final_project_draft

Omer Shasman

5/3/2022

Introduction

On Time Performance analysis of an airline network - This is an important metric for the airline that is calculated as the percentage of flights which are delayed by more than 14 minutes while the aircraft arrives at the gate. There are multiple reasons which contribute to the variation in OTP. An analysis of the OTP metric breaking it down into its individual components namely different delay and historical delays can provide insights into how the OTP for an airline can be managed by operational/process changes. The Department of Transport releases the flight level, On Time Performance data. This dataset also has various other factors which affect the Arrival Delay of a flight. An exploratory analysis of this data with the Arrival Delay as the response variable analyzed against different dimensions provided in the dataset can reveal several insights to improve the OTP of an Airline.

What

As part of my Final Project, I am planning to use a subset of OTP data to perform analysis of delays on actual file arrivals focusing on one particular Station and Airline. Since airline operations are very complex, the arrival delays itself can be due to varying factors, like weather delay, carrier delays, security delays, Late aircraft delay. . . etc or any combinations of any of these in general. My focus is only on 2 types of delays so that I can minimize the complexities in data structures and limit any repeating processes or steps, and rather focus on how to manipulate and do analysis/inference with few variables. Hence I will be considering only 5 years data ranging from year 2014 till 2019 two types of delays "Weather Delays" and "Carrier Delays"

Why

I thought airline is an interesting business with lot of complex operation/data and business itself is most of us are familiar with. Also, with the time constraint we have, there were few sites like Kaggle and DOT On-Time_performance

This data is presented as yearly file in csv format, I have to merge different years data using dply bind command to append rows at the end and build one file.

How

Use RMarkdown and explore Rfunctions that can integrate some of the topics we learned in the class for flight arrival delay analysis. The following are steps which will be followed as part of the project

- Load data into R Markdown using R chunks
- Merge Data using ddplyr

- Filter/melt/massage data using Tidy Data approach
- Use sampling strategy for identifying sample observations.
- Use Stats function to determine mean, median, IQR,...
- Regression - Find any co-relation between total flights arriving at a particualr airport and delays to identify if it's the airport operational/capacity issue or not.
- Could hosted data and R Markdown interface. Pull data from AWS S3 buckets than loading from local machine.

Body

This project perform analysis of flight arrival delays focusing on one particular Station and Airline. Since airline delays are unavoidable there is always a chance that a particualr flight will be delayed. I think this analysis can be used to further study on why a particular delay happens and if the process/schedule/operations can be enhanced or refined to minimize the delay risk in future flights.

Packages Required

```
library(knitr) ##for printing tables in R Markdown
library(dplyr) ##for data munging
```

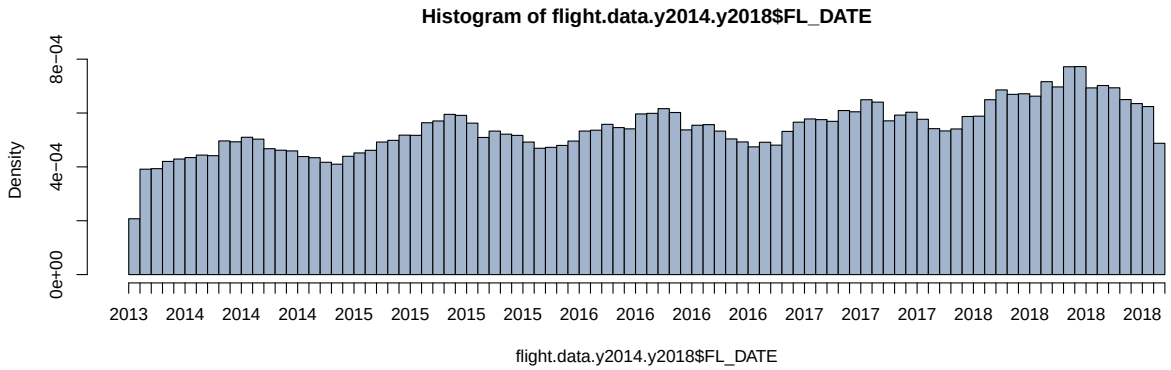
```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2) ## for charts
library(infer) ## for rep_sample_n used for clustered sampling
```

```
library(readr)
flight.data.y2014 <- read_csv("Data/2014.csv")
# head(flight.data.y2014)

flight.data.y2015 <- read_csv("Data/2015.csv")
# head(flight.data.y2015)

flight.data.y2016 <- read_csv("Data/2016.csv")
# head(flight.data.y2016)

flight.data.y2017 <- read_csv("Data/2017.csv")
# head(flight.data.y2017)

flight.data.y2018 <- read_csv("Data/2018.csv")
```

```r
# head(flight.data.y2018)

# Since this is a large dataset, sampling/manipualting on all the observations
# is throwing memory error in my machine. So for ease of processing, I am
# considering a subset of data with arrival station as MSP.

flight.data.y2014 <- flight.data.y2014[flight.data.y2014$DEST %in% 'MSP', ]
flight.data.y2015 <- flight.data.y2015[flight.data.y2015$DEST %in% 'MSP', ]
flight.data.y2016 <- flight.data.y2016[flight.data.y2016$DEST %in% 'MSP', ]
flight.data.y2017 <- flight.data.y2017[flight.data.y2017$DEST %in% 'MSP', ]
flight.data.y2018 <- flight.data.y2018[flight.data.y2018$DEST %in% 'MSP', ]
```

```r
# Combine the 5 vectors(for each files) to a single vector
flight.data.y2014.y2018 <-
  dplyr::bind_rows(flight.data.y2014,
                   flight.data.y2015,
                   flight.data.y2016,
                   flight.data.y2017,
                   flight.data.y2018,
                   )

# replace all na in valriables which we interested in to 0 for summary calculations.
flight.data.y2014.y2018$ARR_DELAY <-
  flight.data.y2014.y2018$ARR_DELAY %>%
  replace(is.na(.), 0)

flight.data.y2014.y2018$LATE_AIRCRAFT_DELAY <-
  flight.data.y2014.y2018$LATE_AIRCRAFT_DELAY %>%
  replace(is.na(.), 0)

flight.data.y2014.y2018$SECURITY_DELAY <-
  flight.data.y2014.y2018$SECURITY_DELAY %>%
  replace(is.na(.), 0)

flight.data.y2014.y2018$WEATHER_DELAY <-
  flight.data.y2014.y2018$WEATHER_DELAY %>%
  replace(is.na(.), 0)

flight.data.y2014.y2018$CARRIER_DELAY <-
  flight.data.y2014.y2018$CARRIER_DELAY %>%
  replace(is.na(.), 0)

hist(flight.data.y2014.y2018$FL_DATE,
     flight.data.y2014.y2018$ARR_DELAY,
     breaks = 100,
     col='lightsteelblue3')
```



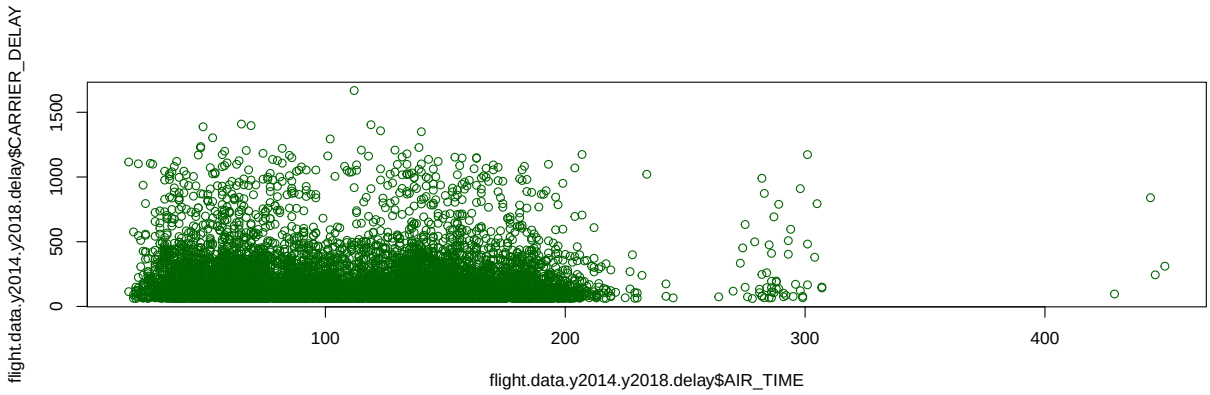

```
summary(flight.data.y2014.y2018$ARR_DELAY)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -119.000  -16.000   -7.000    1.781    4.000 1668.000
```

```
flight.data.y2014.y2018.delay <-
  flight.data.y2014.y2018[flight.data.y2014.y2018$CARRIER_DELAY > 60, ]

summary(flight.data.y2014.y2018.delay$CARRIER_DELAY)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    61.0    81.0   120.0   189.4   218.0  1668.0
```

```
plot(flight.data.y2014.y2018.delay$AIR_TIME,
     flight.data.y2014.y2018.delay$CARRIER_DELAY,
     col = "darkgreen")
```



Topics From Class

Topic 1:

R Markdown - I will be presenting the project in R Markdown and knit the file to a pdf document. Will be using R chunks to demonstrate and build the project components.

Topic 2:

GitHub - Will host the project in github repository for others to view my project components.

Topic 3:

Sampling strategies for an Observational study - Will be using sampling strategies - Simple random sampling, Strtified sampling, Cluster sampling and multistage sampling to group the data together by using different variables from the dataset and then use one of the sampling result to build topic#4 and 5.

```
simple.sampling <- dplyr::sample_n(flight.data.y2014.y2018, 1000, replace=FALSE)
# View(simple.sampling)
simple.sampling
```

```
## # A tibble: 1,000 x 28
##    FL_DATE    OP_CARRIER OP_CARRIER_FL_NUM ORIGIN DEST  CRS_DEP_TIME DEP_TIME
##    <date>     <chr>                  <dbl> <chr>  <chr>        <dbl>    <dbl>
##  1 2017-05-25 WN                       494 STL    MSP           2135     2345
##  2 2014-07-17 DL                      1763 DCA    MSP           1445     1442
##  3 2017-06-04 OO                      4548 PIT    MSP            930      925
##  4 2017-03-11 DL                      1514 PHX    MSP           1025     1020
##  5 2014-09-11 UA                       662 SFO    MSP           1234     1226
##  6 2014-06-07 DL                      2105 SFO    MSP           1520     1513
##  7 2015-02-05 WN                      4277 MKE    MSP           2025     2021
##  8 2018-05-25 AA                       329 DFW    MSP            725      949
##  9 2016-03-27 DL                       888 ATL    MSP           1930     2000
## 10 2018-12-09 DL                       806 SFO    MSP           1100     1058
## # ... with 990 more rows, and 21 more variables: DEP_DELAY <dbl>,
## #   TAXI_OUT <dbl>, WHEELS_OFF <dbl>, WHEELS_ON <dbl>, TAXI_IN <dbl>,
## #   CRS_ARR_TIME <dbl>, ARR_TIME <dbl>, ARR_DELAY <dbl>, CANCELLED <dbl>,
## #   CANCELLATION_CODE <chr>, DIVERTED <dbl>, CRS_ELAPSED_TIME <dbl>,
## #   ACTUAL_ELAPSED_TIME <dbl>, AIR_TIME <dbl>, DISTANCE <dbl>,
## #   CARRIER_DELAY <dbl>, WEATHER_DELAY <dbl>, NAS_DELAY <dbl>,
## #   SECURITY_DELAY <dbl>, LATE_AIRCRAFT_DELAY <dbl>, `Unnamed: 27` <lgl>
```

```
# Here I am making a cluster of where Airline code is the strata
DL <- flight.data.y2014.y2018[flight.data.y2014.y2018$OP_CARRIER %in% 'DL', ]
UA <- flight.data.y2014.y2018[flight.data.y2014.y2018$OP_CARRIER %in% 'UA',]
AA <- flight.data.y2014.y2018[flight.data.y2014.y2018$OP_CARRIER %in% 'AA',]
WN <- flight.data.y2014.y2018[flight.data.y2014.y2018$OP_CARRIER %in% 'WN',]

stratified.sampling <- dplyr::sample_n((UA), 1000, replace=FALSE)
dim(stratified.sampling)
```

```
## [1] 1000   28
```

```
#randomly choose 4 10 groups out of the n
clusters <-
  sample(unique(flight.data.y2014.y2018$OP_CARRIER), size=10, replace=FALSE)

#define sample as all members who belong to one of the 10 operated carriers
clustered_by_op_carrier <-
```

```
  flight.data.y2014.y2018[flight.data.y2014.y2018$OP_CARRIER %in% clusters, ]

#view how many observations came from each tour
table(clustered_by_op_carrier$OP_CARRIER)
```

```
##
##     9E     AA     AS     DL     EV     F9     FL     UA     US     YX
##  11576  33675   3583 315560  22864   7233   1111  16793   6319   5196
```

```
clustered_by_op_carrier
```

```
## # A tibble: 423,910 x 28
##    FL_DATE    OP_CARRIER OP_CARRIER_FL_NUM ORIGIN DEST  CRS_DEP_TIME DEP_TIME
##    <date>     <chr>                  <dbl> <chr>  <chr>        <dbl>    <dbl>
##  1 2014-01-01 EV                      4214 CLE    MSP           1220       NA
##  2 2014-01-01 EV                      4380 EWR    MSP            828      930
##  3 2014-01-01 EV                      4472 IAH    MSP           1156     1154
##  4 2014-01-01 EV                      4667 EWR    MSP           1400     1400
##  5 2014-01-01 EV                      5003 CLE    MSP           1200     1159
##  6 2014-01-01 EV                      5009 SYR    MSP            825      820
##  7 2014-01-01 EV                      4981 RIC    MSP            720      710
##  8 2014-01-01 EV                      4685 IAH    MSP           1917     1914
##  9 2014-01-01 EV                      5407 OMA    MSP           1712     1837
## 10 2014-01-01 EV                      5353 IND    MSP           1735     1742
## # ... with 423,900 more rows, and 21 more variables: DEP_DELAY <dbl>,
## #   TAXI_OUT <dbl>, WHEELS_OFF <dbl>, WHEELS_ON <dbl>, TAXI_IN <dbl>,
## #   CRS_ARR_TIME <dbl>, ARR_TIME <dbl>, ARR_DELAY <dbl>, CANCELLED <dbl>,
## #   CANCELLATION_CODE <chr>, DIVERTED <dbl>, CRS_ELAPSED_TIME <dbl>,
## #   ACTUAL_ELAPSED_TIME <dbl>, AIR_TIME <dbl>, DISTANCE <dbl>,
## #   CARRIER_DELAY <dbl>, WEATHER_DELAY <dbl>, NAS_DELAY <dbl>,
## #   SECURITY_DELAY <dbl>, LATE_AIRCRAFT_DELAY <dbl>, `Unnamed: 27` <lgl>
```

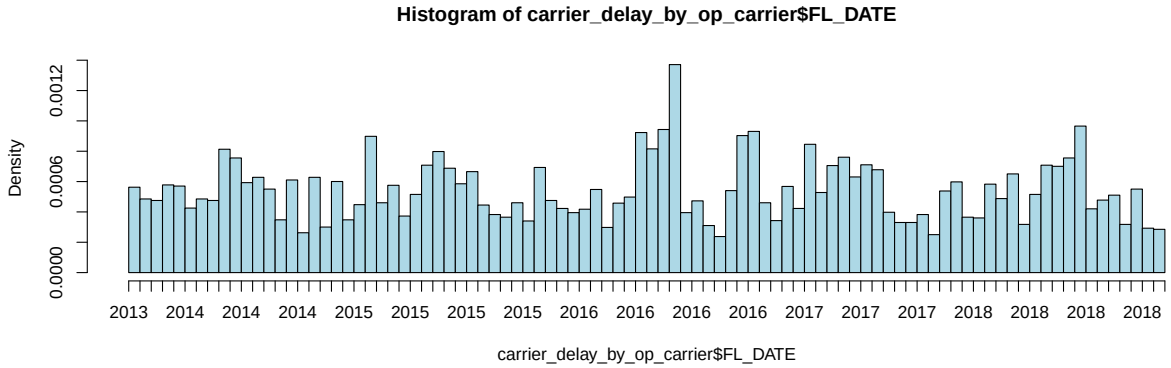
Topic 4:

Detailing Summary statistics (Min. , 1st Qu., Median, Mean, 3rd Qu., Max.) of a variable and plotting graphs using ggplot2

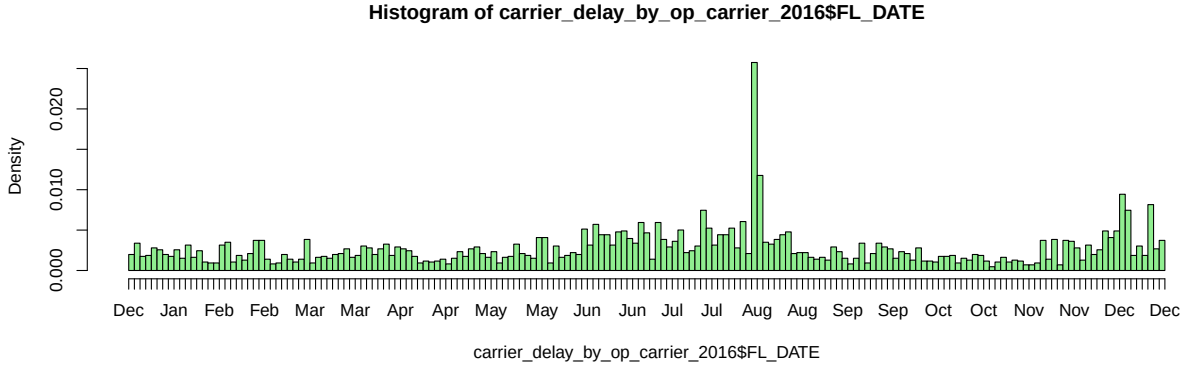
```
carrier_delay_by_op_carrier <-
  clustered_by_op_carrier %>%
  select(c(FL_DATE, OP_CARRIER, CARRIER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%  filter(CARRIER_DELAY > 0)

summary(carrier_delay_by_op_carrier$CARRIER_DELAY)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    7.00   18.00   56.36   48.00 1199.00
```

```
hist(carrier_delay_by_op_carrier$FL_DATE,
     carrier_delay_by_op_carrier$CARRIER_DELAY,
     breaks = 100,
     col = "lightblue")
```

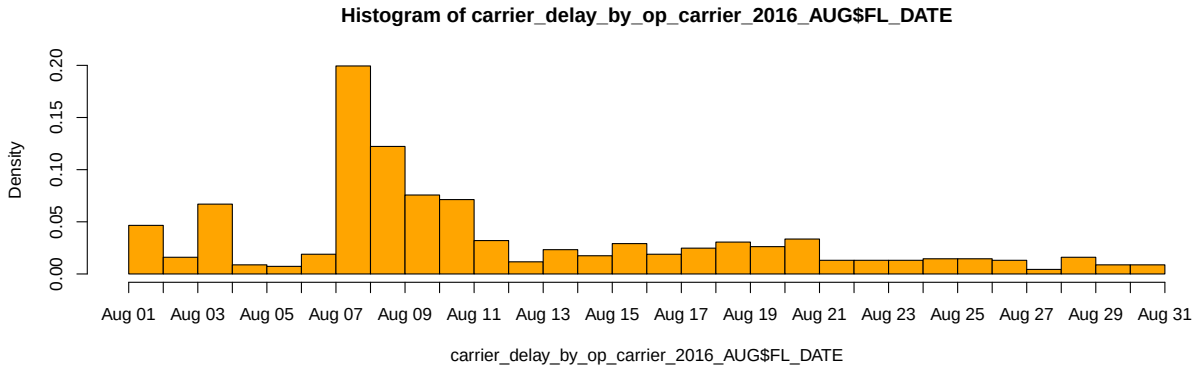



```
carrier_delay_by_op_carrier_2016 <-
  carrier_delay_by_op_carrier

carrier_delay_by_op_carrier_2016$YEAR <-
  format(carrier_delay_by_op_carrier$FL_DATE, "%Y")

carrier_delay_by_op_carrier_2016 <-
  carrier_delay_by_op_carrier_2016 %>%
  filter(YEAR == '2016')

carrier_delay_by_op_carrier_2016$MONTH <-
  format(carrier_delay_by_op_carrier_2016$FL_DATE, "%m")

hist(carrier_delay_by_op_carrier_2016$FL_DATE,
     carrier_delay_by_op_carrier_2016$CARRIER_DELAY,
     breaks =200,
     col = "lightgreen")
```




```
# From the above plot it has been determined that August2016 has an increase in carrier delay. The foll
carrier_delay_by_op_carrier_2016_AUG <-
  carrier_delay_by_op_carrier_2016

carrier_delay_by_op_carrier_2016_AUG <-
  carrier_delay_by_op_carrier_2016_AUG %>%
  filter(MONTH == '08')
```

```
carrier_delay_by_op_carrier_2016_AUG$DAY <-
  format(carrier_delay_by_op_carrier_2016_AUG$FL_DATE,
         "%d")

hist(carrier_delay_by_op_carrier_2016_AUG$FL_DATE,
     carrier_delay_by_op_carrier_2016_AUG$CARRIER_DELAY,
     breaks =40,
     col = "orange")
```



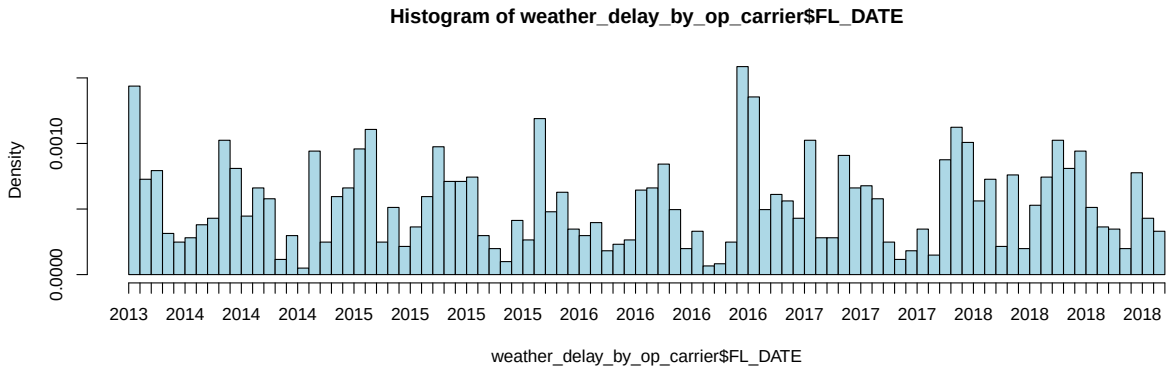

```
summary(carrier_delay_by_op_carrier_2016_AUG$CARRIER_DELAY)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    9.00   22.00   70.74   76.00 1167.00
```

From the news during those days there was a system outage which caused this massive delay/cancellations. Delta System Outage - Aug2016

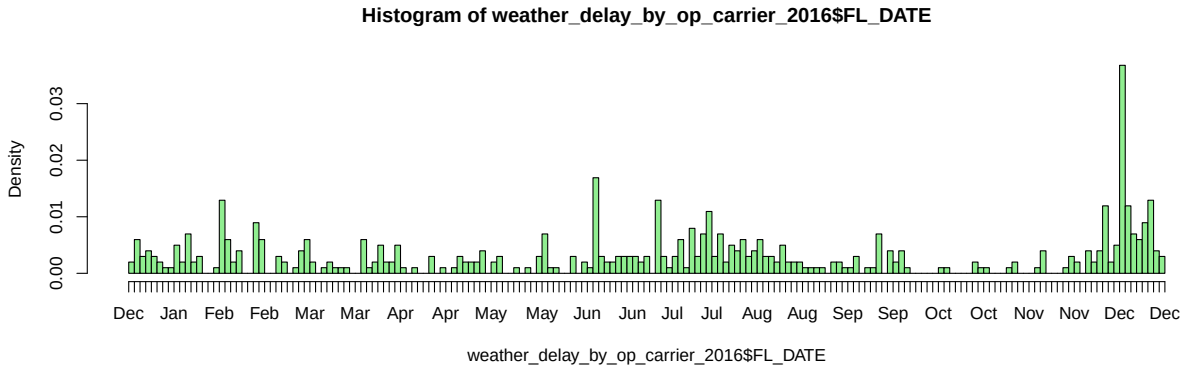
WEATHER DELAY

```
# First select the sub-vectors which contains only he coulums we are interested in
weather_delay_by_op_carrier <-
  clustered_by_op_carrier %>%
  select(c(FL_DATE,
           OP_CARRIER,
           WEATHER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%
  filter(WEATHER_DELAY > 0
         )

hist(weather_delay_by_op_carrier$FL_DATE,
     weather_delay_by_op_carrier$WEATHER_DELAY,
     breaks = 100,
     col = "lightblue")
```




```
# In 2016 we have a increase in delay. Let's find out month by analysis to approximate on which month it
weather_delay_by_op_carrier_2016 <-
  weather_delay_by_op_carrier

weather_delay_by_op_carrier_2016$YEAR <-
  format(weather_delay_by_op_carrier$FL_DATE, "%Y")

weather_delay_by_op_carrier_2016 <-
  weather_delay_by_op_carrier_2016 %>%
  filter(YEAR == '2016')

weather_delay_by_op_carrier_2016$MONTH <-
  format(weather_delay_by_op_carrier_2016$FL_DATE, "%m")

hist(weather_delay_by_op_carrier_2016$FL_DATE,
     weather_delay_by_op_carrier_2016$weather_delay,
     breaks =200, col = "lightgreen")
```

```
## Warning: Unknown or uninitialised column: `weather_delay`.
```



The above histogram shows that whether delays are massive in December month in MSP airport. This could be explained by winter storms related delays.

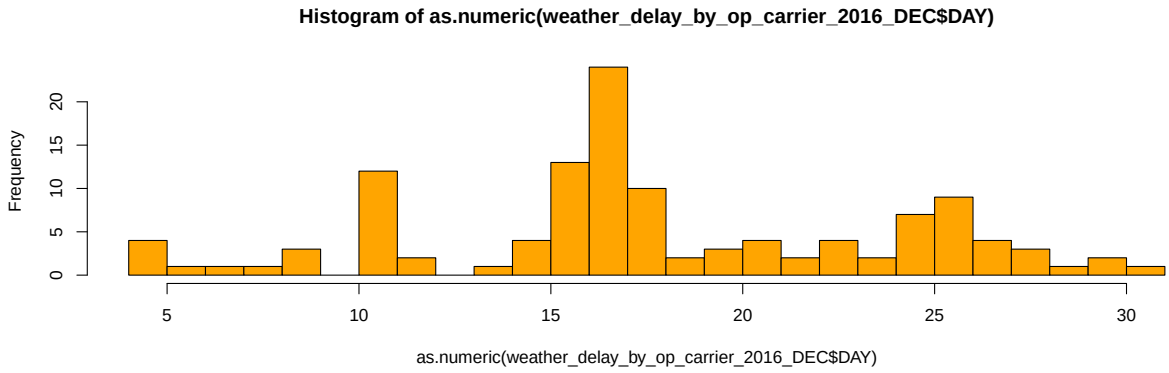
```
weather_delay_by_op_carrier_2016
```

```
## # A tibble: 503 x 5
```

```
##    FL_DATE    OP_CARRIER WEATHER_DELAY YEAR  MONTH
##    <date>     <chr>              <dbl> <chr> <chr>
##  1 2016-01-01 DL                     8 2016  01
##  2 2016-01-02 DL                     4 2016  01
##  3 2016-01-03 DL                    20 2016  01
##  4 2016-01-03 DL                    17 2016  01
##  5 2016-01-03 DL                    59 2016  01
##  6 2016-01-03 DL                    12 2016  01
##  7 2016-01-03 DL                   108 2016  01
##  8 2016-01-04 DL                    28 2016  01
##  9 2016-01-06 DL                    33 2016  01
## 10 2016-01-06 DL                    23 2016  01
## # ... with 493 more rows
```

```
weather_delay_by_op_carrier_2016_DEC <-
  weather_delay_by_op_carrier_2016 %>%
  filter(MONTH == '12')

weather_delay_by_op_carrier_2016_DEC$DAY <-
  format(weather_delay_by_op_carrier_2016_DEC$FL_DATE, "%d")

# weather_delay_by_op_carrier_2016_DEC

hist(as.numeric(weather_delay_by_op_carrier_2016_DEC$DAY),
     weather_delay_by_op_carrier_2016_DEC$weather_delay,
     breaks =30, col = "orange")
```

```
## Warning: Unknown or uninitialised column: 'weather_delay'.
```

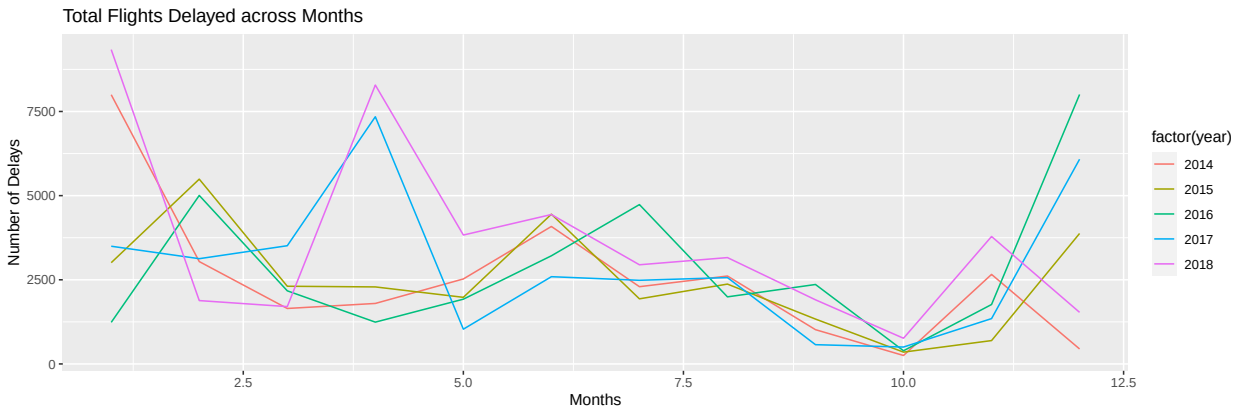



```
summary(weather_delay_by_op_carrier_2016_DEC$weather_delay)
```

```
## Warning: Unknown or uninitialised column: 'weather_delay'.

## Length  Class   Mode
##      0   NULL   NULL
```

```
weather_delay_by_op_carrier_2014.gorupby <-
  flight.data.y2014 %>%
  select(c(FL_DATE, OP_CARRIER, WEATHER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%
  filter(WEATHER_DELAY > 0) %>%
  group_by(as.numeric(format(FL_DATE, "%m"))) %>%
  summarize(total_delayed=sum(WEATHER_DELAY)) %>%
  mutate(year=2014)

weather_delay_by_op_carrier_2015.gorupby <-
  flight.data.y2015 %>%
  select(c(FL_DATE, OP_CARRIER, WEATHER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%
  filter(WEATHER_DELAY > 0) %>%
  group_by(as.numeric(format(FL_DATE, "%m"))) %>%
  summarize(total_delayed=sum(WEATHER_DELAY)) %>%
  mutate(year=2015)

weather_delay_by_op_carrier_2016.gorupby <-
  flight.data.y2016 %>%
  select(c(FL_DATE, OP_CARRIER, WEATHER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%
  filter(WEATHER_DELAY > 0) %>%
  group_by(as.numeric(format(FL_DATE, "%m"))) %>%
  summarize(total_delayed=sum(WEATHER_DELAY)) %>%
  mutate(year=2016)

weather_delay_by_op_carrier_2017.gorupby <-
  flight.data.y2017 %>%
  select(c(FL_DATE, OP_CARRIER, WEATHER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%
  filter(WEATHER_DELAY > 0) %>%
  group_by(as.numeric(format(FL_DATE, "%m"))) %>%
  summarize(total_delayed=sum(WEATHER_DELAY)) %>%
  mutate(year=2017)

weather_delay_by_op_carrier_2018.gorupby <-
  flight.data.y2018 %>%
  select(c(FL_DATE, OP_CARRIER, WEATHER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%
  filter(WEATHER_DELAY > 0) %>%
  group_by(as.numeric(format(FL_DATE, "%m"))) %>%
  summarize(total_delayed=sum(WEATHER_DELAY)) %>%
  mutate(year=2018)

month_Delay<-rbind(weather_delay_by_op_carrier_2014.gorupby,
                   weather_delay_by_op_carrier_2015.gorupby,
                   weather_delay_by_op_carrier_2016.gorupby,
                   weather_delay_by_op_carrier_2017.gorupby,
                   weather_delay_by_op_carrier_2018.gorupby)

ggplot(month_Delay,
       aes(x = `as.numeric(format(FL_DATE, "%m"))`,
```

```
          y = total_delayed,
          color = factor(year), group = factor(year))) +
geom_line(linetype = 1) +
  labs(title="Total Flights Delayed across Months",y = 'Number of Delays',x = 'Months', fill='YEAR')
```



The above plot shows weather delays during 2014-2018 and it's clear that most of the weather related delays happens during year start/end.

Topic 5:

Regression (if an increase in number of schedules has any impact/variace on carrier delays).

```
flight.data.y2018
```

```
## # A tibble: 159,365 x 28
##    FL_DATE    OP_CARRIER OP_CARRIER_FL_NUM ORIGIN DEST  CRS_DEP_TIME DEP_TIME
##    <date>     <chr>                  <dbl> <chr>  <chr>        <dbl>    <dbl>
##  1 2018-01-01 UA                      2118 DEN    MSP           1245     1239
##  2 2018-01-01 UA                      1728 SFO    MSP           2320     2319
##  3 2018-01-01 UA                       878 IAH    MSP           1955     2032
##  4 2018-01-01 UA                       774 ORD    MSP           2245     2244
##  5 2018-01-01 UA                       669 DEN    MSP           2027     2026
##  6 2018-01-01 UA                       573 DEN    MSP            945      944
##  7 2018-01-01 UA                       215 DEN    MSP            756      746
##  8 2018-01-01 AS                        28 SEA    MSP           1750     1748
##  9 2018-01-01 AS                        36 SEA    MSP           1000      951
## 10 2018-01-01 9E                      3615 GFK    MSP           1310     1302
## # ... with 159,355 more rows, and 21 more variables: DEP_DELAY <dbl>,
## #   TAXI_OUT <dbl>, WHEELS_OFF <dbl>, WHEELS_ON <dbl>, TAXI_IN <dbl>,
## #   CRS_ARR_TIME <dbl>, ARR_TIME <dbl>, ARR_DELAY <dbl>, CANCELLED <dbl>,
## #   CANCELLATION_CODE <chr>, DIVERTED <dbl>, CRS_ELAPSED_TIME <dbl>,
## #   ACTUAL_ELAPSED_TIME <dbl>, AIR_TIME <dbl>, DISTANCE <dbl>,
## #   CARRIER_DELAY <dbl>, WEATHER_DELAY <dbl>, NAS_DELAY <dbl>,
## #   SECURITY_DELAY <dbl>, LATE_AIRCRAFT_DELAY <dbl>, `Unnamed: 27` <lgl>
```

```
carrier_delay_by_op_carrier_2018.totalFlights <-
  flight.data.y2018 %>%
  select(c(FL_DATE, OP_CARRIER, CARRIER_DELAY)) %>%
```

```
  filter(OP_CARRIER == 'DL') %>%
  group_by(FL_DATE) %>% count() %>% mutate(year=2018)

carrier_delay_by_op_carrier_2018.carrierDelay <-
  flight.data.y2018 %>%
  select(c(FL_DATE, OP_CARRIER, CARRIER_DELAY)) %>%
  filter(OP_CARRIER == 'DL') %>%
  filter(CARRIER_DELAY >= 0) %>%
  group_by(FL_DATE) %>%
  summarize(total_delayed=sum(CARRIER_DELAY > 0)) %>%
  mutate(year=2018)

dataset <- bind_cols(carrier_delay_by_op_carrier_2018.totalFlights,
                     carrier_delay_by_op_carrier_2018.carrierDelay)
```

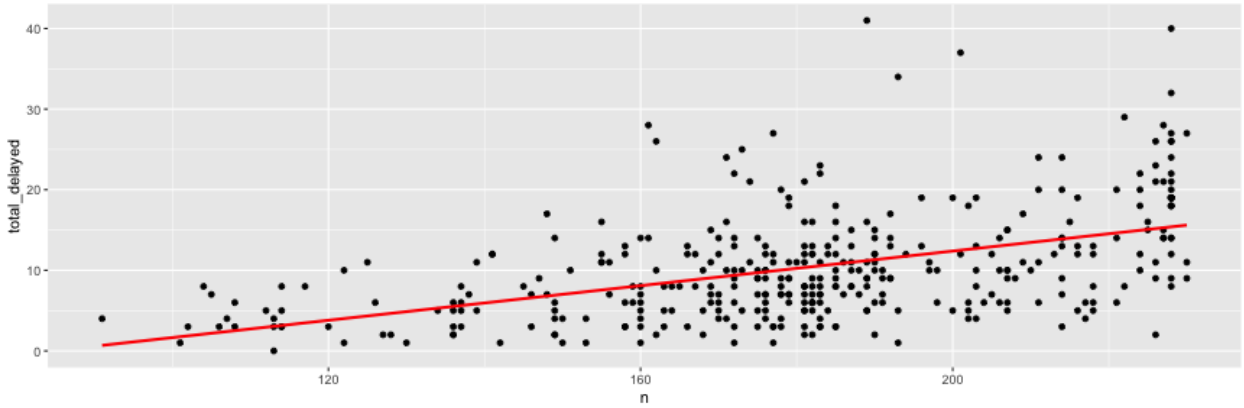
```
## New names:
## * `FL_DATE` -> `FL_DATE...1`
## * `year` -> `year...3`
## * `FL_DATE` -> `FL_DATE...4`
## * `year` -> `year...6`
```

```
linear_model <- lm(total_delayed ~ n,
                   data=dataset)
summary(linear_model)
```

```
##
## Call:
## lm(formula = total_delayed ~ n, data = dataset)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.1745  -3.8807  -0.8355   3.1588  29.7973
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -9.08524    1.89497  -4.794 2.38e-06 ***
## n            0.10734    0.01031  10.412  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.017 on 363 degrees of freedom
## Multiple R-squared:   0.23,  Adjusted R-squared:  0.2278
## F-statistic: 108.4 on 1 and 363 DF,  p-value: < 2.2e-16
```

```
ggplot(dataset, aes(x=n,
                    y=total_delayed)) +
  geom_point() +
  geom_smooth(method='lm', se=FALSE, col="red", size=1)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



There appears to be have a linear relation between carrier delay and total flights. Since there is a linear relation between number of flights arrived and carrier delay, it could be due to airline related issue(like crew/pilot scheduling issue or some other operational issues.)

Additional topic: Access to aws s3 bucket in RMarkdown. As a fun expiriment I am trying to store the generated file or plots by a chunk into aws s3. Still this work is in progress

```
library("aws.s3")
Sys.setenv(
  "AWS_ACCESS_KEY_ID" = "AKIAUTK5NLVJF67UNMH5",
  "AWS_SECRET_ACCESS_KEY" = "",
  "AWS_DEFAULT_REGION" = "us-east-1"
)
```

```
bucketlist()
```

```
## data frame with 0 columns and 0 rows
```

```
tempdir()
```

```
## [1] "/var/folders/mh/rsqncfcs2lj5vxkprghf86d00000gn/T//Rtmps5uZYf"
```

```
tempfile()
```

```
## [1] "/var/folders/mh/rsqncfcs2lj5vxkprghf86d00000gn/T//Rtmps5uZYf/file66f73a8033c"
```

```
write.csv('writeup.pdf', file.path(tempdir(),"writeup.pdf"))
```

```
# put_object(
#   file = file.path(tempdir(), writeup.pdf),
#   object = 'writeup.pdf',
#   bucket = 'seis631-finalproject'
# )
```

Conclusion

I designed this project as a way to review some of the topics we learned in the class/homework/assignments to reinforce some topics learned and also as an opportunity to refer back some of the materials. Hence I thought of picking a variety of topics like sampling strategies, summary statistics, ANOVA and regressions will be the best approach and most I can get from this project. If I have more time, I would have included some more topics (like binom, dbinom, geom...etc distributions) and see if my dataset have variables that can fit these distributions. Given only a academic background in statistics almost almost 20 years ago, I think this subject has given me much learning experience in statistics and I appreciate how these topics are applicable to find solutions in reality.