INFX 573: Problem Set 1 - Exploring Data

Amol Surve

Due: Monday, October 11, 2016

Collaborators: Abhishek Gupta

Instructions:

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

1. Download the `problemset1.Rmd` file from Canvas. Open `problemset1.Rmd` in RStudio and supply your solutions to the assignment by editing `problemset1.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.
4. Collaboration on problem sets is acceptable, and even encouraged, but each student must turn in an individual write-up in his or her own words and his or her own work. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. 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 R Markdown file to `YourLastName_YourFirstName_ps1.Rmd`, knit a PDF and submit both the PDF file on Canvas.

Setup:

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

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

Problem 1: Exploring the NYC Flights Data

In this problem set we will use the data on all flights that departed NYC (i.e. JFK, LGA or EWR) in 2013. You can find this data in the `nycflights13` R package.

(a) Importing and Inspecting Data:

Load the data and describe in a short paragraph how the data was collected and what each variable represents. Perform a basic inspection of the data and discuss what you find.

```r
#Now Loading the Dataset in myData dataframe
myData<-flights

head(myData)
```

```
## # A tibble: 6 × 19
##    year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>    <int>          <int>     <dbl>    <int>
## 1  2013     1     1      517            515         2      830
## 2  2013     1     1      533            529         4      850
## 3  2013     1     1      542            540         2      923
## 4  2013     1     1      544            545        -1     1004
## 5  2013     1     1      554            600        -6      812
## 6  2013     1     1      554            558        -4      740
## # ... with 12 more variables: sched_arr_time <int>, 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
#outouts first 6 observatios in the dataset

tail(myData)
```

```
## # A tibble: 6 × 19
##    year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>    <int>          <int>     <dbl>    <int>
## 1  2013     9    30       NA           1842        NA       NA
## 2  2013     9    30       NA           1455        NA       NA
## 3  2013     9    30       NA           2200        NA       NA
## 4  2013     9    30       NA           1210        NA       NA
## 5  2013     9    30       NA           1159        NA       NA
## 6  2013     9    30       NA            840        NA       NA
## # ... with 12 more variables: sched_arr_time <int>, 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
#outputs last 6 observations in the dataset

str(myData)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    336776 obs. of  19 variables:
##  $ year          : int  2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
##  $ month         : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ day           : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ dep_time      : int  517 533 542 544 554 554 555 557 557 558 ...
##  $ sched_dep_time: int  515 529 540 545 600 558 600 600 600 600 ...
##  $ dep_delay     : num  2 4 2 -1 -6 -4 -5 -3 -3 -2 ...
##  $ arr_time      : int  830 850 923 1004 812 740 913 709 838 753 ...
##  $ sched_arr_time: int  819 830 850 1022 837 728 854 723 846 745 ...
##  $ arr_delay     : num  11 20 33 -18 -25 12 19 -14 -8 8 ...
##  $ carrier       : chr  "UA" "UA" "AA" "B6" ...
##  $ flight        : int  1545 1714 1141 725 461 1696 507 5708 79 301 ...
##  $ tailnum       : chr  "N14228" "N24211" "N619AA" "N804JB" ...
##  $ origin        : chr  "EWR" "LGA" "JFK" "JFK" ...
##  $ dest          : chr  "IAH" "IAH" "MIA" "BQN" ...
##  $ air_time      : num  227 227 160 183 116 150 158 53 140 138 ...
##  $ distance      : num  1400 1416 1089 1576 762 ...
##  $ hour          : num  5 5 5 5 6 5 6 6 6 6 ...
##  $ minute        : num  15 29 40 45 0 58 0 0 0 0 ...
##  $ time_hour     : POSIXct, format: "2013-01-01 05:00:00" "2013-01-01 05:00:00" ...
```

```
#outputs the structure of the dataset

summary(myData)
```

```
##       year          month             day           dep_time
##  Min.   :2013   Min.   : 1.000   Min.   : 1.00   Min.   :   1
##  1st Qu.:2013   1st Qu.: 4.000   1st Qu.: 8.00   1st Qu.: 907
##  Median :2013   Median : 7.000   Median :16.00   Median :1401
##  Mean   :2013   Mean   : 6.549   Mean   :15.71   Mean   :1349
##  3rd Qu.:2013   3rd Qu.:10.000   3rd Qu.:23.00   3rd Qu.:1744
##  Max.   :2013   Max.   :12.000   Max.   :31.00   Max.   :2400
##                                                  NA's   :8255
##  sched_dep_time   dep_delay          arr_time    sched_arr_time
##  Min.   : 106   Min.   : -43.00   Min.   :   1   Min.   :   1
##  1st Qu.: 906   1st Qu.:  -5.00   1st Qu.:1104   1st Qu.:1124
##  Median :1359   Median :  -2.00   Median :1535   Median :1556
##  Mean   :1344   Mean   :  12.64   Mean   :1502   Mean   :1536
##  3rd Qu.:1729   3rd Qu.:  11.00   3rd Qu.:1940   3rd Qu.:1945
##  Max.   :2359   Max.   :1301.00   Max.   :2400   Max.   :2359
##                 NA's   :8255      NA's   :8713
##    arr_delay          carrier              flight       tailnum
##  Min.   : -86.000   Length:336776      Min.   :   1   Length:336776
##  1st Qu.: -17.000   Class :character   1st Qu.: 553   Class :character
##  Median :  -5.000   Mode  :character   Median :1496   Mode  :character
##  Mean   :   6.895                      Mean   :1972
##  3rd Qu.:  14.000                      3rd Qu.:3465
##  Max.   :1272.000                      Max.   :8500
##  NA's   :9430
##     origin              dest              air_time        distance
##  Length:336776      Length:336776      Min.   : 20.0   Min.   :  17
##  Class :character   Class :character   1st Qu.: 82.0   1st Qu.: 502
##  Mode  :character   Mode  :character   Median :129.0   Median : 872
##                                        Mean   :150.7   Mean   :1040
##                                        3rd Qu.:192.0   3rd Qu.:1389
##                                        Max.   :695.0   Max.   :4983
##                                        NA's   :9430
##       hour           minute        time_hour
##  Min.   : 1.00   Min.   : 0.00   Min.   :2013-01-01 05:00:00
##  1st Qu.: 9.00   1st Qu.: 8.00   1st Qu.:2013-04-04 13:00:00
##  Median :13.00   Median :29.00   Median :2013-07-03 10:00:00
##  Mean   :13.18   Mean   :26.23   Mean   :2013-07-03 05:02:36
##  3rd Qu.:17.00   3rd Qu.:44.00   3rd Qu.:2013-10-01 07:00:00
##  Max.   :23.00   Max.   :59.00   Max.   :2013-12-31 23:00:00
##
```

```
#Displays the insights such as mean, median, missing values for columns, etc.

distinct_df = myData %>% distinct(month)
#Displays the distinct values of the months to check whether the data is for whole year or not.
print(distinct_df)
```

```
## # A tibble: 12 × 1
##    month
##    <int>
## 1      1
## 2     10
## 3     11
## 4     12
## 5      2
## 6      3
## 7      4
## 8      5
## 9      6
## 10     7
## 11     8
## 12     9
```

```
#prints the distinct values of the months

sum(!duplicated(myData$dest))
```

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

```
#Displays the unique destinations where flights are going

sum(!duplicated(myData$origin))
```

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

```
#Displays the unique destinations where flights are going

sum(!duplicated(myData$carrier))
```

```
## [1] 16
```

```
#Displays the unique destinations where flights are going
```

Describing the variables in the myData Dataset

There are total 19 variables in the dataset

1. year: int. It tells us the year of the flights departure as well as arrival for all the observations

2. month: int. It tells us the month number of the flights departure as well as arrival for all the observations

3. day: int. It tells us the day of the flights departure as well as arrival for all the observations

4. dep_time: int. it gives the 24 hour (hour-minutes) format of the flight's actual departure time

5. sched_dep_time: int. It gives the 24 hour (hour-minutes) format of the flights scheduled departure time

6. dep_delay: num. Gives the departure delay for the flights which is having both negative or positive values. Negative delay shows that flight is departed earlier.

7. arr_time: int. Gives the arrival delay for the flights which is having both negative or positive values. Negative delay shows the earlier arrival.
8. sched_arr_time: int.Gives the scheduled arrival time for theflight in 24 hour (hour-minutes) format.
9. arr_delay: num. Gives the arrival delay for the flights having both positive and negative values. Negative values show arrival before scheduled time
10. carrier: chr. Two letter airline carrier abbreviation
11. flight: int. Displays the flight number
12. tailnum: chr. Describes the tail number for the plane
13. origin: chr. Displays the flight arrival airport with three letter abbreviation
14. dest: chr. Displays the Destination of the flight with three letter abbreviation
15. air_time: num. Displays the total duration of the aircraft in the air
16. distance: num. Displays total distanced travelled by the airplane
17. hour: num. Displays hour of the scheduled departure of the flight
18. minute: num. Displays minute of the scheduled departure of the flight
19. time_hour: POSIXct. Scheduled departure date and hout of the flight in the Date-Time format.

Data Inspection:

Based on the str() function, the overall structure of the data is shown with all the variables and their types.

Summary() gives the insights of the data such as:

1. Departure time for 8255 flights are missing, that lead to missing departure delay values. Also, arrival time for 8713 flights are missing that lead to 9430 missing arrival delays.Thus,
2. There are just 3 distinct origin and 105 distinct destination values both stored as char. Also, there are 16 distinct carrier values stored as char. We can convert them to factors since using factors over char as a variable would help in further search because factor limits the value and would optimize the looping for large datasets.

(b) Formulating Questions:

Consider the NYC flights data. Formulate two motivating questions you want to explore using this data. Describe why these questions are interesting and how you might go about answering them.

Question 1:

What are the possible factors that are affecting the flight delays?

Why?:

The question is interesting from the passenger's point of view. One of the most frustrating parts of the journey is flight delays. Aircraft carriers with lower fight delays is highly responsible to pull more passengers thus increasing the profitability of the carrier. Thus, efficiency can be calculated for the carriers based on delays and for that it is highly important to find out all the factors contributing to delays.

Approach to answer the question 1:

1. In order to apprach this question, there are number of variables that we need to consider:

1.1. dep_delay (summing with arr_delay to find total delay) 1.2. arr_delay (summing with arr_delay to find total delay) 1.3. month 1.4. total_delay(arr_delay+dep_delay) 1.5. carrier 1.6. origin 1.7. hour

2. Sum of departure delays and arrival delays to get total delay and then group by variable of interest such as carrier, hour, origin, etc.

3. I am also merging the airlines dataset here to get the carrier's full names

---

Question 2:

Find out the natural factors that are most probably responsible for affecting the speed of the flight?

Why?:

Flight speed is affected by several natural factors and it would be interesting to see the factors responsible for increasing or deceasing the flight speed as it is also responsible for affecting the delay.

Approach to answer the question 2:

1. In order to get the speed of the flight, divide the total distance by total time taken for the flight.

2. Add that as a new variable in the dataset for indicating the speed of the flight
3. Plot the speed versus various natural components like visibility, wind speed, etc. to analyze the effect of natural variables on flight speed.

(c) Exploring Data:

For each of the questions you proposed in Problem 1b, perform an exploratory data analysis designed to address the question. At a minimum, you should produce two visualizations related to each question. Be sure to describe what the visuals show and how they speak to your question of interest.

Question 1:

Based on the flights details, find out the factors responsible for the arrival and departure delays?

In order to deal with the delays, we are considering the total delay which can be calculated as sum of the departure and arrival delays and storing it as a DelaySum. But while plotting the delay, instead of just taking it as a sum, considering the mean since mean is always a good measure in order to deal wit drastically varying numbers.

```
myData$DelaySum<-rowSums(myData[,c("dep_delay","arr_delay")],na.rm=TRUE)#creating a new column called "
myData[,c("dep_delay","arr_delay","DelaySum")]#Checking whether DelaySum is added or not
```

```
## # A tibble: 336,776 × 3
##    dep_delay arr_delay DelaySum
##        <dbl>     <dbl>    <dbl>
## 1          2        11       13
## 2          4        20       24
## 3          2        33       35
## 4         -1       -18      -19
## 5         -6       -25      -31
## 6         -4        12        8
## 7         -5        19       14
## 8         -3       -14      -17
## 9         -3        -8      -11
## 10        -2         8        6
## # ... with 336,766 more rows
```

```
flightnames<-airlines#adding airlines dataset into flightnames frame in order to get the full flight na
myData<-merge(myData, flightnames, by="carrier")
head(myData)#to display the first 6 observations
```

```
##   carrier year month day dep_time sched_dep_time dep_delay arr_time
## 1      9E 2013     2   5      827            830        -3     1032
## 2      9E 2013     8  23     1901           1905        -4     2051
## 3      9E 2013     6   2      805            810        -5      949
## 4      9E 2013    10  26     2139           1935       124     2358
## 5      9E 2013     7   7       NA           2030        NA       NA
## 6      9E 2013     2  18     1459           1505        -6     1621
##   sched_arr_time arr_delay flight tailnum origin dest air_time distance
## 1           1023         9   4220  N8698A    JFK  RDU       78      427
## 2           2103       -12   3360  N926XJ    JFK  PIT       61      340
## 3           1027       -38   3538  N925XJ    JFK  MSP      145     1029
## 4           2145       133   3470  N928XJ    JFK  CVG      102      589
## 5           2156        NA   4218    <NA>    JFK  PHL       NA       94
## 6           1637       -16   3393  N910XJ    JFK  DCA       46      213
##   hour minute           time_hour DelaySum              name
## 1    8     30 2013-02-05 08:00:00        6 Endeavor Air Inc.
## 2   19      5 2013-08-23 19:00:00      -16 Endeavor Air Inc.
## 3    8     10 2013-06-02 08:00:00      -43 Endeavor Air Inc.
## 4   19     35 2013-10-26 19:00:00      257 Endeavor Air Inc.
## 5   20     30 2013-07-07 20:00:00        0 Endeavor Air Inc.
## 6   15      5 2013-02-18 15:00:00      -22 Endeavor Air Inc.
```

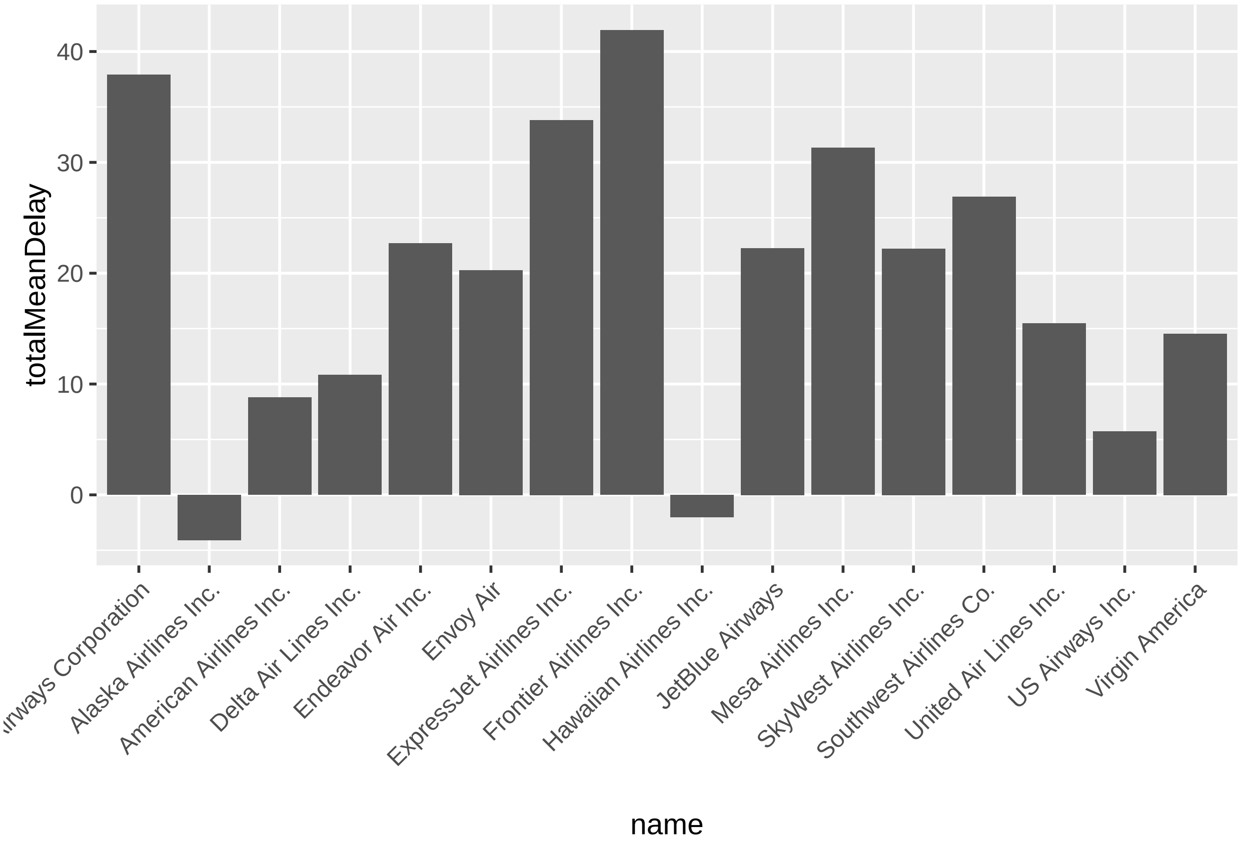
```
head(myData[,c("carrier","name")])
```

```
##   carrier              name
## 1      9E Endeavor Air Inc.
## 2      9E Endeavor Air Inc.
## 3      9E Endeavor Air Inc.
## 4      9E Endeavor Air Inc.
## 5      9E Endeavor Air Inc.
## 6      9E Endeavor Air Inc.
```

1.1 Plotting Ariline Carrier Name vs Mean Total Delay

In order to analyze the effect of delay on carrier to get to know which carrier is with the highest and the lowest delay, the values are plotted. As you can see in the graph, the Alaska Airlines is the one with the lowest delay Frontier Airlines is the one with maximum delay.

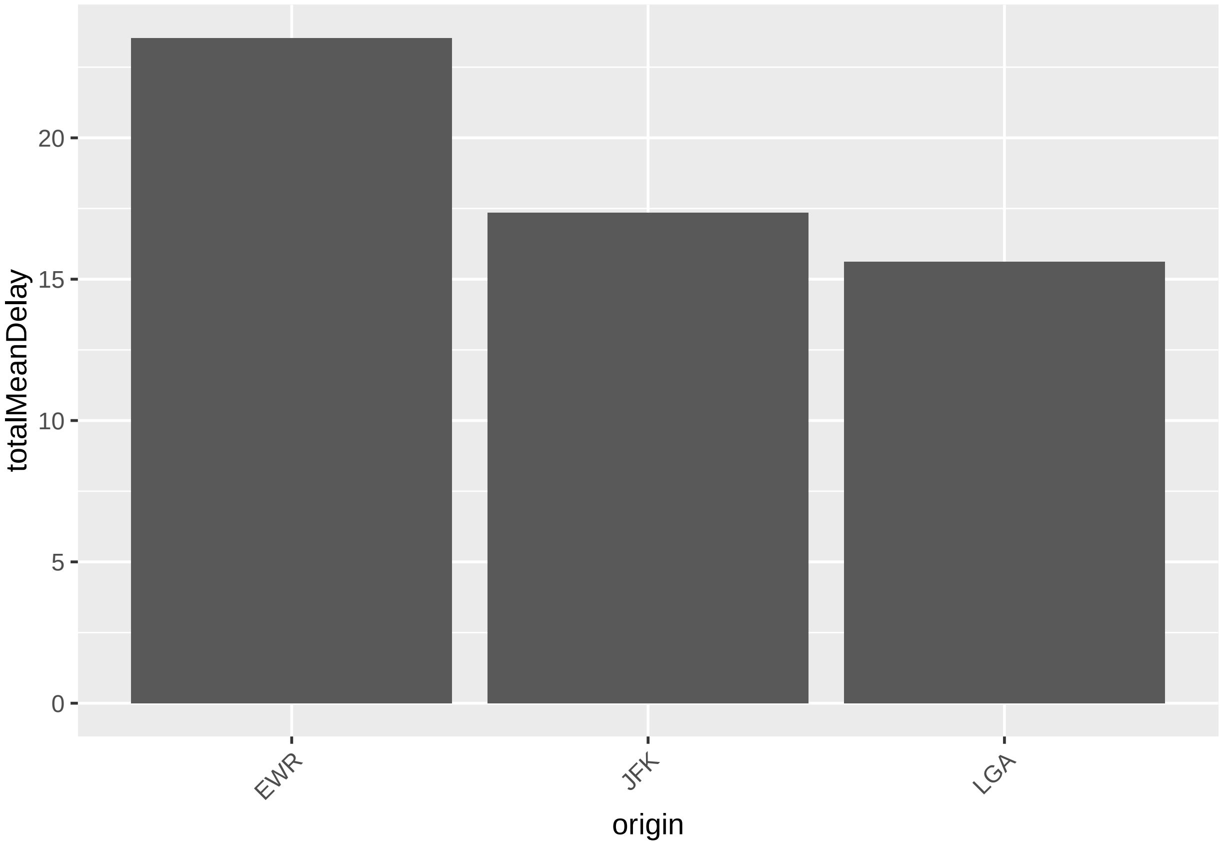
```
myData %>%
  dplyr::group_by(name) %>%#grouping based on name
  dplyr::summarise(totalMeanDelay = mean(DelaySum)) %>%#assigning mean of the delay as totalDelay in or
  ggplot(aes(x = name, y = totalMeanDelay)) + geom_bar(stat = "identity") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))#plotting the mean delay for every carrier
```



1.2 Plotting Origin vs Mean Delay

After analyzing the delay with respect to the carriers, I wanted to analyze it with respect to the origin to see which origin is the one with maximum delay. Plotting is done using dplyr where you can play around with details such as lebels angels, range, etc. As you can see in the graph, EWR is the one with maximum delay and LGA is the one with minimum delay.

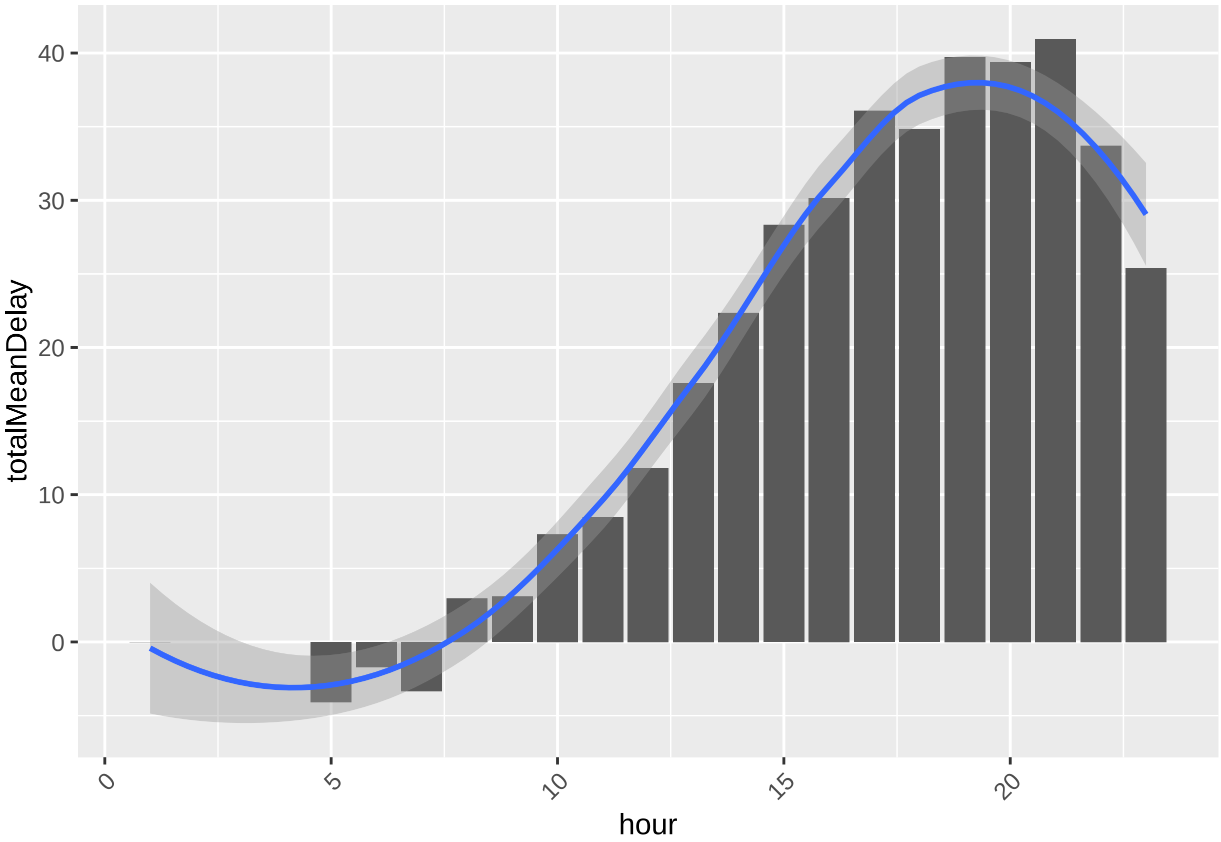
```
myData %>%
  dplyr::group_by(origin) %>%#grouping based on origin
  dplyr::summarise(totalMeanDelay = mean(DelaySum)) %>%#assigning mean of the delay as totalDelay in or
  ggplot(aes(x = origin, y = totalMeanDelay)) + geom_bar(stat = "identity") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))#plotting the mean delay for every carrier
```



1.3 Plotting Hour vs Mean Delay

It will be interesting to see the delays in daily patterns to analyze the most favorable as well as the least favorable hour of the day to fly. as you can see in the graph, first of all, there are no flights leaving between 12:00AM to 4:30AM. Also, flights till 7:30AM are with negative delays indicating that they are leavign early. As the day passes, the delay is increasing with 9:00PM being the most delayed hour and then again starts decreasing as it approaches to 12:00AM.

```
myData %>%
  dplyr::group_by(hour) %>%#grouping based on origin
  dplyr::summarise(totalMeanDelay = mean(DelaySum)) %>%#assigning mean of the delay as totalDelay in or
  ggplot(aes(x = hour, y = totalMeanDelay)) + geom_bar(stat = "identity") + geom_smooth()+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))#plotting the mean delay for every carrier
```



Question 2:

Find out the natural factors that are most probably responsible for affecting the speed of the flight?

Natural factors are hugely responsible for affecting the speed of the flights which subsequently delays them. Hence, it is interesting to see how the flight speed gets affected with natural factors such as wind speed, visibility.

Here, we are calculating the speed by dividing the total distance travelled by the flight by total time of travel. We will be multiplying it with 60 to get the speed per hour. We are also assigning zero values to the flight speed data which is unavailable.

```
myData$flightSpeed<-(myData[,"distance"]/myData[,"air_time"])*60#getting the flight speed in miles per
myData$flightSpeed[is.na(myData$flightSpeed)]<-0#assigning zero to the NA observations
```

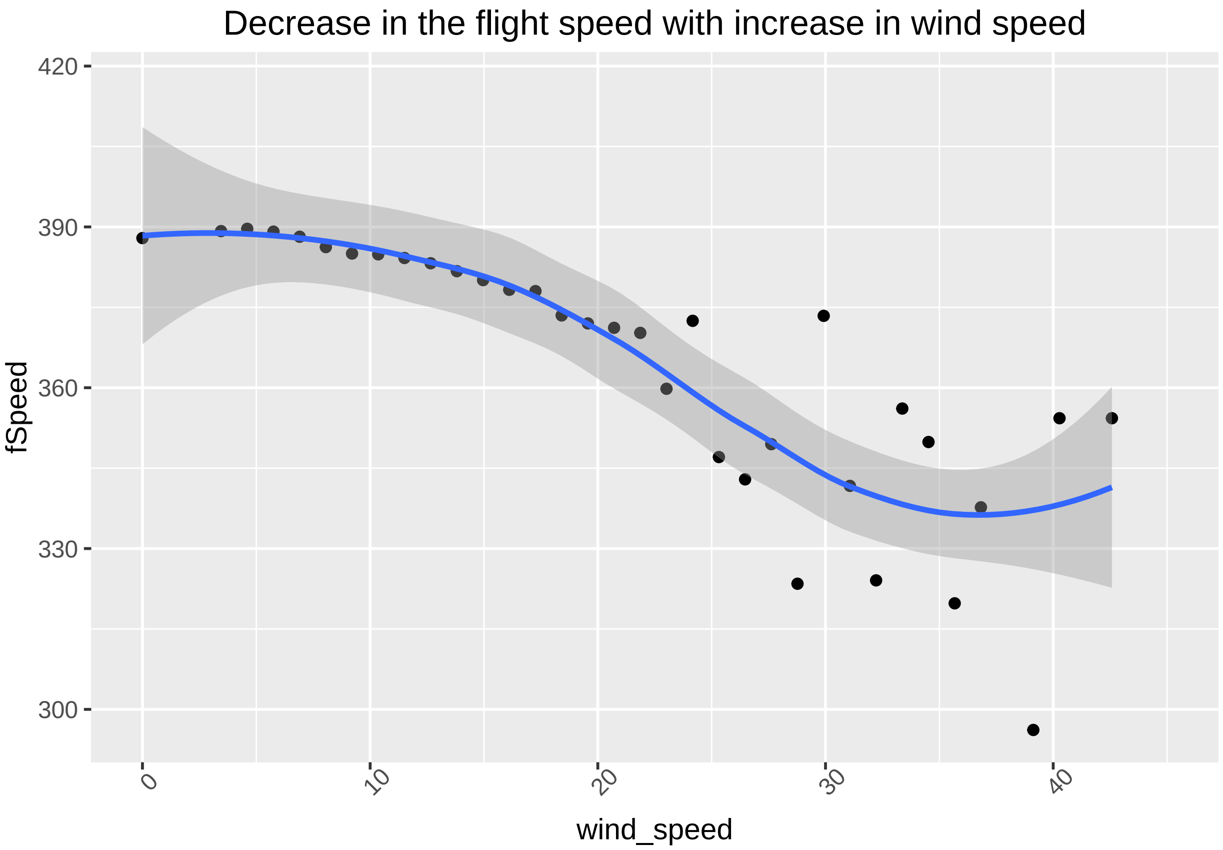
Now, merging with the weather dataset in order to access the variables such as wind_speed, visibility, etc.

```
myData<-merge(myData,weather,by="time_hour")#merging with the weather dataset by "time_hour variable"
```

2.1 Plotting Wind Speed vs Flight speed

As you can see from the graph, the flight speed decreases with the increase in the wind speed. Also, we are taking values till 45 usinf xlim(0,45) since the values beyond 45 for windspeed are not available.

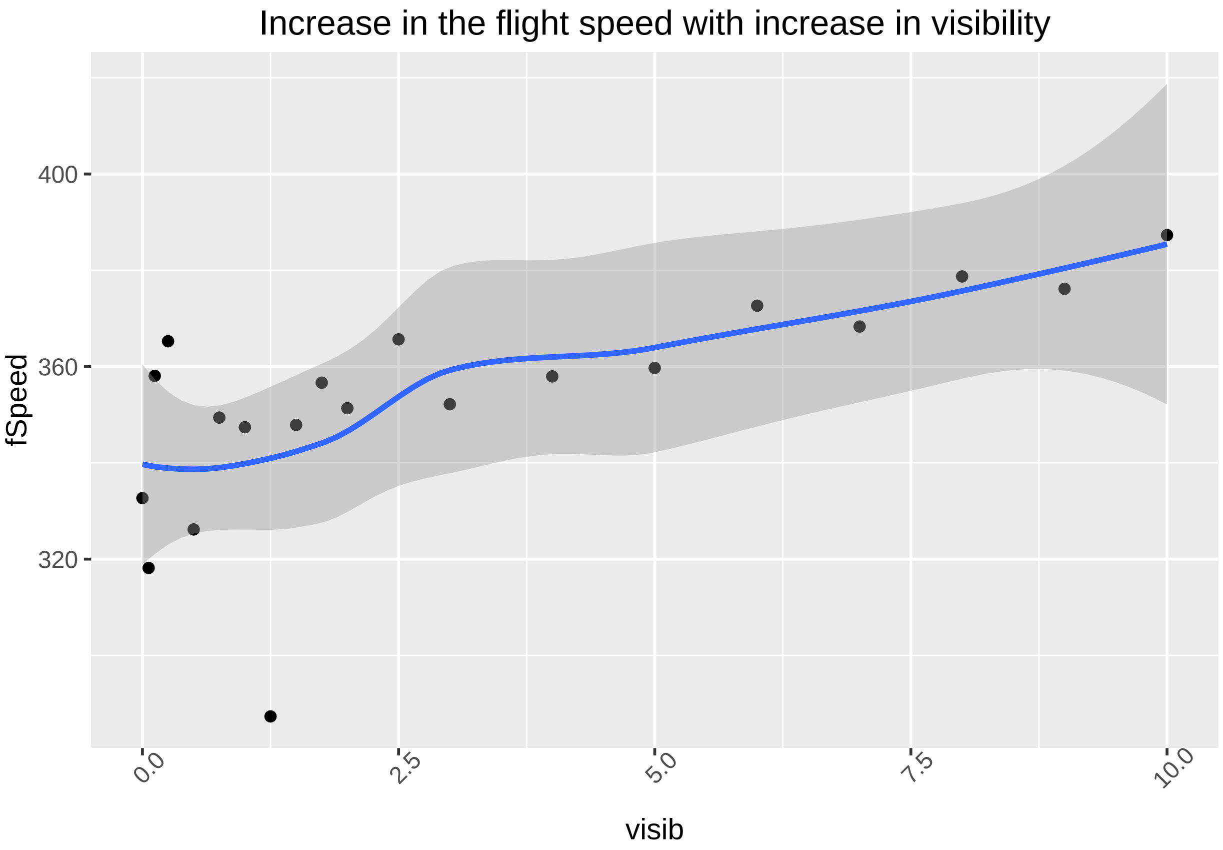
```
myData %>%
  dplyr::group_by(wind_speed) %>%#grouping based on origin
  dplyr::summarise(fSpeed = mean(flightSpeed)) %>%#assigning mean of the delay as totalDelay in order t
  ggplot(aes(x = wind_speed, y = fSpeed)) + geom_point() + geom_smooth()+
  theme(axis.text.x = element_text(angle = 45))+xlim(0,45)+ggtitle("Decrease in the flight speed with i
```



2.2 Plotting Visibility vs Flight Speed

As you can see in the graph, as the visibility increases, the flight speed is also increasing.

```
myData %>%
  dplyr::group_by(visib) %>%#grouping based on origin
  dplyr::summarise(fSpeed = mean(flightSpeed)) %>%#assigning mean of the delay as totalDelay in order t
  ggplot(aes(x = visib, y = fSpeed)) + geom_point() + geom_smooth()+
  theme(axis.text.x = element_text(angle = 45))+ggtitle("Increase in the flight speed with increase in
```



(d) Ethical Concerns:

After completing the exploratory analysis from Problem 1c, do you have any concerns about your findings? Comment on any ethical and/or privacy concerns you have with your analysis.

1. As we have analyzed the effect of natural factors, we have not really considered all the natural factors that might affect the flight speed. Wind Speed affects the flight speed but the direction of the wind i.e. whether it is in the direction of the flight or in the direction opposite to that of the flight affects it differently.
2. For anylysis to be concrete, it is desirable to have maximum number of observations in the dataset available. But, as we had seen in the data inspection, there are lot of NA values. For example, wind direction is having 20651 NA values whereas pressure is having 108743 NA values. We also don't have values for 28216 flights regarding their air time. Also, 8255 arrival delays and 9430 departure delays are missing which can significantly affect the carrier delay analysis if these missing values turned out to be long. Hence, large amount of missing data raises significant concerns regarding the analysis.
3. Also, the analysis should not be generalized since it is specific to the NYC fLights data. Hence, when it comes to the weather conditions on other regions, the analysis might be completely different based on the weather conditions and also the variables such as carriers, number of aircrafts flying every day, capacity of the airport, etc. Hence, the alayisis is subjective based on my point of view as well as limited to nys flights data. More data along with numerous other variables will be required to create a near general flights analysis.