# IMT 573: Problem Set 2 - Working with Data Part 1

Srushti Chaukhande

Due: Tuesday, October 15, 2024 by 10:00AM PT

**Collaborators:**

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

1. Download the `problem_set2.Rmd` file from Canvas. Open `problem_set2.Rmd` in RStudio and supply your solutions to the assignment by editing `problem_set1.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. Collaboration shouldn't be confused with group project work (where each person does a part of the project). Working on problem sets should be your individual contribution. More on that in point 8.
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 `ps1_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.

# Instructions

# 1. Exploring COVID-19 Data

This dataset asks you to assemble and manipulate a COVID-19 dataset, and use it for a few illustrative figures. It replicates many real-worlds problems, including mismatching variable coding and differing variable names. We expect you to use dplyr-framework but you are welcome to use something else. Many questions also include hints and suggestions, these are designed to help you in case you do not have a good idea about how to proceed. But if you know better then you are welcome to follow other routes.

1. Remember that just numerical answer is not enough. Always comment and explain your results. You can add R code inline as:

   In this data we have flights…

   (remember: these are backticks!)

2. Be sure to include well-documented (e.g. commented) code chunks, figures, tables, and clearly written text explanations as necessary. All figures should be clearly labeled and appropriately referenced within the text.

3. Be sure that each visualization adds value to your written explanation; avoid redundancy – you do not need four different visualizations of the same pattern. Don't output irrelevant, or too much of the relevant information. A few figures is helpful. A few thousand figures is just noise.

Most of the data is downloaded from John Hopkins university COVID-19 data project. The main variables in the monthly files are

**FIPS** US only. Federal Information Processing Standards code that uniquely identifies counties within the USA.

**Admin2** County name. US only.

**Province_State** Province, state or dependency name.

**Country_Region** Country, region or sovereignty name. The names of locations included on the Website correspond with the official designations used by the U.S. Department of State. Confirmed Counts include confirmed and probable (where reported).

**Deaths** Counts include confirmed and probable (where reported).

This data is supplemented with information from the US Dpartment of State, and Wikipedia. The data is on canvas: here.

## Load a single month of African data (25pt)

1. (3pt) Load the list of African countries countries-africa.csv. How many African countries are listed here?

2. (4pt) Collect all the names of covid data files covid-global… into a character vector. How many files are there?

   Hint: check out the function list.files, and its argument pattern.

The next task is to extract only data for African countries from the COVID files. Let's start simple and pick just a single file, e.g. the most recent one for October 2021. When you have managed with this code you'll loop over all the files afterwards. (But expect issues.)

3. (2pt) Load the COVID data file for October 2021. Ensure you know the variables there.

But the global data file contains not just African countries, so you have just to select the African ones from the list. Unfortunately not all the names match.

4. (3pt) How many African countries in the African Country list do you find in the covid data? Do not attempt to adjust the names for now. Hint: you can use the %in% operator to check which elements are in the list.
5. (4pt) Which African countries are not matched in the COVID data? Again, do not adjust the names for now. Hint: if you did it correctly, then you find 9 entities (some of those are more like territories).

As you can see, not all African countries/territories that are listed under the same name. We do not care too much about Mayotte and a few other islands, but we care about Congo (both DR and R), and Ivory coast.

6. (3pt) Why should we care more about these three countries and less about other entities?
7. (3pt) Next, find how are the names of these three countries (Two Congos and Ivory Coast) written in the covid data.

You may also consult US Dept of State list of countries, this is the name form COVID data is using.

8. (3pt) Amend the list of African countries in a way that you can extract all the necessary African countries (you may leave out the islands/territories) from COVID data. Demonstrate that it works.

   If you did it correctly, you should only have left with "Réunion (France)", "Western Sahara", "Cape Verde", "Mayotte (France)", "São Tomé and Príncipe", and "Saint Helena, Ascension and Tristan da Cunha (UK)" that are not included.

**Solution 1**

1. **Solution:**

Insert Response

```
countries_africa <- read.csv("/Users/srushti/Downloads/covid/countries-africa.csv", sep = "\t")
## Count the. number of rows in csv using nrows function
nrow(countries_africa)
```

```
## [1] 58
```

```
filter(countries_africa, country=="Ivory Coast")
```

```
##   rank     country population         year                          source
## 1   16 Ivory Coast 22,671,331 May 15, 2014 Preliminary 2014 census result
```

2. **Solution:**

Insert Response

```r
## The following code puts all covid-global files into a character vector "covid_files"
covid_files <- list.files(path = "/Users/srushti/Downloads/covid/", pattern = "covid-global", full.name
# Count the number of files
num_files <- length(covid_files)
# Number of files
print(num_files)
```

```
## [1] 22
```

3. **Solution:**

Insert Response

```r
# load october 21 dataset
october_21_data <- read.csv("/Users/srushti/Downloads/covid/covid-global_10-01-2021.csv", sep = "\t")
```

4. **Solution:**

Insert Response

```r
#Get unique countries from october's covid data
covid_countries <- unique(october_21_data$Country_Region)
#find african countries in covid data
matching_countries <- countries_africa$country[countries_africa$country %in% covid_countries]
#number of african countries found
length(matching_countries)
```

```
## [1] 49
```

5. **Solution:**

Insert Response

```r
# load october 21 dataset
october_21_data <- read.csv("/Users/srushti/Downloads/covid/covid-global_10-01-2021.csv", sep = "\t")
# get non matching countries
non_matching_countries <- countries_africa$country[!(countries_africa$country %in% covid_countries)]
length(non_matching_countries)
```

```
## [1] 9
```

```r
print(non_matching_countries)
```

```
## [1] "Democratic Republic of the Congo"
## [2] "Ivory Coast"
## [3] "Republic of the Congo"
## [4] "Réunion (France)"
## [5] "Western Sahara"
## [6] "Cape Verde"
## [7] "Mayotte (France)"
## [8] "São Tomé and Príncipe"
## [9] "Saint Helena, Ascension and Tristan da Cunha (UK)"
```

6. **Solution:**

Insert Response

- We should care more about DR Congo, R Congo and Ivory Coast than other countries because the other countries are sparsely populated and mostly island territories which makes them less succeptible to covid.

7. **Solution:**

Insert Response

**Names of 3 countries**

-Republic of Congo Congo (Brazzaville)

-Democratic Republic of the Congo Congo (Kinshasa)

-Ivory Coast Côte d'Ivoire

8. **Solution:**

Insert Response

```
library(stringr)
standardize_name <- function(name) {
  name <- tolower(name)
  name <- str_replace_all(name, "[^a-z ]", "")
  name <- str_trim(name)
  return(name)
}

# Standardize African country names
countries_africa$standardized_name <- sapply(countries_africa$country, standardize_name)

# Manually adjust some country names
countries_africa$standardized_name[countries_africa$standardized_name == "cote divoire"] <- "ivory coas
countries_africa$standardized_name[countries_africa$standardized_name == "congo"] <- "congo brazzaville
countries_africa$standardized_name[countries_africa$standardized_name == "democratic republic of the co

# Standardize COVID data country names
october_21_data$standardized_country <- sapply(october_21_data$Country_Region, standardize_name)
# Filter COVID data for African countries
african_covid_data <- october_21_data %>%
  filter(standardized_country %in% countries_africa$standardized_name)

# Check which African countries are not included
missing_countries <- setdiff(countries_africa$country,
                             countries_africa$country[countries_africa$standardized_name %in% african_c
```

## 2. Exploring SharksAttack Data

We are working with Global Shark Attack file (which was also explored in Problem Set 1), a compilation of all reported shark attacks on humans. See Sharkattackfile.net for more details and the original excel data sheet.

Your task is to perform data cleaning and exploratory analysis, geared toward the following question: Which country, Australia or South Africa, is more dangerous in terms of shark attacks on people?

```
gsa <- read.csv("/Users/srushti/Downloads/GSAF5.csv")
dim(gsa)
```

```
## [1] 6970   23
```

**2.1 Clean data (20 pts)** Now it is time to do some data cleaning before we can go and try to answer the question.

1. (5pt) Now let's look at whether the attack was fatal (the variable Fatal Y/N. As the first step, rename this variable to something more suitable, e.g. fatal. We ask you also to only keep variables you need below and drop all the others. (You may want to return to this question later and add/remove additional variables.) You may also rename other variables if you wish. Hint: dplyr's select function allows you to keep/drop/rename variables.
2. (5pt) Lets focus on reasonable recent time span only. Only keep the reasonably recent cases based on the year variable. Explain your reasoning when selecting the time span. How many cases are you left with? Below we work on this subset only. Now it is time to analyze if the attack was fatal.
3. (5pt) What kind of different values do you see in the fatal variable? Comment the values you see. Do you have an idea why do you see some of these figures?
4. (5pt) Now let's convert the fatal column into a logical variable: TRUE if the attack was fatal and FALSE if not. Convert the cases where you are unsure into missings. Explain your for decisions you make here. Hint: "It is too much work to handle cases like …" is a perfectly valid reason. Hint 2: check out dplyr's mutate function, and vectorized if/else ifelse.

**Solution 2.1**

1. **Solution:**

Insert Response

```
#gsa <- gsa %>% rename_with(~ ifelse(. == "" | . == "X", "fatal", .), everything())
names(gsa)[names(gsa) == "" | names(gsa) == "X"] <- "Fatal"
gsa <- gsa %>% select(Date,Year,Type,Country,State,Location,Fatal)
```

2. **Solution:**

Insert Response

- I am considering past 10 years for this analysis and 1227 cases have been reported since past 10 years. I feel 10 years is good enough to work with as it shows what areas and types of cases are most recent. I didn't go beyond 10 years because that is too long ago and many advancements like new diving destinations have been introduced since then.

•

```
#five_year_attacks <- filter(gsa, Year %in% c(2024,2023,2022,2021,2020))
gsa <- gsa %>%
  filter(Year >= 2014 & Year <= 2024)
nrow(gsa)
```

```
## [1] 1227
```

3. **Solution:**

Insert Response

I see the following values in fatal variable : 1. Y 2. N 3. " " (Likely indicates unknown or unreported status) 4. M (This appears to be an error or inconsistency in data entry) 5. F (This appears to be an error or inconsistency in data entry) 6. n (lower case for N) 7. Nq (Seems like a typo for N) 8. UNKNOWN ((Likely indicates unknown or unreported status)) 9. 2017 (Error in data entry) 10. Y x 2 (Error in data entry) 11. " N" (by mistake a space was entered) 12. "N" (by mistake a space was entered) 13. y (lower case for y)

```
unique_fatal_values <- unique(gsa$Fatal)
unique_fatal_values
```

```
## [1] "Y"  "N"  ""   "M"  "F"  "n"  "Nq"
```

4. **Solution:**

Insert Response

I am not sure how to interpret cases like '2017', 'Y X 2'

```
gsa <- gsa %>% mutate(fatal_logical = case_when(
    Fatal == "Y" ~ TRUE,
    Fatal == "N" ~ FALSE,
    Fatal == "n" ~ FALSE,
    Fatal == "Nq" ~ FALSE,
    Fatal == " N" ~ FALSE,
    Fatal == "N " ~ FALSE,
    Fatal == "y" ~ TRUE,
  ))
```

**2.2 Austalia or South Africa? (12pt)** Finally, let's try to answer the question about which country is more dangerous.

1. (4pt) Filter the data to only contain cases from these two countries. How many cases do you have from each country? Which percentage of those is fatal? Hint: check out functions table and prop.table.
2. (4pt) Now try to answer the question: which country is more dangerous? Are you able to answer it? What do you think, what can this analysis and your answer be used for? Explain your reasoning.
3. (4pt) Finally, returning to your analysis and the original data (not the one you have cleaned), do you see any ethical issues here? Can your results be misused? Can this data used in a harmful way?

**Solution 2.2**

1. **Solution:**

Insert Response

None of the 5 cases are fatal. 4 are from Australia and 1 from South Africa

```
sharks_SA_AUS <- filter(gsa, Country %in% c("Australia", "South Africa"))
sharks_SA_AUS
```

```
##          Date Year       Type      Country              State
## 1   23-Jul-24 2024 Unprovoked    Australia                NSW
## 2   18-Jul-24 2024 Unprovoked    Australia Western Australia
## 3   20-Apr-24 2024 Unprovoked    Australia   South Australia
## 4   20-Apr-24 2024 Unprovoked    Australia Western Australia
## 5 30-Dec-2022 2022   Provoked South Africa                KNZ
##                              Location Fatal fatal_logical
## 1    North Shore Beach, Port Macquarie     N         FALSE
## 2                 Trigg beach Sterling     N         FALSE
## 3                   West beach Glenelg     N         FALSE
## 4 Lighthouse bombie surf spot Exemouth     N         FALSE
## 5                         Protea Banks     N         FALSE
```

```
country_summary <- sharks_SA_AUS %>%
  group_by(Country) %>%
  summarise(
    total_cases = n(),
    fatal_cases = sum(fatal_logical == TRUE, na.rm = TRUE),
    fatal_percentage = (fatal_cases / total_cases) * 100
  )

# Display the results
print(country_summary)
```

```
## # A tibble: 2 x 4
##   Country      total_cases fatal_cases fatal_percentage
##   <chr>              <int>       <int>            <dbl>
## 1 Australia              4           0                0
## 2 South Africa           1           0                0
```

```
# Calculate overall statistics
total_cases <- sum(country_summary$total_cases)
total_fatal_cases <- sum(country_summary$fatal_cases)
overall_fatal_percentage <- (total_fatal_cases / total_cases) * 100

# Create a table of fatal vs non-fatal cases
fatality_table <- table(sharks_SA_AUS$Country, sharks_SA_AUS$fatal_logical)
print("Fatality table:")
```

```
## [1] "Fatality table:"
```

```r
print(fatality_table)
```

```
##
##                FALSE
##   Australia        4
##   South Africa     1
```

```r
# Calculate proportions
fatality_proportions <- prop.table(fatality_table, margin = 1) * 100
print("\nFatality proportions (%):")
```

```
## [1] "\nFatality proportions (%):"
```

```r
print(fatality_proportions)
```

```
##
##                FALSE
##   Australia      100
##   South Africa   100
```

2. **Solution:**

Insert Response

- Australia is more dangerous on the basis of number of cases reported which is more than South Africa in past 10 years.
- This analysis can be used to determine which country is more attack prone in recent times.

3. **Solution:**

Insert Response

- I do see an Ethical issue which is that the resulting data set of attacks from SA and Australia from past 10 years contains only 5 entries which is not enough to make a claim.
- Privacy concerns: The dataset includes personal information such as names, ages, and specific locations. This could potentially be used to identify individuals, especially in cases of fatalities, which may cause distress to families and survivors.
- Bias in reporting: There might be inconsistencies in how incidents are reported across different regions or over time. This could lead to skewed perceptions of risk in certain areas.
- Economic impact: Misuse of this data could negatively impact tourism in coastal areas, affecting local economies that depend on beach and water activities.
- Neglect of broader context: Focusing solely on shark attacks ignores other, often more significant, risks associated with ocean activities and may divert attention from more pressing marine conservation issues.

# 3. Exploring NYC Data (35 pts)

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

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

**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.

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

1. (2 pts) How many fligths out of NYC are there in the data?
2. (2 pts) How many NYC airports are included in this data? Which airports are these?
3. (3 pts) Into how many airports did the airlines fly from NYC in 2013?
4. (3 pts) How many flights were there from NYC to Seattle (airport code `SEA`)?
5. (3 pts) Were the any flights from NYC to Spokane `(GAG)`?
6. (3 pts) What about missing destination codes? Are there any destinations that do not look like valid airport codes (i.e. three-letter-all-upper case)?
7. (4 pts) 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?

**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:

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

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

**Solution 3.1**

1. How many flights out of NYC are there in the data?

**Solution:**

Insert Response

We can count the number of rows using summarize then use pull to extract the count or simply calculate number of rows

```r
data(flights)
total_flights <- flights %>% summarise(count = n()) %>%pull(count)
total_flights
```

```
## [1] 336776
```

```r
nrow(flights)
```

```
## [1] 336776
```

2. How many NYC airports are included in this data? Which airports are these?

**Solution:**

Insert Response

3. Into how many airports did the airlines fly from NYC in 2013?

**Solution:**

Insert Response

```r
head(flights)
```

```
## # A tibble: 6 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
## # 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>
```

```r
distinct_destinations <- flights %>%
  summarise(count = n_distinct(dest))

# Print the results
print(distinct_destinations)
```

```
## # A tibble: 1 x 1
##   count
##   <int>
## 1   105
```

4. How many flights were there from NYC to Seattle (airport code `SEA`)?

**Solution:**

Insert Response

```r
# Filter flights to Seattle (SEA)
nyc_to_sea <- flights %>%
  filter(dest == "SEA")

# Count the number of flights
num_flights_to_sea <- nrow(nyc_to_sea)
print(num_flights_to_sea)
```

```
## [1] 3923
```

5. Were the any flights from NYC to Spokane (`GAG`)?

**Solution:**

Insert Response

```r
nyc_to_spokane <- flights %>%
  filter(dest == "GAG")
nrow(nyc_to_spokane)
```

```
## [1] 0
```

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)?

**Solution:**

Insert Response

```r
missing_dest <- flights %>% filter(is.na(dest)) %>% nrow()
missing_dest
```

```
## [1] 0
```

```r
# Check for invalid destination codes (not 3 uppercase letters)
invalid_dest <- flights %>% filter(!grepl("^[A-Z]{3}$", dest)) %>% select(dest) %>% distinct()
invalid_dest
```

```
## # A tibble: 0 x 1
## # i 1 variable: dest <chr>
```

7. 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?

**Solution:**

Insert Response

Yes, I was able to answer all the questions so far and the questions are well defined. Nycflightsdata is good to learn dplyr functions and extract useful information out.

**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) (3 pts) Typical Delays** What is the typical delay of flights in this data?

**Solution:**

Insert Response

```
new_var <- flights %>% mutate(mean_dep_delay = mean(dep_delay, na.rm = TRUE),median_dep_delay = median(
print(new_var)
```

```
## # A tibble: 336,776 x 23
##     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
## # i 336,766 more rows
## # i 15 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>, mean_dep_delay <dbl>,
## #   median_dep_delay <dbl>, mean_arr_delay <dbl>, median_arr_delay <dbl>
```

**(2) (3 pts) 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?

**Solution:**

Insert Response

- Missing data: There are 8,255 NA values in 'dep_delay' There are 9,430 NA values in 'arr_delay'
- Implausible or invalid entries: Departure delays range from -43 to 1301 minutes Arrival delays range from -86 to 1272 minutes The negative delays are somewhat unexpected but could represent early departures or arrivals. However, the extremely large delays (over 20 hours) might be worth investigating further.

I used the mean and median to calculate flight delay

**(3) (3 pts) 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.

**Solution:**

Insert Response

```r
# Create a new variable for total delay
flights_with_delay <- flights %>%
  mutate(total_delay = dep_delay + arr_delay)

# Calculate average delays by destination
destination_delays <- flights_with_delay %>%
  group_by(dest) %>%
  summarise(
    avg_total_delay = mean(total_delay, na.rm = TRUE),
    avg_dep_delay = mean(dep_delay, na.rm = TRUE),
    avg_arr_delay = mean(arr_delay, na.rm = TRUE),
    num_flights = n()
  ) %>%
  arrange(desc(avg_total_delay))

# Display the top 3 worst destinations for delays
print(head(destination_delays, 3))
```

```
## # A tibble: 3 x 5
##   dest  avg_total_delay avg_dep_delay avg_arr_delay num_flights
##   <chr>           <dbl>         <dbl>         <dbl>       <int>
## 1 CAE              75.6          35.6          41.8         116
## 2 TUL              68.5          34.9          33.7         315
## 3 OKC              59.8          30.6          30.6         346
```

**(4) (3 pts) 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.

'**Solution:**

Insert Response

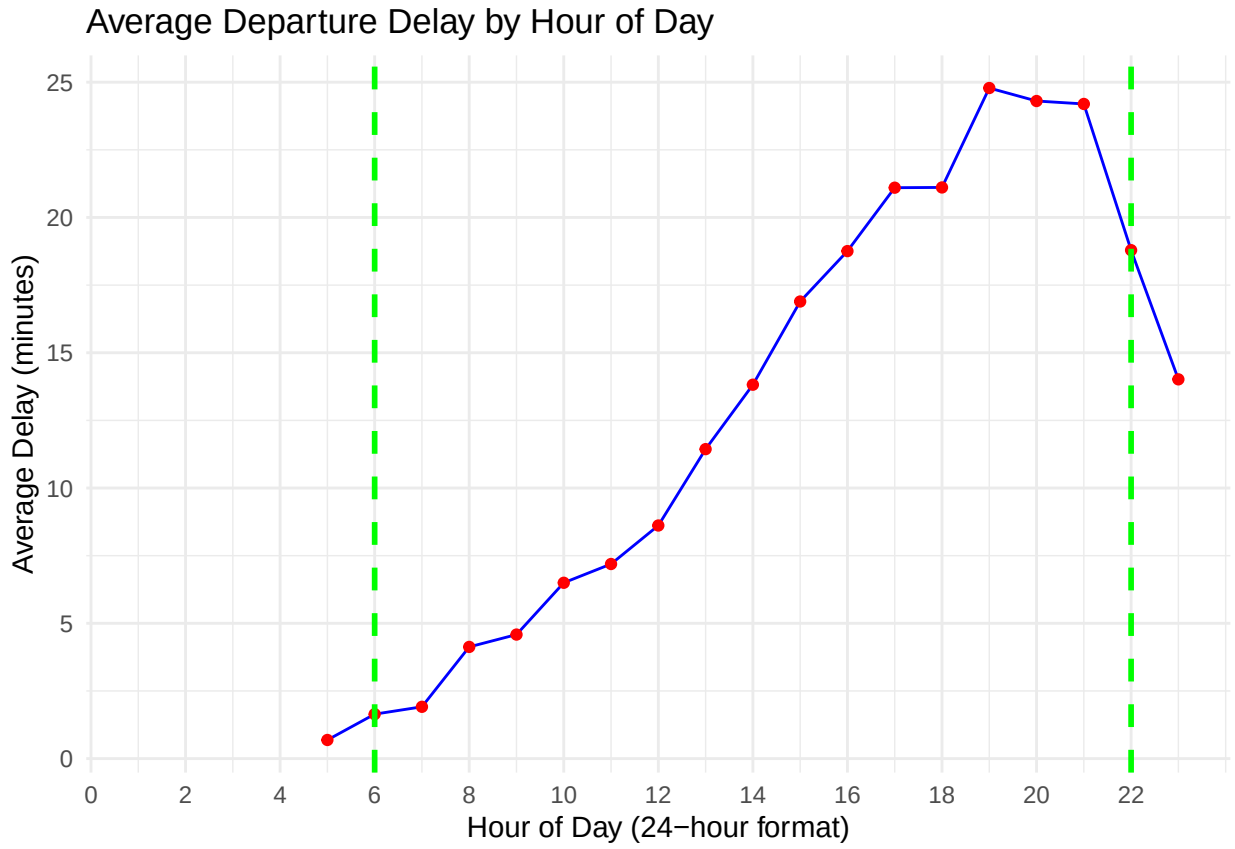
```r
# Calculate average delay for each hour
hourly_delays <- flights %>%
  group_by(hour) %>%
  summarise(avg_delay = mean(dep_delay, na.rm = TRUE)) %>%
  filter(!is.na(hour))

# Create the plot
ggplot(hourly_delays, aes(x = hour, y = avg_delay)) +
  geom_line(color = "blue") +
  geom_point(color = "red") +
  labs(title = "Average Departure Delay by Hour of Day",
       x = "Hour of Day (24-hour format)",
       y = "Average Delay (minutes)") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(0, 23, by = 2)) +
  geom_vline(xintercept = c(6, 22), linetype = "dashed", color = "green", size = 1) +
  annotate("text", x = 3, y = max(hourly_delays$avg_delay), label = "Night", color = "darkgreen") +
  annotate("text", x = 14, y = max(hourly_delays$avg_delay), label = "Day", color = "darkgreen") +
  annotate("text", x = 23, y = max(hourly_delays$avg_delay), label = "Night", color = "darkgreen")
```

```
## 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.
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_line()`).
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_point()`).
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_text()`).
## Removed 1 row containing missing values or values outside the scale range
## (`geom_text()`).
## Removed 1 row containing missing values or values outside the scale range
## (`geom_text()`).
```




```
# Print the data
print(hourly_delays)
```

```
## # A tibble: 20 x 2
##     hour avg_delay
```

```
##    <dbl>     <dbl>
##  1     1   NaN
##  2     5     0.688
##  3     6     1.64
##  4     7     1.91
##  5     8     4.13
##  6     9     4.58
##  7    10     6.50
##  8    11     7.19
##  9    12     8.61
## 10    13    11.4
## 11    14    13.8
## 12    15    16.9
## 13    16    18.8
## 14    17    21.1
## 15    18    21.1
## 16    19    24.8
## 17    20    24.3
## 18    21    24.2
## 19    22    18.8
## 20    23    14.0
```

**(5) (3 pts) 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?

**Solution:**

After completing the exploratory analyses, there are indeed some concerns about the questions and findings: 1. Seasonal patterns: "How do average flight delays vary by month in 2013 for NYC airports?" 2. Carrier comparison: "Which airlines have the highest and lowest average delays for flights departing from NYC airports in 2013?" 3. Weather impact: "Is there a correlation between weather conditions (e.g., precipitation, visibility) and flight delays at NYC airports?"