IMT 573: Problem Set 3

Working with Data: Part I

Ali Qazi

Due: July 11, 2021

Collaborators: Akeel Qazi, Anthony Mercado

Instructions: Before beginning this assignment, please ensure you have access to R and RStudio; this can be on your own personal computer or on the IMT 573 R Studio Cloud.

1. Download the `03_ps_workingdata.Rmd` file from Canvas or save a copy to your local directory on RStudio Cloud. Supply your solutions to the assignment by editing `03_ps_workingdata.Rmd`.
2. Replace the "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 no need four different visualizations of the same pattern.
4. Collaboration on problem sets is fun and useful, and we encourage it, 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.
5. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment.
6. Remember partial credit will be awarded for each question for which a serious attempt at finding an answer has been shown. 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:
7. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit`, download and rename the knitted PDF file to `ps3_YourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.

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

```
# Load standard libraries
library(tidyverse)
library(nycflights13)
library(ggplot2)
library(dplyr)
library(knitr) # this will keep code on the page
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
```

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.

```
# Load the nycflights13 library which includes data on all
# lights departing NYC
data(flights)
# Note the data itself is called flights, we will make it
# into a local df for readability
flights <- tbl_df(flights)
```

```
## Warning: `tbl_df()` was deprecated in dplyr 1.0.0.
## Please use `tibble::as_tibble()` instead.
```

```
# Look at the help file for information about the data fl
# ights
flights
```

```
## # A tibble: 336,776 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     1     1      517            515         2      830            819
##  2  2013     1     1      533            529         4      850            830
##  3  2013     1     1      542            540         2      923            850
##  4  2013     1     1      544            545        -1     1004           1022
##  5  2013     1     1      554            600        -6      812            837
##  6  2013     1     1      554            558        -4      740            728
##  7  2013     1     1      555            600        -5      913            854
##  8  2013     1     1      557            600        -3      709            723
##  9  2013     1     1      557            600        -3      838            846
## 10  2013     1     1      558            600        -2      753            745
## # ... with 336,766 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 <dttm>
```

```
# summary(flights)
```

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 questions in order to describe and summarize the `flights` data.

#begin{enumerate}

#item How many flights out of NYC are there in the data?

```
summary(flights$dest, count = n())
```

```
##    Length     Class      Mode 
##    336776 character character
```

There are 336776 flights in the data and all have departed from different airports in New York.

#item How many NYC airports are included in this data? Which airports are these?

```
flight_origin <- table(flights$origin)
# Find unique flight origin
flight_origin
```

```
## 
##    EWR    JFK    LGA 
## 120835 111279 104662
```

NYC's 3 airports are JFK, LGA, and EWR.

#item Into how many airports did the airlines fly from NYC in 2013?

```
airports = filter(flights, year == 2013) %>%
    group_by(dest)
n_distinct(airports$dest, na.rm = FALSE)
```

```
## [1] 105
```

In 2013 fligth from 105 airports.

item How many flights were there from NYC to Seattle (airport code SEA)?

```
fltnum_SEA = filter(flights, flights$dest == "SEA")
summarise(fltnum_SEA, count = n())
```

```
## # A tibble: 1 x 1
##   count
##   <int>
## 1  3923
```

There were 3923 flights from NYC to Seattle.

#item Were the any flights from NYC to Spokane #texttt{(GAG)}?

```
flightnums_GAG = filter(flights, flights$dest == "GAG")
summarise(flightnums_GAG, count = n())
```

```
## # A tibble: 1 x 1
##   count
##   <int>
## 1     0
```

0 flights from NYC to Spokane.

#item What about missing destination codes? Are there any destinations that do not look like valid airport codes (i.e. three-letter-all-upper case)?

```
unique(flights$dest)  # inspect any missing dest code
```

```
##   [1] "IAH" "MIA" "BQN" "ATL" "ORD" "FLL" "IAD" "MCO" "PBI" "TPA" "LAX" "SFO"
##  [13] "DFW" "BOS" "LAS" "MSP" "DTW" "RSW" "SJU" "PHX" "BWI" "CLT" "BUF" "DEN"
##  [25] "SNA" "MSY" "SLC" "XNA" "MKE" "SEA" "ROC" "SYR" "SRQ" "RDU" "CMH" "JAX"
##  [37] "CHS" "MEM" "PIT" "SAN" "DCA" "CLE" "STL" "MYR" "JAC" "MDW" "HNL" "BNA"
##  [49] "AUS" "BTV" "PHL" "STT" "EGE" "AVL" "PWM" "IND" "SAV" "CAK" "HOU" "LGB"
##  [61] "DAY" "ALB" "BDL" "MHT" "MSN" "GSO" "CVG" "BUR" "RIC" "GSP" "GRR" "MCI"
##  [73] "ORF" "SAT" "SDF" "PDX" "SJC" "OMA" "CRW" "OAK" "SMF" "TUL" "TYS" "OKC"
##  [85] "PVD" "DSM" "PSE" "BHM" "CAE" "HDN" "BZN" "MTJ" "EYW" "PSP" "ACK" "BGR"
##  [97] "ABQ" "ILM" "MVY" "SBN" "LEX" "CHO" "TVC" "ANC" "LGA"
```

There are 0 destinations that do not look like valid airport codes.

#end{enumerate}

(b) Reflect and Question Comment 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?

The questions are all well_defined. I was able to answer all of the questions. In addition, the data was good enough to answer all questions.

Problem 2: NYC Flight Delays

Flights are often delayed. Let's look at 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.

(a) Typical Delays What is the typical delay of flights in this data?

```
delay = group_by(flights, flights$dep_delay)
delay = summarise(delay, count = n(), na.rm = TRUE)
arrange(delay, desc(count))  # returns in descending order
```

```
## # A tibble: 528 x 3
##    `flights$dep_delay` count na.rm
##                  <dbl> <int> <lgl>
##  1                  -5 24821 TRUE
##  2                  -4 24619 TRUE
##  3                  -3 24218 TRUE
##  4                  -2 21516 TRUE
##  5                  -6 20701 TRUE
##  6                  -1 18813 TRUE
##  7                  -7 16752 TRUE
##  8                   0 16514 TRUE
##  9                  -8 11791 TRUE
## 10                  NA  8255 TRUE
## # ... with 518 more rows
```

```
arrdelay = group_by(flights, flights$arr_delay)
arrdelay = summarise(arrdelay, count = n(), na.rm = TRUE)
arrange(arrdelay, desc(count))  # returns in descending order
```

```
## # A tibble: 578 x 3
##    `flights$arr_delay` count na.rm
##                  <dbl> <int> <lgl>
##  1                  NA  9430 TRUE
##  2                 -13  7177 TRUE
##  3                 -10  7088 TRUE
##  4                 -12  7046 TRUE
##  5                 -14  6975 TRUE
##  6                 -11  6863 TRUE
##  7                  -9  6815 TRUE
##  8                 -15  6796 TRUE
##  9                  -7  6677 TRUE
## 10                 -17  6668 TRUE
## # ... with 568 more rows
```

The typical delay of arriving flights is -13. The typical delay of departing flights is -5 minutes.

(b) 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?

```
is.numeric(flights$arr_delay)  # checking if all delay is numeric
```

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

```
sum(is.na(flights$arr_delay))
```

```
## [1] 9430
```

For part A I used both arrival and departure. The first row or arrival delays is missing data. That null value has the 9430 counts, meaning the usual arrival delay time is invalid.

(b) 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.

```
flights %>%
    filter(dep_delay > 0) %>%
    group_by(dest) %>%
    summarize(delay = mean(dep_delay)) %>%
    arrange(desc(delay))
```

```
## # A tibble: 103 x 2
##    dest  delay
##    <chr> <dbl>
##  1 TVC    69.2
##  2 TYS    65.7
##  3 TUL    64.7
##  4 BHM    63.9
##  5 DSM    63.4
##  6 CHO    60.9
##  7 RIC    57.0
##  8 CRW    55.9
##  9 CVG    55.7
## 10 ILM    55.4
## # ... with 93 more rows
```

```
head(3)
```

```
## [1] 3
```

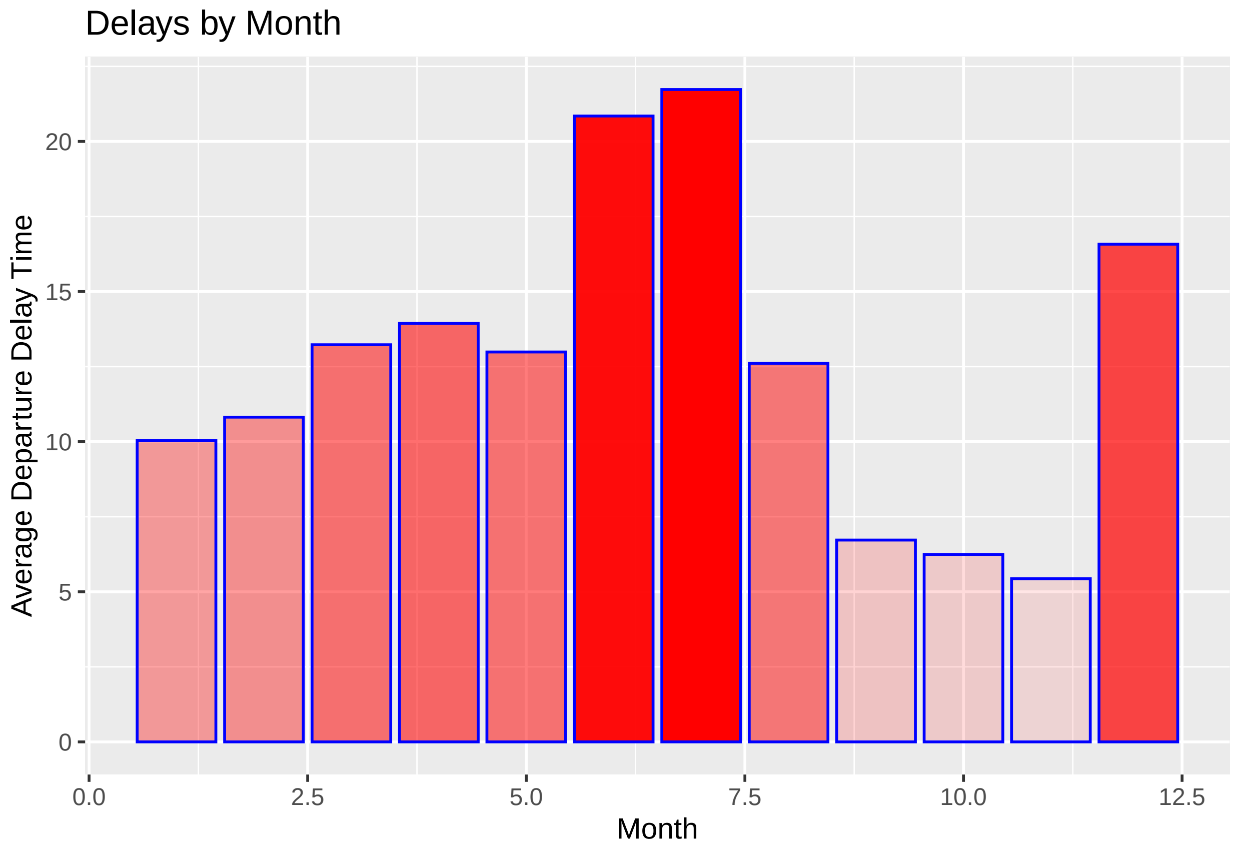
TVC, TYS, and TUL are the three worst from NYC. I took the average of departure delays as the variable of choice as passengers do not like waiting in the airport for planes to depart.

(b) Seasonal 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 representation of your findings.

```
season <- group_by(flights, month)
season <- summarise(season, count = n(), delay = mean(dep_delay,
    na.rm = TRUE))
arrange(season, desc(delay))  # sorted in descending order
```

```
## # A tibble: 12 x 3
##    month count delay
##    <int> <int> <dbl>
##  1     7 29425 21.7
##  2     6 28243 20.8
##  3    12 28135 16.6
##  4     4 28330 13.9
##  5     3 28834 13.2
##  6     5 28796 13.0
##  7     8 29327 12.6
##  8     2 24951 10.8
##  9     1 27004 10.0
## 10     9 27574  6.72
## 11    10 28889  6.24
## 12    11 27268  5.44
```

```
ggplot(data = season, mapping = aes(x = month, y = delay)) +
    # High values in delay will appear more red than low
    # values in delay
geom_bar(aes(alpha = delay), colour = "blue", fill = "red", stat = "identity") +
    xlab("Month") + ylab("Average Departure Delay Time") + ggtitle("Delays by Month") +
    theme(legend.position = "none")  # No need for legend
```



From the visual June and July, are the months with the most common delays. This can be due to cancelations, heat and understaff. From the visualization the months that experience the most delays are a darker red.

(3) Challenge Your Results After completing the exploratory analyses from Problem 2, do you have any concerns about your findings? How well defined was your original question? Do you still believe this question can be answered using this dataset? Comment on any ethical and/or privacy concerns you have with your analysis.

I have no concerns with my findings. The original questions were defined well, however, it can include whether the delays should be arrival or departure. With the dataset all questions can still be answered. There are no ethical and/or privacy concerns with my analysis.

Problem 3: Let's Fly to 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`).

begin{enumerate}

item How many flights were there from NYC airports to Portland in 2013?

```
flights %>%
    filter(dest == "PDX") %>%
    count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1  1354
```

There are 1354 flights from NYC to PDX.

#item How many airlines fly from NYC to Portland?

```
flightnumber_pdx <- filter(flights, flights$dest == "PDX")
summarise(flightnumber_pdx, Flight_PDX = n_distinct(carrier))  # Counts number of unique values
```

```
## # A tibble: 1 x 1
##   Flight_PDX
##        <int>
## 1          3
```

There are three airlines that fly from NYC to Portland.

#item Which are these airlines (find the 2-letter abbreviations)? How many times did each of these go to Portland?

```
flights %>%
    filter(dest == "PDX") %>%
    group_by(carrier) %>%
    count()
```

```
## # A tibble: 3 x 2
## # Groups:   carrier [3]
##   carrier     n
##   <chr>   <int>
## 1 B6        325
## 2 DL        458
## 3 UA        571
```

The airlines that went to PDX are JetBlue Airways, Delta Air Lines, and United Airlines.

#item How many unique airplanes fly from NYC to PDX? #textcolor{blue}{Hint: airplane tail number is a unique identifier of an airplane.}

```
flights %>%
    filter(dest == "PDX") %>%
    summarize(n_distinct(tailnum))
```

```
## # A tibble: 1 x 1
##   `n_distinct(tailnum)`
##                   <int>
## 1                   492
```

There are 492 unique airplanes that fly from NYC to PDX.

#item How many different airplanes arrived from each of the three NYC airports to Portland?

```
flights %>%
    filter(dest == "PDX") %>%
    group_by(origin) %>%
    summarize(n_distinct(tailnum))
```

```
## # A tibble: 2 x 2
##   origin `n_distinct(tailnum)`
##   <chr>                  <int>
## 1 EWR                         297
## 2 JFK                         195
```

There are two different airplanes, EWR and JFK,that arrive from the three NYC airports to Portland.

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

```
flights %>%
    filter(dest == "PDX") %>%
    mutate(delay = dep_delay > 15) %>%
    filter(!is.na(delay)) %>%
    summarize(delayed_dep = sum(delay)/n(), ontime = sum(1 -
        delay)/n())
```

```
## # A tibble: 1 x 2
##   delayed_dep ontime
##         <dbl>  <dbl>
## 1       0.268  0.732
```

About 27% of flights to PDX were delayed at departure by more than 15 minutes.

#item Is one of the New York airports noticeably worse in terms of departure delays for flights to Portland, OR than others?

```
flights %>%
    filter(dest == "PDX") %>%
    mutate(delay = dep_delay > 15) %>%
    filter(!is.na(delay)) %>%
    group_by(origin) %>%
    summarize(delayed_dep = sum(delay)/n(), ontime = sum(1 -
        delay)/n())
```

```
## # A tibble: 2 x 3
##   origin delayed_dep ontime
##   <chr>        <dbl>  <dbl>
## 1 EWR          0.296  0.704
## 2 JFK          0.247  0.753
```

It looks like there is not a NYC airport that is noticeably worse in terms of departure delays for flights to PDX. EWR is slightly worse than JFK.

#end{enumerate}

(b) Reflect and Question Comment 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?

```
flights %>%
    filter(dest == "PDX", is.na(dep_delay)) %>%
    count()  # Returns not available values
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1     6
```

I was able to answer all questions as they were all well defined. In my analysis above, I see that there are 6 missing observations.