you able to answer all of these questions? Are all questions well defined? Is the data good enough to answer all these?

RESPONSE: These questions are not all well-defined. The one asks the percentage. I calculated the percentage based on all the flights to Portland. But maybe the question refers to the percentage based on all the flights that delay at departure. And the question that asks if one of the NYC airports is noticeably worse is not well-defined. What the criteria to judge the airport by departure delays. Is it average delay or how many times when delay occurs or delays longer. Also, I don't think the data is good enough. The longest departure delay in the above analysis is 899 minutes. I doubt the reliability of the data.

Extra Credit

Seasonal Delays Let's get back to the question of flight delays. Flight delays may be partly related to weather, as you might have experienced for yourself. We do not have weather information here but let's analyze how it is related to season. Which seasons have the worst flights delays? Why might this be the case? In your communication of your analysis use one graphical visualization and one tabular respresentation of your findings.

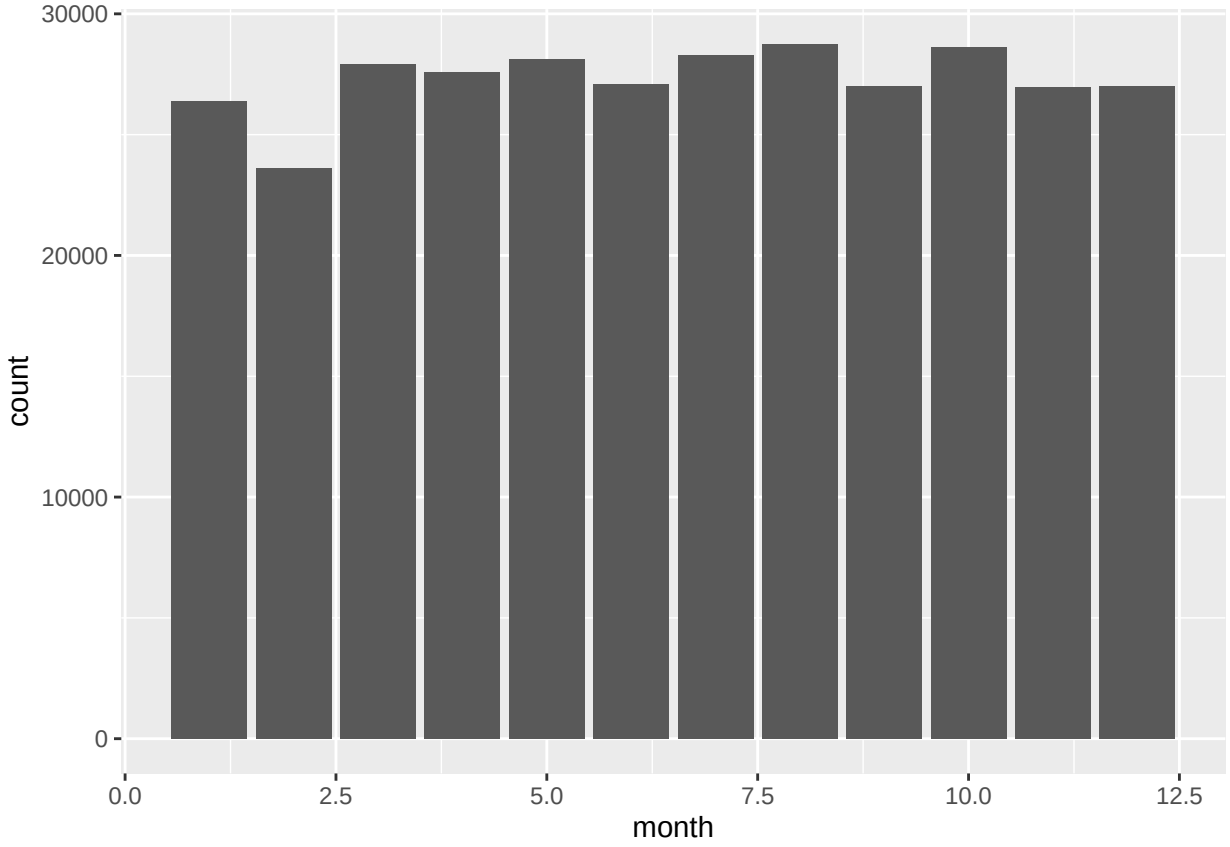
```
flights_clean <- na.omit(flights)
# check all flights of each month to make sure we are observing the differences on a similar base.
m <- flights_clean %>% count(month)
```

```
knitr::kable(m, caption = 'number of flights by month')
```

Table 3: number of flights by month

month	n
1	26398
2	23611
3	27902
4	27564
5	28128
6	27075
7	28293
8	28756
9	27010
10	28618
11	26971
12	27020

```
# table is hard to see the differences of these numbers
ggplot(flights_clean) +
  geom_bar(mapping = aes(x=month,fill=month))
```

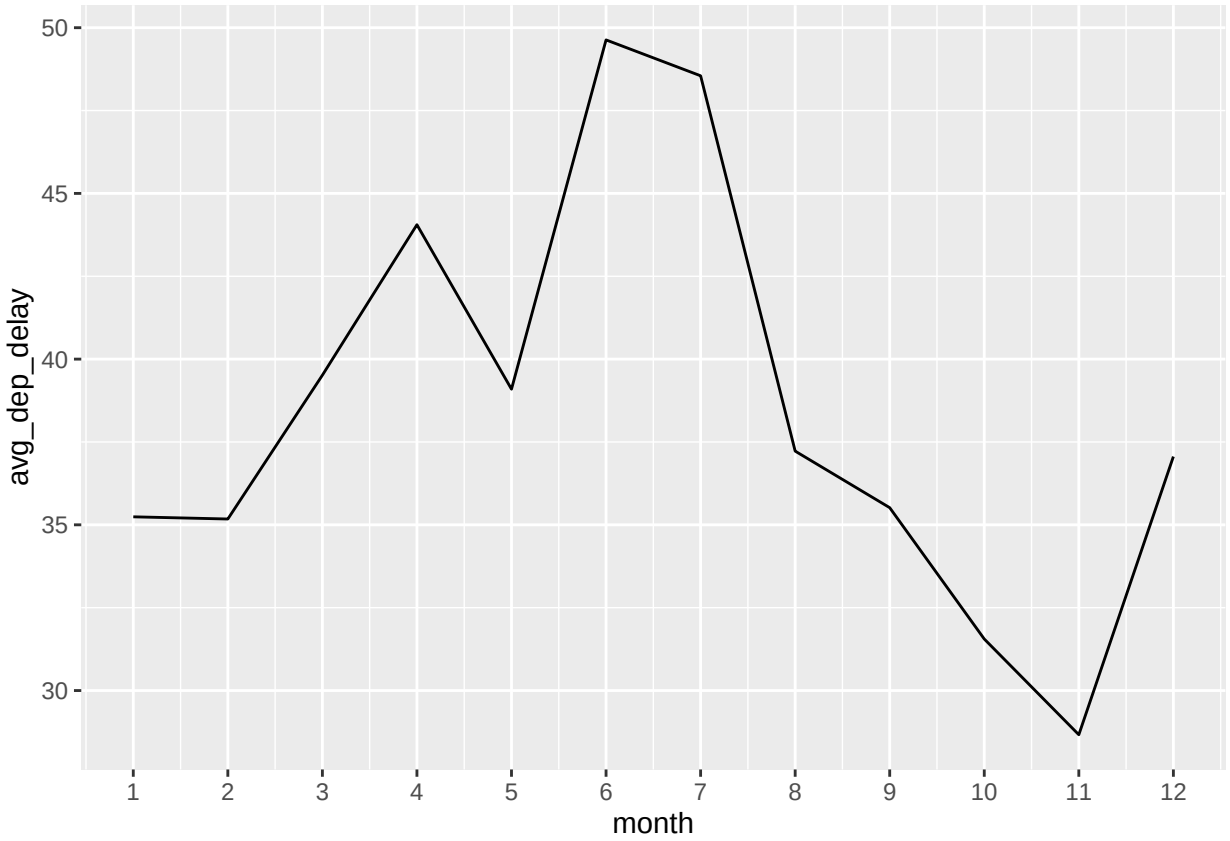



```
# as we are observing delays, so I eliminates the condition when the flights take off early.
p<-flights_clean %>%
  filter(dep_delay>0) %>%
```

```
  group_by(month) %>%
  summarise(avg_dep_delay=mean(dep_delay))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
# observe the seasonal delay by average departure delay by month
ggplot(data=p)+
  geom_line(mapping = aes(x=month,y=avg_dep_delay))+
  scale_x_continuous(breaks = seq(1,12,by=1))
```



RESPONSE: According to [wikipedia](https://en.wikipedia.org/wiki/Season), winter starts at December, spring starts at March, summer starts at June, and autum starts at September. Summer has the worst flights delays. Maybe there are more storms and lightnings in summer.