NYC_Project The Flights Dataset

Ter oenW el

Ner yev} 5; 0645=

- Q29. Linear Regression Analysis between two major airliness.
- Q30. Relationship of Time with 3 other dependent variables.
- Q31. Dive deep into three specific Airliness.
- Q32. Summaries within airports from chosen three flights.
- Q33. Overall Share in Mosiac plot
- Q34. Relationship between Weather and Departure delays.
- Q35. Trend in mean departure delay by visibility
- Q36. Relationship between Pressure and Departure delays
- Q37. Speeds by carrier
- Q39. United's delays by NYC airport.
- Q40. Departure delays by month
- Q41. How many seats are on a plane?
- Q42. Ovearll Stastical delays by month
- Q43. On time departure rate for NYC airports
- Q44. Overall Delayed and Ontime flights
- Q46 How many planes were late than 5 Minutes.
- Q47. Analysing on Single Plane
- Q48. Looking other components of Weather on flight Datasets.
- Take Home Message:

Group 2 Project:

Instructor: LESSIA SHAJENKO

ALY 6040- 71233 DATA MINING APPLICATIONS FALL 2018 CPS PROPOSAL DRAFT TEAM: FLIGHT ANALYST:

Northeastern University

Main Outcome of this notebook:

By following the steps and running this notebook in your laptop, you will be able to understand and explore flights that were departed from the three major New York City airport in 2013. We will also generate and do graphical and statistical analysis along the ways to address why were their delays in the flight. If you follow the code and knit this file in pdf you will be able to learn the indispensable skills of data processing and subsetting. So happy Learning and exploring in the built datasets.

Introduction:

In our lifetime we likely have all flown on airplanes once or many times. Or we might have known someone who has flown the plane, not a pilot but a traveler. One thing we hear from most of the travelers is about the planes getting delayed or canceled because of many factors besides weather and mechanical conditions. We also have witnessed many times in news the plane making Emergency Landing because of Failed Mechanical Components. We have also encountered frequently that some flights are delayed because of a variety of conditions from certain airports. As a data anlayst, we always try to understand if there are any ways we could avoid having to deal with these flight delays. If we have data available could we make sense of all these delays or a certain time of year we should avoid trying to fly? Throughout this project, I am going to analyze data related to flights which are contained in the nycflights13 package (Wickham 2018). It is easy to download in R studio and not hard to find.

Specifically, when we download into R workspace from the library this package contains five datasets saved as “data frames”. The nycflighsts datasets contain the information mostly about all domestic flights departing from Big Apple(New York City) in the year 2013. From the datasets, we can get the information about the airports.

There are three airports.

- **Newark Liberty International (EWR)**
- **John F. Kennedy International (JFK)**
- **LaGuardia (LGA) Airports**

The **flights** data frame is the main dataset in the package not only contains detailed information for all the flights that had departed from NYC in the year 2013 but also information about **airlines**,**airports** and **weather**. It is pretty good datasets to analyze and understand a little bit about the Aviation industry and datasets is real world example itself from New York.

New York City ranks just above the middle of the pack for both average snowfall (23 inches per snow season) and days with measurable snow in a year (14 days). Despite that, snowstorms can cripple air travel for a full day or longer at this airport, as well as the major airports nearby. Snowstorms aren’t the only factors that trigger delays. Frontal systems can also cause delays when they’re accompanied by low clouds.Of the New York area’s three major airports, JFK saw the fewest arrivals in the time period we studied, but there were more than enough delays to vault this hub into our top 10.

Information about the nycflights package:

The package was created on June 22th, 2014, by Hadley Wickam.(Remember that name). Every data Analyst or Data Scientist who works in R knows him. For more details regarding the package, please refer to the nycflights13 introduction page (http://cran.r-project.org/web/packages/nycflights13/nycflights13.pdf).

Goals and Outcomes of this project:

We will together explore the data sets in the ‘nycflights13’ package to find out about the flight delays, cancellation, trends, insights and uncover interesting anomalies regarding traffic volume across airline carriers as well as the three different airports. The datasets are easier to understand and a lot of things can be done using the datasets. So I am excited about the project. Let’s dive deeper and explore the unseen journey of nycflights13.

Datasets Description

Info about Datasets

- flights: information on all `dim` dimension flights.
- airlines: translation between two letter IATA carrier codes and names (16 in total)
- planes: construction information about each of 3,322 planes used.
- weather: hourly meteorological data (about 8705 observations) for each of the three NYC airports
- airports: airport names and locations.

Sourceof the original data: RITA, Bureau of transportation statistics

• http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236
(http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236)

Lets first start to check all the packages that we need to run and analyse this whole workbook.

```
webshot::install_phantomjs() #this following code helps to convert it in pdf document ni
cely
```

```
## phantomjs has been installed to /Users/pankajshah/Library/Application Support/Phantom
JS
```

These is the session info of these R codes. Check for the specific package and the library version if you run into any of the issues.

```
sessionInfo()
```

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Sierra 10.12
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## loaded via a namespace (and not attached):
##  [1] compiler_3.5.0  backports_1.1.2 magrittr_1.5    rprojroot_1.3-2
##  [5] tools_3.5.0     htmltools_0.3.6 yaml_2.2.0      Rcpp_1.0.0
##  [9] stringi_1.2.4   rmarkdown_1.10  knitr_1.20      webshot_0.5.1
## [13] stringr_1.3.1   digest_0.6.18   evaluate_0.12
```

Check for the specific packages in your specific Laptop.

```
if(!require(dplyr)) install.packages("dplyr")
```

```
## Loading required package: dplyr
```

```
##
## 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
```

```
if(!require(ggplot2)) install.packages("ggplot2")
```

```
## Loading required package: ggplot2
```

```
if(!require(nycflights13)) install.packages("nycflights13")
```

```
## Loading required package: nycflights13
```

```
if(!require(grid)) install.packages("grid")
```

```
## Loading required package: grid
```

```
if(!require(vcd)) install.packages("vcd")
```

```
## Loading required package: vcd
```

```
if(!require(readr)) install.packages("readr")
```

```
## Loading required package: readr
```

```
if(!require(moderndive)) install.packages("moderndive")
```

```
## Loading required package: moderndive
```

```
if(!require(kableExtra)) install.packages("kableExtra")
```

```
## Loading required package: kableExtra
```

```
if(!require(lubridate)) install.packages("lubridate")
```

```
## Loading required package: lubridate
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
today <- Sys.Date()
today
```

```
## [1] "2019-01-17"
```

Library

Load helpful necessay packages.

```
library(ggplot2)
library(dplyr)
library(nycflights13) # Our datasets is coming from these library
library(pander)
library(grid)
library(vcd)
library(readr)
library(moderndive)
library(kableExtra)
library(statsr)
library(lubridate)
library(geometry)
library(rticles)
library(revealjs)
```

Dimensions

The following code will display the dimension of our datasets as it is very important to know how big your data is even before we dive in. Each observation (row) that we see below in the **flights** data frame represents a separate flight originated from one of the three airports from the New York City in the year 2013.

Structure of Flights datasets

```
str(flights)
```

```
## 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" ...
```

Data Dictionary

From the above outputs. Let's understand the column names which will help us to diagnose the data much easier. The gsp yq r req i wand gsp yq r hi wgvmhxsr ware given below:

First Three column names

- **year, month, day** - Date of departure (DATETIME)
- **dep_time** - Departure time, in minutes mrpsgepxmqi ~sri
- **sched_dep_time** - Scheduled Departure time, in minutes mrpsgepxmqi ~sri
- **dep_delay** - Departure delays, in minutes Rikexmzi xmqiwvitviwrxievw hitevxyiw8ewzepw
- **arr_time** - Arrival Time, in minutes Xliwievi mrpsgepxmqi ~sri
- **sched_arr_time** - Scheduled Arrival, in minutes time mrpsgepxmqi ~sri
- **arr_delay** - Arrival delays, in minutes. Rikexmzi xmqiwvitviwrxievw ewzepw
- **hour, minute** - Time of departure broken in to hour and minutes
- **carrier** - Two letter carrier abbreviation. **For more detail refer to airlines datasets to get the names**
- **tailnum** - Plane tail number (Every plane has its unique TailNum) Xlmoewe Prgiwgi Tpexi Ryqfivsjiegl zilrgpo
- **flight** - Flight number (Unique Number for each flight)
- **origin** - Origin (One Aiprot out of Three) . See **airports** data set for additional metadata
- **dest** - Final Destination
- **air_time** - Amount of time Plane has spent in the air. Starting from origin to reach the final Destination.
- **distance** - Distance travelled

An example of the observations is shown as follows:
Lets look at first observation of the data.First observation that has been recorded in our datasets.

Information about the datasets.

```
df_info <- function(x) {
  data  <- as.character(substitute(x))  ##data frame name
  size <- format(object.size(x), units="Mb")  ##size of data frame in Mb

  ##column information
  column.info <- data.frame( column        = names(sapply(x, class)),
                             #class         = sapply(x, class),
                             unique.values = sapply(x, function(y) length(unique(y))),
                             missing.count = colSums(is.na(x)),
                             missing.pct   = round(colSums(is.na(x)) / nrow(x) * 100, 2
))

  row.names(column.info) <- 1:nrow(column.info)

  list(data.frame     = data.frame(name=data, size=size),
       dimensions     = data.frame(rows=nrow(x), columns=ncol(x)),
       column.details = column.info)
}
Sys.timezone() # Will Display Time zone of your zone
```

```
## [1] "America/New_York"
```

Information about the Airlines

```
df_info(airlines)
```

```
## $data.frame
##       name size
## 1 airlines 0 Mb
##
## $dimensions
##   rows columns
## 1   16       2
##
## $column.details
##    column unique.values missing.count missing.pct
## 1 carrier            16             0           0
## 2    name            16             0           0
```

Information about the Weather

```
df_info(weather)
```

```
## $data.frame
##      name   size
## 1 weather 2.8 Mb
##
## $dimensions
##    rows columns
## 1 26115      15
##
## $column.details
##        column unique.values missing.count missing.pct
## 1      origin             3             0        0.00
## 2        year             1             0        0.00
## 3       month            12             0        0.00
## 4         day            31             0        0.00
## 5        hour            24             0        0.00
## 6        temp           174             1        0.00
## 7        dewp           154             1        0.00
## 8       humid          2500             1        0.00
## 9    wind_dir            38           460        1.76
## 10 wind_speed            37             4        0.02
## 11  wind_gust            38         20778       79.56
## 12     precip            59             0        0.00
## 13   pressure           469          2729       10.45
## 14      visib            20             0        0.00
## 15  time_hour          8714             0        0.00
```

Information about the Airports

```
df_info(airports)
```

```
## $data.frame
##       name   size
## 1 airports 0.3 Mb
##
## $dimensions
##   rows columns
## 1 1458       8
##
## $column.details
##   column unique.values missing.count missing.pct
## 1    faa          1458             0           0
## 2   name          1440             0           0
## 3    lat          1456             0           0
## 4    lon          1458             0           0
## 5    alt           911             0           0
## 6     tz             7             0           0
## 7    dst             3             0           0
## 8  tzone            10             0           0
```

Information about the flights

```
df_info(flights)
```

```
## $data.frame
##      name    size
## 1 flights 38.8 Mb
##
## $dimensions
##     rows columns
## 1 336776      19
##
## $column.details
##            column unique.values missing.count missing.pct
## 1            year             1             0        0.00
## 2           month            12             0        0.00
## 3             day            31             0        0.00
## 4        dep_time          1319          8255        2.45
## 5  sched_dep_time          1021             0        0.00
## 6       dep_delay           528          8255        2.45
## 7        arr_time          1412          8713        2.59
## 8  sched_arr_time          1163             0        0.00
## 9       arr_delay           578          9430        2.80
## 10        carrier            16             0        0.00
## 11         flight          3844             0        0.00
## 12        tailnum          4044          2512        0.75
## 13         origin             3             0        0.00
## 14           dest           105             0        0.00
## 15       air_time           510          9430        2.80
## 16       distance           214             0        0.00
## 17           hour            20             0        0.00
## 18         minute            60             0        0.00
## 19      time_hour          6936             0        0.00
```

Head of Flights datasets

```
head(flights, 1) %>% table()
```

```
## , , day = 1, dep_time = 517, sched_dep_time = 515, dep_delay = 2, arr_time = 830, sch
ed_arr_time = 819, arr_delay = 11, carrier = UA, flight = 1545, tailnum = N14228, origin
= EWR, dest = IAH, air_time = 227, distance = 1400, hour = 5, minute = 15, time_hour = 2
013-01-01 05:00:00
##
##       month
## year   1
##   2013 1
```

Interpretation:

- year : 2013
- month : 1 (JAN)
- day : 1st
- dep_time : 5:17 AM(EST)
- sched_dep_time : 5:15 AM (EST)

- dep_delay : -2 (Negative 2 Min i.e 5:15AM - 5:17 AM) Departed early before Scheduled.
- arr_time : 8:30 AM
- scheduled_air_time : 8:19 AM
- arr_delay : 11 Minutes (8:19- 8:30 AM)
- carrier : UA
- flight : 1545
- tailNum : 14228
- Origin : EWR(Newark Liberty International)
- dest : IAH
- air_time : 227 Minutes i.e 2 HR 27 Minutes. (8:30 - 5:17)
- distance: 1400 Miles
- HR & Min: 5 Hr 15 Minutes.

WORD ANALYSIS

- For Example The flight # 1545 is The United Airlines flight (tail NUM: 14228) which originated from Newark Liberty International(EWR) airport in New York, NY flew to George Bush Intercontinental Airport (IAH) in Houston Texas on January 1, 2013 departing at 5:17 am (EST) and arriving at 8:30 AM (CST). It flew a distance of 1400 miles for 2 hours and 27 minutes, spending in the air minutes and leaving EWR airport earlier than the scheduled time by 2 minutes. Arrived at Houston Airport 11 Minutes in delay than scheduled flight.

Flights Dataset preparation:

Q1.TOP 10 carriers from all three airports.

```
flights_table <- flights %>%
                   group_by(carrier) %>%
                      summarise(number = n()) %>%
                      arrange(desc(number)) %>% top_n(10)
```

```
## Selecting by number
```

```
flights_table$carrier <- as.factor(flights_table$carrier)
flights_table$number <- as.numeric(flights_table$number)
flights_table
```

carrier <fctr>	number <dbl>
UA	58665
B6	54635
EV	54173
DL	48110
AA	32729

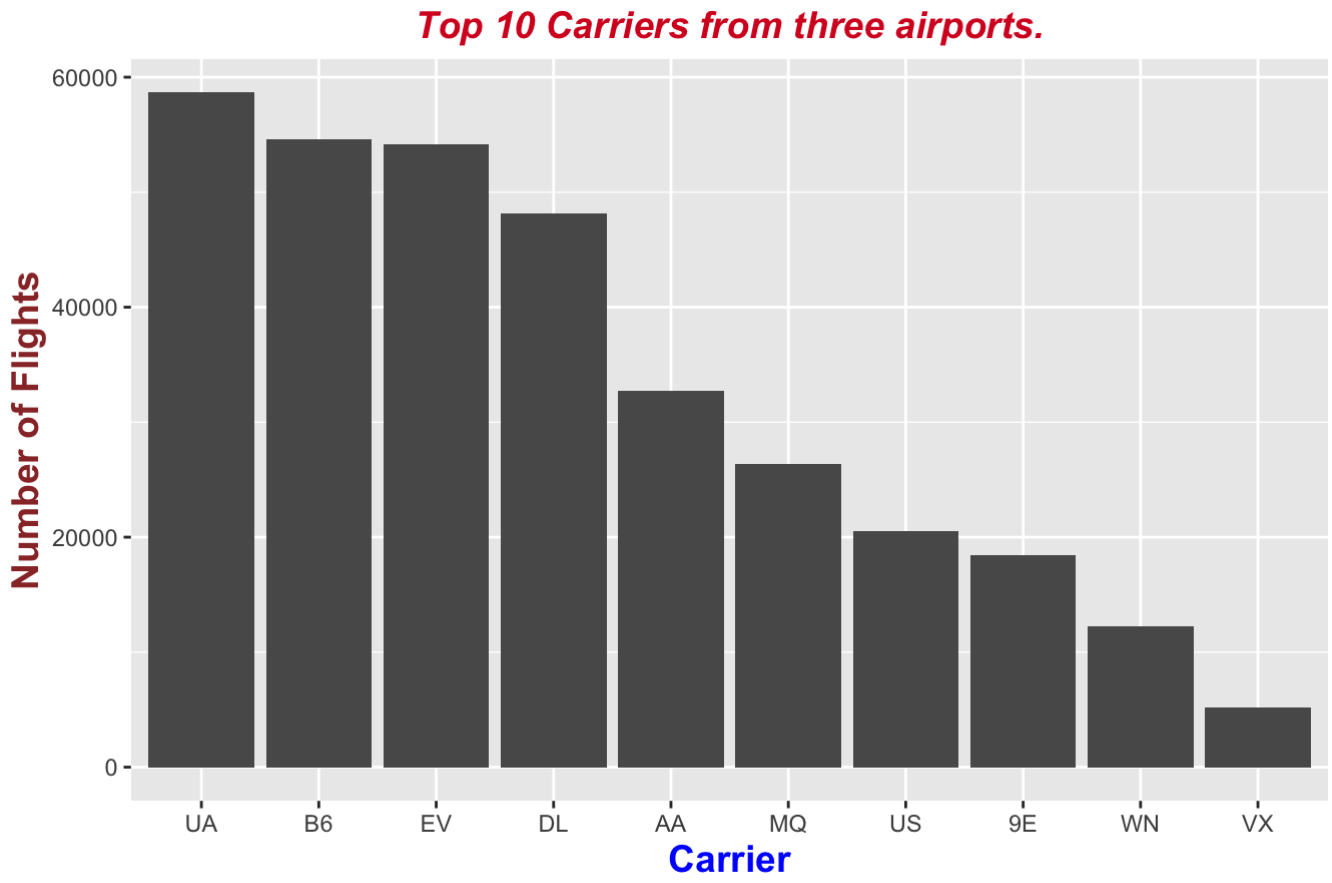
carrier <fctr>	number <dbl>
MQ	26397
US	20536
9E	18460
WN	12275
VX	5162

1-10 of 10 rows

ANALYSIS

We see that UA flew most from all three airports [58665]. Below we will plot in descending order.

```
ggplot(flights_table, aes(x=reorder(carrier,-number), y=number)) +
  geom_bar(stat="identity") +
  labs(x="Carrier", y="Number of Flights")+
  ggtitle("Top 10 Carriers from three airports.") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
    plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#1E1E20"
,face="bold.italic" ),
    plot.title = element_text(color="#D70026", size=14, face="bold.italic", hjust = 0.5
),
    axis.title.x = element_text(color="blue", size=14, face="bold"),
    axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

ANALYSIS

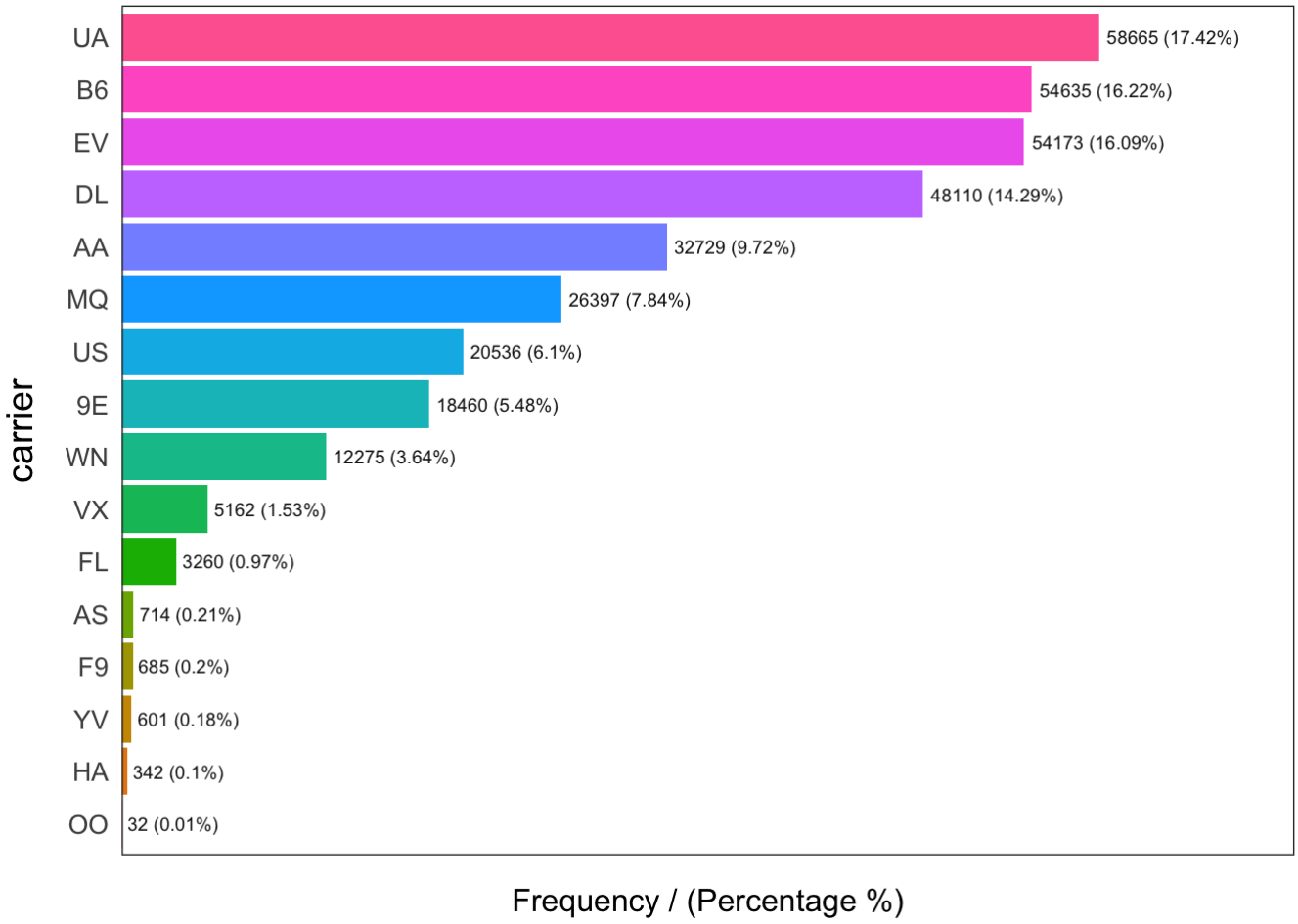
From the above barplot, we can conclude that United Airways(UA) has done more flights from all three airports combined. Followed by Jet Blue (B6) & (EV).

Basic Data Exploratory

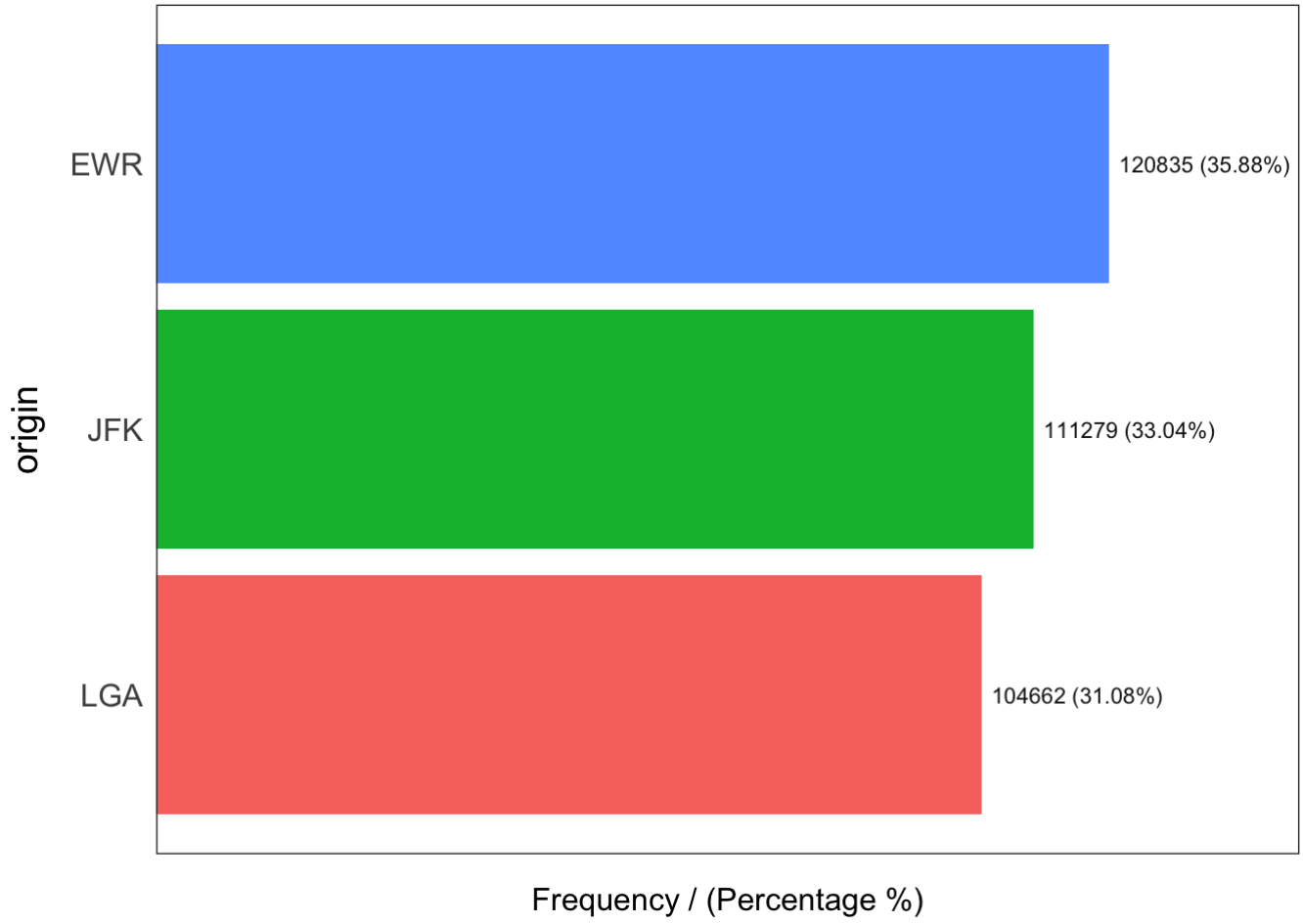
Lets do some basic data Exploratory Analysis before we dive into detail.

```
basic_eda <- function(data)
{
  library(Hmisc)
  library(funModeling)
  library(tidyverse)

  #profiling_num(data)
  #plot_num(data)
  #describe(data)
  #df_status(data)
  #glimpse(data)
  freq(data)
  sapply(data, function(x) sum(is.na(x)))
}
```

```
options(max.print = 100)
basic_eda(flights)
```



```
##    carrier frequency percentage cumulative_perc
## 1       UA     58665      17.42           17.42
## 2       B6     54635      16.22           33.64
## 3       EV     54173      16.09           49.73
## 4       DL     48110      14.29           64.02
## 5       AA     32729       9.72           73.74
## 6       MQ     26397       7.84           81.58
## 7       US     20536       6.10           87.68
## 8       9E     18460       5.48           93.16
## 9       WN     12275       3.64           96.80
## 10      VX      5162       1.53           98.33
## 11      FL      3260       0.97           99.30
## 12      AS       714       0.21           99.51
## 13      F9       685       0.20           99.71
## 14      YV       601       0.18           99.89
## 15      HA       342       0.10           99.99
## 16      OO        32       0.01          100.00
##
##      tailnum frequency percentage cumulative_perc
## 1       <NA>      2512       0.75            0.75
## 2     N725MQ       575       0.17            0.92
## 3     N722MQ       513       0.15            1.07
## 4     N723MQ       507       0.15            1.22
## 5     N711MQ       486       0.14            1.36
## 6     N713MQ       483       0.14            1.50
## 7     N258JB       427       0.13            1.63
## 8     N298JB       407       0.12            1.75
## 9     N353JB       404       0.12            1.87
## 10    N351JB       402       0.12            1.99
## 11    N735MQ       396       0.12            2.11
## 12    N328AA       393       0.12            2.23
## 13    N228JB       388       0.12            2.35
## 14    N338AA       388       0.12            2.47
## 15    N327AA       387       0.11            2.58
## 16    N335AA       385       0.11            2.69
## 17    N0EGMQ       371       0.11            2.80
## 18    N274JB       370       0.11            2.91
## 19    N324JB       370       0.11            3.02
## 20    N229JB       364       0.11            3.13
## 21    N534MQ       364       0.11            3.24
## 22    N542MQ       363       0.11            3.35
## 23    N190JB       362       0.11            3.46
## 24    N183JB       361       0.11            3.57
## 25    N296JB       357       0.11            3.68
##  [ reached getOption("max.print") -- omitted 4019 rows ]
```


origin
EWR
JFK
LGA
120835 (35.88%)
111279 (33.04%)
104662 (31.08%)
Frequency / (Percentage %)

```
##   origin frequency percentage cumulative_perc
## 1    EWR    120835      35.88           35.88
## 2    JFK    111279      33.04           68.92
## 3    LGA    104662      31.08          100.00
##
##     dest frequency percentage cumulative_perc
## 1    ORD     17283       5.13            5.13
## 2    ATL     17215       5.11           10.24
## 3    LAX     16174       4.80           15.04
## 4    BOS     15508       4.60           19.64
## 5    MCO     14082       4.18           23.82
## 6    CLT     14064       4.18           28.00
## 7    SFO     13331       3.96           31.96
## 8    FLL     12055       3.58           35.54
## 9    MIA     11728       3.48           39.02
## 10   DCA      9705       2.88           41.90
## 11   DTW      9384       2.79           44.69
## 12   DFW      8738       2.59           47.28
## 13   RDU      8163       2.42           49.70
## 14   TPA      7466       2.22           51.92
## 15   DEN      7266       2.16           54.08
## 16   IAH      7198       2.14           56.22
## 17   MSP      7185       2.13           58.35
## 18   PBI      6554       1.95           60.30
## 19   BNA      6333       1.88           62.18
## 20   LAS      5997       1.78           63.96
## 21   SJU      5819       1.73           65.69
## 22   IAD      5700       1.69           67.38
## 23   BUF      4681       1.39           68.77
## 24   PHX      4656       1.38           70.15
## 25   CLE      4573       1.36           71.51
##  [ reached getOption("max.print") -- omitted 80 rows ]
```

```
##           year          month            day       dep_time sched_dep_time
##              0              0              0           8255              0
##      dep_delay       arr_time sched_arr_time      arr_delay        carrier
##           8255           8713              0           9430              0
##         flight        tailnum         origin           dest       air_time
##              0           2512              0              0           9430
##       distance           hour         minute      time_hour
##              0              0              0              0
```

Q2. Descrptive Analysis of Flights Datasets.

Lets breakdown the individual elements

```
# Describe the flights data by each columns to get Insights
describe(flights)
```

```
## flights
##
##  19  Variables      336776  Observations
## --------------------------------------------------------------------------
## year
##        n  missing distinct     Info     Mean      Gmd
##   336776        0        1        0     2013        0
##
## Value        2013
## Frequency  336776
## Proportion      1
## --------------------------------------------------------------------------
## month
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0       12    0.993    6.549    3.929        1        2
##      .25      .50      .75      .90      .95
##        4        7       10       11       12
##
## Value          1     2     3     4     5     6     7     8     9    10
## Frequency  27004 24951 28834 28330 28796 28243 29425 29327 27574 28889
## Proportion 0.080 0.074 0.086 0.084 0.086 0.084 0.087 0.087 0.082 0.086
##
## Value         11    12
## Frequency  27268 28135
## Proportion 0.081 0.084
## --------------------------------------------------------------------------
## day
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0       31    0.999    15.71    10.12        2        4
##      .25      .50      .75      .90      .95
##        8       16       23       28       29
##
## lowest :  1  2  3  4  5, highest: 27 28 29 30 31
## --------------------------------------------------------------------------
## dep_time
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   328521     8255     1318        1     1349    561.9      624      703
##      .25      .50      .75      .90      .95
##      907     1401     1744     2008     2112
##
## lowest :    1    2    3    4    5, highest: 2356 2357 2358 2359 2400
## --------------------------------------------------------------------------
## sched_dep_time
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0     1021        1     1344    538.6      630      705
##      .25      .50      .75      .90      .95
##      906     1359     1729     1945     2050
##
## lowest :  106  500  501  505  510, highest: 2345 2352 2355 2358 2359
## --------------------------------------------------------------------------
## dep_delay
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   328521     8255      527    0.998    12.64    29.47       -9       -7
```

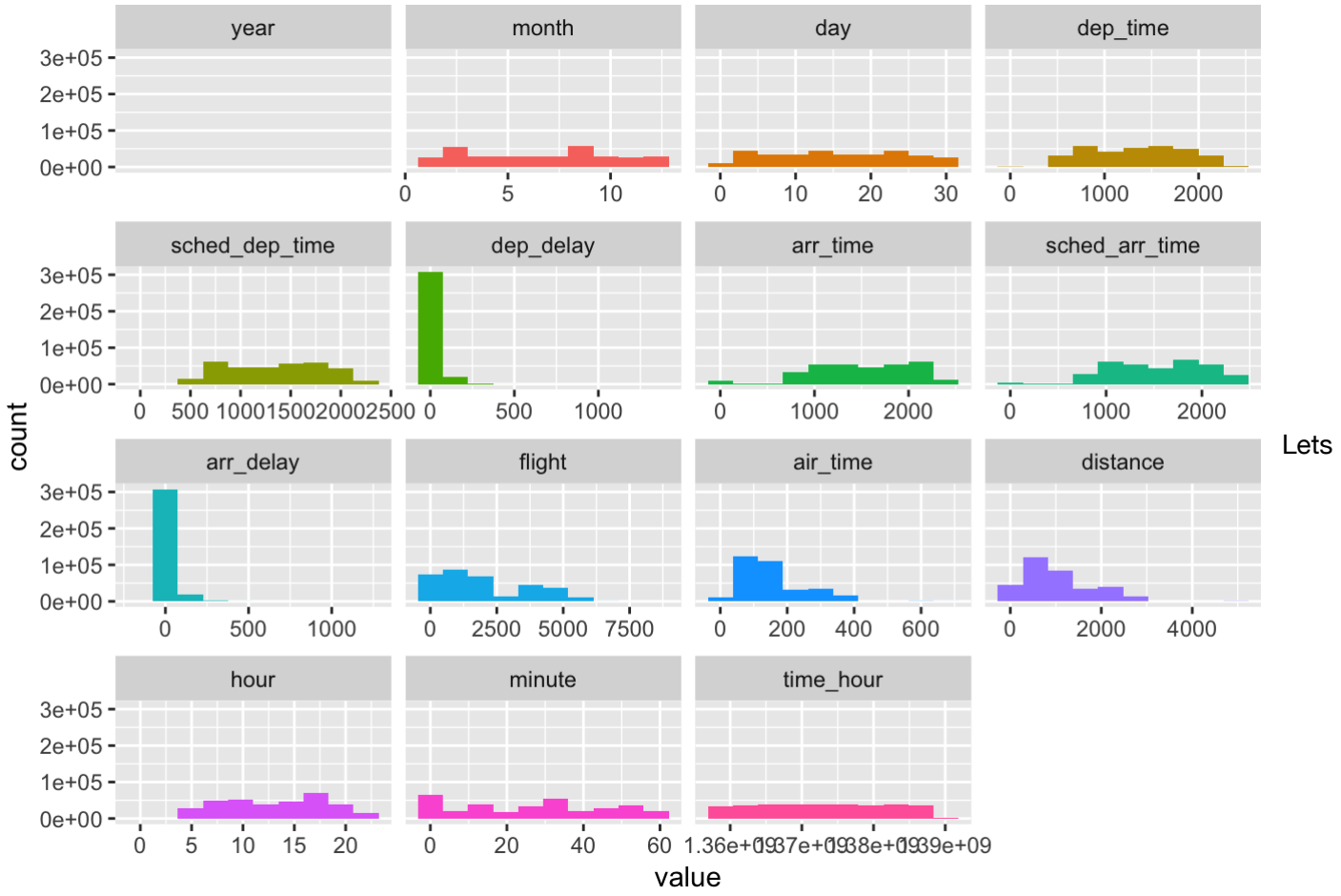
```
##      .25      .50      .75      .90      .95
##       -5       -2       11       49       88
##
## lowest :  -43  -33  -32  -30  -27, highest: 1005 1014 1126 1137 1301
## ---------------------------------------------------------------------------
## arr_time
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   328063     8713     1411        1     1502    602.2      736      853
##      .25      .50      .75      .90      .95
##     1104     1535     1940     2159     2248
##
## lowest :    1    2    3    4    5, highest: 2356 2357 2358 2359 2400
## ---------------------------------------------------------------------------
## sched_arr_time
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0     1163        1     1536    565.2      815      917
##      .25      .50      .75      .90      .95
##     1124     1556     1945     2200     2246
##
## lowest :    1    2    3    4    5, highest: 2355 2356 2357 2358 2359
## ---------------------------------------------------------------------------
## arr_delay
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   327346     9430      577        1    6.895    39.23      -32      -26
##      .25      .50      .75      .90      .95
##      -17       -5       14       52       91
##
## lowest :  -86  -79  -75  -74  -73, highest:  989 1007 1109 1127 1272
## ---------------------------------------------------------------------------
## carrier
##        n  missing distinct
##   336776        0       16
##
## Value         9E    AA    AS    B6    DL    EV    F9    FL    HA    MQ
## Frequency  18460 32729   714 54635 48110 54173   685  3260   342 26397
## Proportion 0.055 0.097 0.002 0.162 0.143 0.161 0.002 0.010 0.001 0.078
##
## Value         OO    UA    US    VX    WN    YV
## Frequency     32 58665 20536  5162 12275   601
## Proportion 0.000 0.174 0.061 0.015 0.036 0.002
## ---------------------------------------------------------------------------
## flight
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0     3844        1     1972     1817       91      209
##      .25      .50      .75      .90      .95
##      553     1496     3465     4471     4695
##
## lowest :    1    2    3    4    5, highest: 6171 6177 6180 6181 8500
## ---------------------------------------------------------------------------
## tailnum
##        n  missing distinct
##   334264     2512     4043
##
## lowest : D942DN N0EGMQ N10156 N102UW N103US, highest: N997DL N998AT N998DL N999DN N9E
```

```
AMQ
## ------------------------------------------------------------------------
## origin
##        n  missing distinct
##   336776        0        3
##
## Value         EWR    JFK    LGA
## Frequency  120835 111279 104662
## Proportion  0.359  0.330  0.311
## ------------------------------------------------------------------------
## dest
##        n  missing distinct
##   336776        0      105
##
## lowest : ABQ ACK ALB ANC ATL, highest: TPA TUL TVC TYS XNA
## ------------------------------------------------------------------------
## air_time
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   327346     9430      509        1    150.7    101.1       40       47
##      .25      .50      .75      .90      .95
##       82      129      192      319      339
##
## lowest :  20  21  22  23  24, highest: 679 683 686 691 695
## ------------------------------------------------------------------------
## distance
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0      214        1     1040      786      199      214
##      .25      .50      .75      .90      .95
##      502      872     1389     2446     2475
##
## lowest :   17   80   94   96  116, highest: 2576 2586 3370 4963 4983
## ------------------------------------------------------------------------
## hour
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0       20    0.996    13.18    5.365        6        7
##      .25      .50      .75      .90      .95
##        9       13       17       19       20
##
## Value          1     5     6     7     8     9    10    11    12    13
## Frequency      1  1953 25951 22821 27242 20312 16708 16033 18181 19956
## Proportion 0.000 0.006 0.077 0.068 0.081 0.060 0.050 0.048 0.054 0.059
##
## Value         14    15    16    17    18    19    20    21    22    23
## Frequency  21706 23888 23002 24426 21783 21441 16739 10933  2639  1061
## Proportion 0.064 0.071 0.068 0.073 0.065 0.064 0.050 0.032 0.008 0.003
## ------------------------------------------------------------------------
## minute
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   336776        0       60    0.992    26.23    22.14        0        0
##      .25      .50      .75      .90      .95
##        8       29       44       55       58
##
## lowest :  0  1  2  3  4, highest: 55 56 57 58 59
## ------------------------------------------------------------------------
```

```
## time_hour
##                       n                 missing                distinct
##                  336776                       0                    6936
##                    Info                    Mean                     Gmd
##                       1 2013-07-03 05:22:54 1970-05-01 05:53:45
##                     .05                     .10                     .25
## 2013-01-20 12:00:00 2013-02-08 16:00:00 2013-04-04 13:00:00
##                     .50                     .75                     .90
## 2013-07-03 10:00:00 2013-10-01 07:00:00 2013-11-24 14:00:00
##                     .95
## 2013-12-13 06:00:00
##
## lowest : 2013-01-01 05:00:00 2013-01-01 06:00:00 2013-01-01 07:00:00 2013-01-01 08:0
0:00 2013-01-01 09:00:00
## highest: 2013-12-31 19:00:00 2013-12-31 20:00:00 2013-12-31 21:00:00 2013-12-31 22:0
0:00 2013-12-31 23:00:00
## ---------------------------------------------------------------------------
```

Visual Representation of Datasets

```
# Attributes which are not identical across measure variables will be dropped.Also all b
iwidht must be positive.
plot_num(flights)
```



Lets check the status of our datasets. Let us provide indepth knowledge of missing datasets.

DF status of Flights

```
df_status(flights) # Its good for analysing Missing datasets and to spot unique numbers.
```

```
##          variable q_zeros p_zeros q_na p_na q_inf p_inf           type
## 1            year       0    0.00    0 0.00     0     0        integer
## 2           month       0    0.00    0 0.00     0     0        integer
## 3             day       0    0.00    0 0.00     0     0        integer
## 4        dep_time       0    0.00 8255 2.45     0     0        integer
## 5  sched_dep_time       0    0.00    0 0.00     0     0        integer
## 6       dep_delay   16514    4.90 8255 2.45     0     0        numeric
## 7        arr_time       0    0.00 8713 2.59     0     0        integer
## 8  sched_arr_time       0    0.00    0 0.00     0     0        integer
## 9       arr_delay    5409    1.61 9430 2.80     0     0        numeric
## 10        carrier       0    0.00    0 0.00     0     0      character
## 11         flight       0    0.00    0 0.00     0     0        integer
##    unique
## 1       1
## 2      12
## 3      31
## 4    1318
## 5    1021
## 6     527
## 7    1411
## 8    1163
## 9     577
## 10     16
## 11   3844
##  [ reached getOption("max.print") -- omitted 8 rows ]
```

As we have seen that we have missing datasets in the number of columns. We need to address those variables before we diagnose our flights' datasets.

Handling the Missing datasets

```
print(" The number of missing values in Flights datasets is ")
```

```
## [1] " The number of missing values in Flights datasets is "
```

```
sapply(flights, function(x) sum(is.na(x)))
```

```
##           year          month            day       dep_time sched_dep_time
##              0              0              0           8255              0
##      dep_delay       arr_time sched_arr_time      arr_delay        carrier
##           8255           8713              0           9430              0
##         flight        tailnum         origin           dest       air_time
##              0           2512              0              0           9430
##       distance           hour         minute      time_hour
##              0              0              0              0
```

```
print("The number of missing values in Airlines datasets is")
```

```
## [1] "The number of missing values in Airlines datasets is"
```

```
sapply(airlines, function(x) sum(is.na(x)))
```

```
## carrier    name
##       0       0
```

```
print(" The number of missing values in Airports datasets is ")
```

```
## [1] " The number of missing values in Airports datasets is "
```

```
sapply(airports, function(x) sum(is.na(x)))
```

```
##   faa  name   lat   lon   alt    tz   dst tzone
##     0     0     0     0     0     0     0     0
```

```
print(" The number of missing values in Weather datasets is ")
```

```
## [1] " The number of missing values in Weather datasets is "
```

```
sapply(weather, function(x) sum(is.na(x)))
```

```
##     origin       year      month        day       hour       temp
##          0          0          0          0          0          1
##       dewp      humid   wind_dir wind_speed  wind_gust     precip
##          1          1        460          4      20778          0
##   pressure      visib  time_hour
##       2729          0          0
```

MISSING DATA ANALYSIS

- In Flights datasets we have
    - dep_time : 8255 Missing data
    - dep_delay : 8255 Missing data

    - arr_time : 8713 Missing data
    - arr_delay : 9430 Missing data
    - tailnum : 2512 Missing data
    - air_time : 9430 Missing data
- We don't have any Missing data in Airlines datasets.
- We don't have any Missing data in Airports datasets.
- In Weather datasets we have:
- temp : 1 Missing data
- dewp : 1 Missing data
- humid : 1 Missing data
- wind_dir : 460 Missing data
- wind_speed : 4 Missing data
- wind_gust : 20778 Missing data
- pressure : 2729 Missing data

Just for simplicity purpose (quick and dirty analysis), We will omit all the missing datasets to get the gist of Exploratory Data Analysis but while building upon the model or making a coorelation assumption as well as doing more detail linear Regression we will take care of these missing data in detail effort. For now let's drop the missing datasets.

```
flt_1 <- flights %>% na.omit()
# Sanity check
sum(is.na(flt_1))
```

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

```
# Detail Sanity check
sapply(flt_1, function(x) sum(is.na(x)))
```

```
##           year          month            day       dep_time sched_dep_time
##              0              0              0              0              0
##      dep_delay       arr_time sched_arr_time      arr_delay        carrier
##              0              0              0              0              0
##         flight        tailnum         origin           dest       air_time
##              0              0              0              0              0
##       distance           hour         minute      time_hour
##              0              0              0              0
```

We can Confirm that the "flt_1" datasets we just created from flights datasets don't have any missing datasets in it. we have dropped rows from the original datasets. Sanity check was run through.

After Handling the missing data we can work with the datasets. So lets dive have a glimpse of our datasets and fix its class so that we can easily explore datasets without any hickups.

```
glimpse(flt_1)
```

```
## Observations: 327,346
## Variables: 19
## $ year           <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013,...
## $ month          <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ day            <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ dep_time       <int> 517, 533, 542, 544, 554, 554, 555, 557, 557, 55...
## $ sched_dep_time <int> 515, 529, 540, 545, 600, 558, 600, 600, 600, 60...
## $ dep_delay      <dbl> 2, 4, 2, -1, -6, -4, -5, -3, -3, -2, -2, -2, -2...
## $ arr_time       <int> 830, 850, 923, 1004, 812, 740, 913, 709, 838, 7...
## $ sched_arr_time <int> 819, 830, 850, 1022, 837, 728, 854, 723, 846, 7...
## $ arr_delay      <dbl> 11, 20, 33, -18, -25, 12, 19, -14, -8, 8, -2, -...
## $ carrier        <chr> "UA", "UA", "AA", "B6", "DL", "UA", "B6", "EV",...
## $ flight         <int> 1545, 1714, 1141, 725, 461, 1696, 507, 5708, 79...
## $ tailnum        <chr> "N14228", "N24211", "N619AA", "N804JB", "N668DN...
## $ origin         <chr> "EWR", "LGA", "JFK", "JFK", "LGA", "EWR", "EWR"...
## $ dest           <chr> "IAH", "IAH", "MIA", "BQN", "ATL", "ORD", "FLL"...
## $ air_time       <dbl> 227, 227, 160, 183, 116, 150, 158, 53, 140, 138...
## $ distance       <dbl> 1400, 1416, 1089, 1576, 762, 719, 1065, 229, 94...
## $ hour           <dbl> 5, 5, 5, 5, 6, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 5,...
## $ minute         <dbl> 15, 29, 40, 45, 0, 58, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ time_hour      <dttm> 2013-01-01 05:00:00, 2013-01-01 05:00:00, 2013...
```

Type Conversion

```
#convert the types of existing variables from character to factor and store back to its
 datasets.
flt_1$carrier<-factor(flt_1$carrier)
flt_1$tailnum<-factor(flt_1$tailnum)
flt_1$origin<-factor(flt_1$origin)
flt_1$dest<-factor(flt_1$dest)
```

Join

Flights datasets with Airliness, Airports

Let's create one dataset with all the necessary information. We have a dataset with carrier name but we are missing its full acronym, which can be founded in airlines datasets. Below in these codes, we will join flt_1 with airlines' datasets.

```
##  [1] "year"           "month"          "day"            "dep_time"
##  [5] "sched_dep_time" "dep_delay"      "arr_time"       "sched_arr_time"
##  [9] "arr_delay"      "carrier"        "flight"         "tailnum"
## [13] "origin"         "dest"           "air_time"       "distance"
## [17] "hour"           "minute"         "time_hour"      "name"
```

After we have successfully joined flights datasets with the Airlines's datasets. We can similarly do with Airports datasets. Let's look at the column name of airport datasets. Here We have faa code given to each airport which can be matched with airports origin and destination.

```
colnames(airports)
```

```
## [1] "faa"   "name"  "lat"   "lon"   "alt"   "tz"    "dst"   "tzone"
```

```
# unique(airports$faa) #  Uncomment it see all the unique airports faa code.
```

```
# Left join faa code with destination.
flt_1 <- left_join(flt_1, airports , by = c("dest" = "faa"), copy = false)
#colnames(flt_1) # Uncomment to do sanity check.
flt_1 <- rename(flt_1, "arrival_airport"= "name") # change name which is coming from air
ports to arrival_airport.

# Left join faa code with origin.
flt_1 <- left_join(flt_1, airports , by = c("origin" = "faa"), copy = false)
#colnames(flt_1) # Uncomment to do sanity check.
flt_1 <- rename(flt_1, "departure_airport"= "name") # change name which is coming from a
irports to arrival_airport.

#colnames(flt_1) # Uncomment to do sanity check.
glimpse(flt_1) # Lets have a look to our datasets, confirm everything has been imported
 as we wish for.
```

```
## Observations: 327,346
## Variables: 34
## $ year              <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 20...
## $ month             <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ day               <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ dep_time          <int> 517, 533, 542, 544, 554, 554, 555, 557, 557,...
## $ sched_dep_time    <int> 515, 529, 540, 545, 600, 558, 600, 600, 600,...
## $ dep_delay         <dbl> 2, 4, 2, -1, -6, -4, -5, -3, -3, -2, -2, -2,...
## $ arr_time          <int> 830, 850, 923, 1004, 812, 740, 913, 709, 838...
## $ sched_arr_time    <int> 819, 830, 850, 1022, 837, 728, 854, 723, 846...
## $ arr_delay         <dbl> 11, 20, 33, -18, -25, 12, 19, -14, -8, 8, -2...
## $ carrier           <chr> "UA", "UA", "AA", "B6", "DL", "UA", "B6", "E...
## $ flight            <int> 1545, 1714, 1141, 725, 461, 1696, 507, 5708,...
## $ tailnum           <fct> N14228, N24211, N619AA, N804JB, N668DN, N394...
## $ origin            <chr> "EWR", "LGA", "JFK", "JFK", "LGA", "EWR", "E...
## $ dest              <chr> "IAH", "IAH", "MIA", "BQN", "ATL", "ORD", "F...
## $ air_time          <dbl> 227, 227, 160, 183, 116, 150, 158, 53, 140, ...
## $ distance          <dbl> 1400, 1416, 1089, 1576, 762, 719, 1065, 229,...
## $ hour              <dbl> 5, 5, 5, 5, 6, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6,...
## $ minute            <dbl> 15, 29, 40, 45, 0, 58, 0, 0, 0, 0, 0, 0, 0, ...
## $ time_hour         <dttm> 2013-01-01 05:00:00, 2013-01-01 05:00:00, 2...
## $ carrier_name      <chr> "United Air Lines Inc.", "United Air Lines I...
## $ arrival_airport   <chr> "George Bush Intercontinental", "George Bush...
## $ lat.x             <dbl> 29.98443, 29.98443, 25.79325, NA, 33.63672, ...
## $ lon.x             <dbl> -95.34144, -95.34144, -80.29056, NA, -84.428...
## $ alt.x             <int> 97, 97, 8, NA, 1026, 668, 9, 313, 96, 668, 1...
## $ tz.x              <dbl> -6, -6, -5, NA, -5, -6, -5, -5, -5, -6, -5, ...
## $ dst.x             <chr> "A", "A", "A", NA, "A", "A", "A", "A", "A", ...
## $ tzone.x           <chr> "America/Chicago", "America/Chicago", "Ameri...
## $ departure_airport <chr> "Newark Liberty Intl", "La Guardia", "John F...
## $ lat.y             <dbl> 40.69250, 40.77725, 40.63975, 40.63975, 40.7...
## $ lon.y             <dbl> -74.16867, -73.87261, -73.77893, -73.77893, ...
## $ alt.y             <int> 18, 22, 13, 13, 22, 18, 18, 22, 13, 22, 13, ...
## $ tz.y              <dbl> -5, -5, -5, -5, -5, -5, -5, -5, -5, -5, -5, ...
## $ dst.y             <chr> "A", "A", "A", "A", "A", "A", "A", "A", "A",...
## $ tzone.y           <chr> "America/New_York", "America/New_York", "Ame...
```

Factor Conversion

For calculations purpose, we will convert the following variables to factor. i.e. carrier_name, departure_airport, origin, arrival_airport so that we can do basic calculations.

```
flt_1$carrier_name<-factor(flt_1$carrier_name)
flt_1$departure_airport<-factor(flt_1$departure_airport)
flt_1$origin<-factor(flt_1$origin)
flt_1$arrival_airport<-factor(flt_1$arrival_airport)
flt_1$month <- factor(flt_1$month)
```

Flight delays due to weather are an unfortunate reality, especially at some of the busiest airport hubs in the United States.After we have done all these type conversions and fixing our datasets without any missing data. Our datasets are ready to get explored. So without any delays, let's jump into Exploratory Data Analysis. First Lets

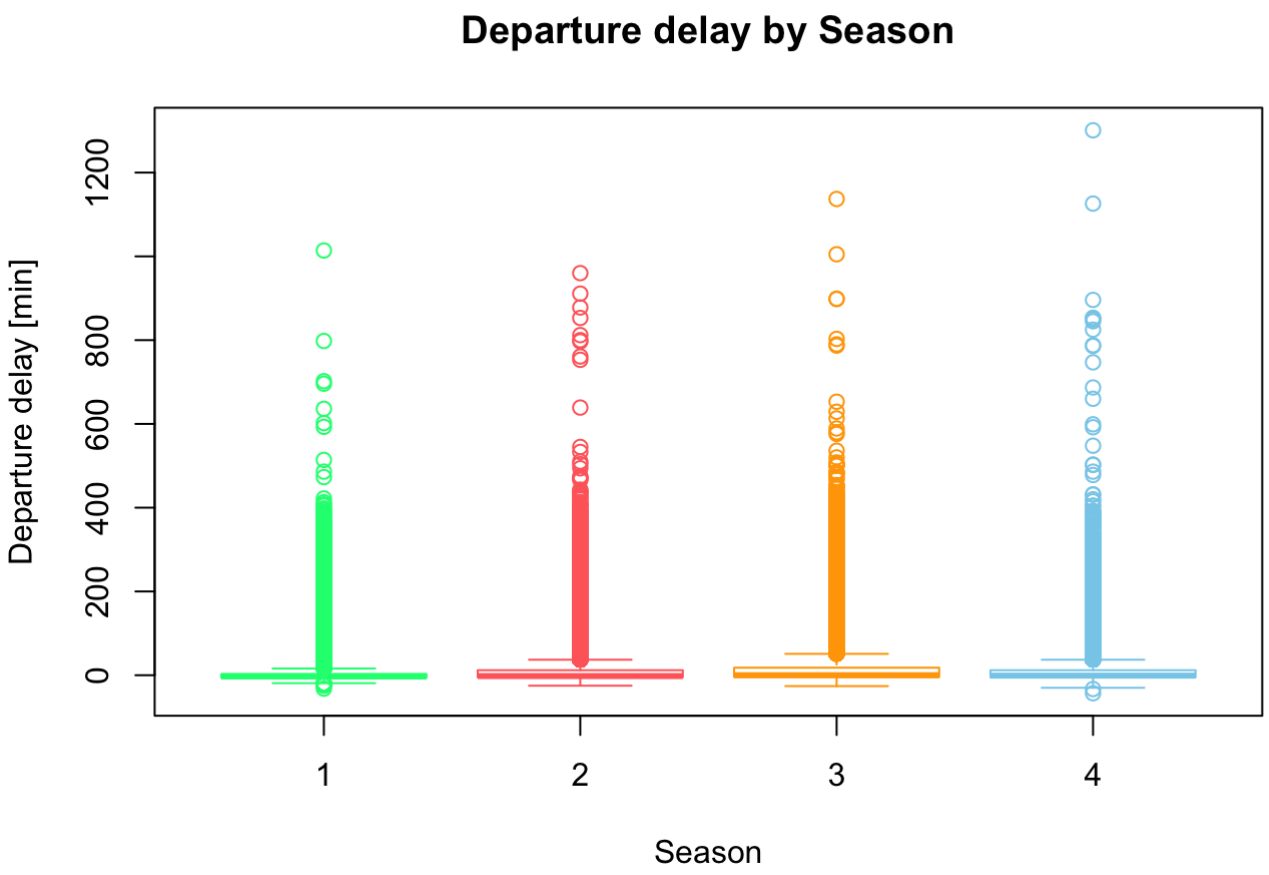
break down the casual delay by Season. We would like to know if the delays are common in all season or it is only in one season. We have month datasets so we will convert month into the season according to a standard Calendar month and see if there are any Seasonal delays.

Q3. Departure delay by Season

```
#create seasons
flt_1<- flt_1 %>% mutate(Season = ifelse(month %in% c(6,7,8), "Summer",
                                                    ifelse(month %in% c(9,10,11), "Fal
l",
                                                           ifelse(month %in% c(12,1,2
), "Winter",
                                                                  "Spring"))))
table(flt_1$Season)
```

```
##
##   Fall Spring Summer Winter
##  82599  83594  84124  77029
```

```
# Boxplot
boxplot(formula = dep_delay ~ Season,
           data = flt_1,
        main = 'Departure delay by Season',
        xlab = 'Season',
        ylab = 'Departure delay [min]',
        border = c('springgreen', 'indianred1', 'orange', 'skyblue'),
        names = c('Spring', 'Summer', 'Fall', 'Winter') +
        theme_dark()+
        theme(
        plot.title = element_text(color="red", size=14, face="bold.italic", hjust = 0.5
),
        axis.title.x = element_text(color="blue", size=14, face="bold"),
        axis.title.y = element_text(color="#993333", size=14, face="bold")))
```



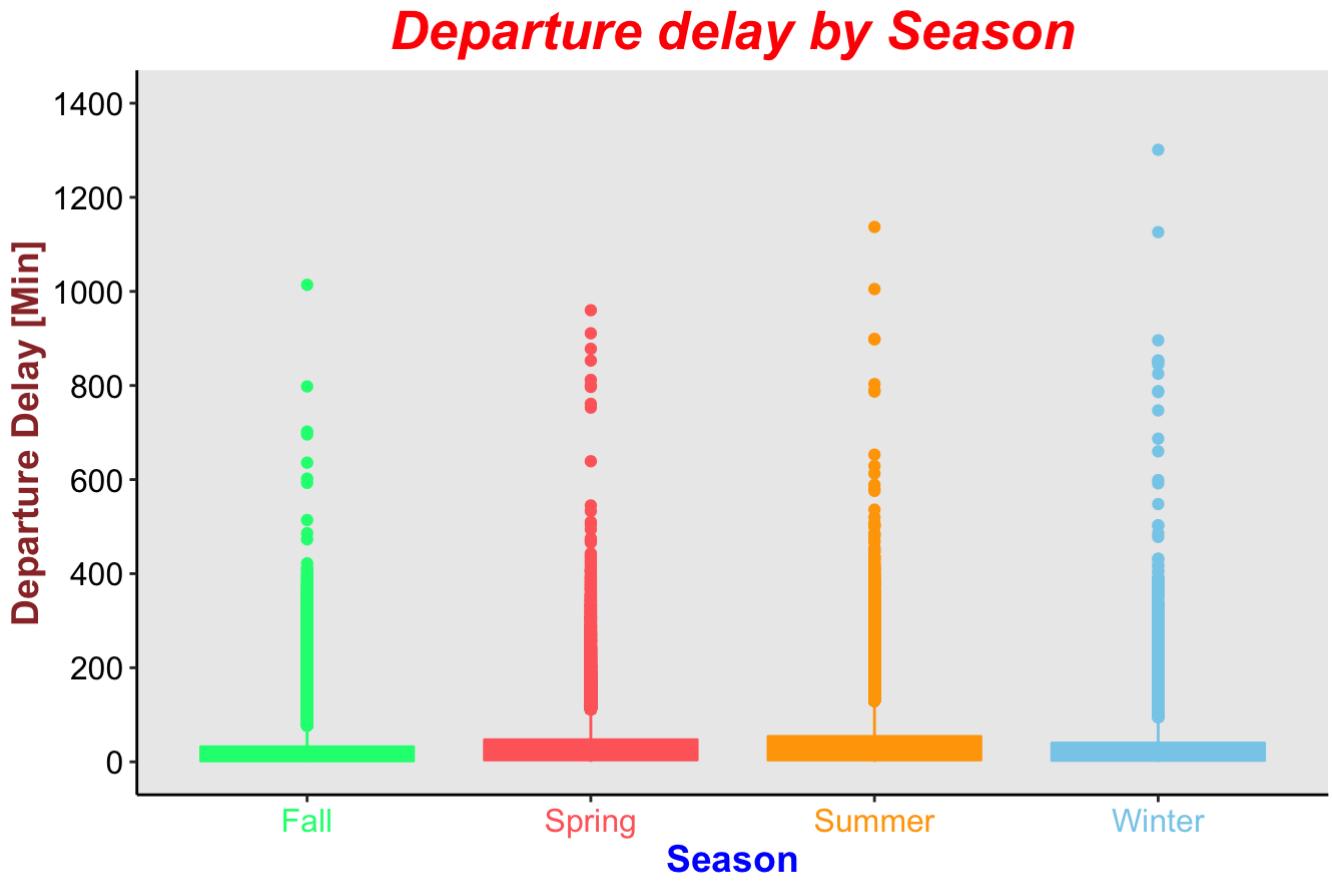

```
aggregated_mean_sd_median <- cbind(
mean = aggregate(formula = dep_delay ~ Season,
           data = flt_1,
           FUN = mean,
            na.rm = T),
sd = aggregate(formula = dep_delay ~ Season,
           data = flt_1,
           FUN = sd,
            na.rm = T),
median= aggregate(formula = dep_delay ~ Season,
           data = flt_1,
           FUN = median,
            na.rm = T)
)
aggregated_mean_sd_median
```

mean.Season <chr>	mean.dep_delay <dbl>	sd.Season <chr>	sd.dep_delay <dbl>	median.Season <chr>	median.dep_delay <dbl>
Fall	6.097616	Fall	31.05661	Fall	-3
Spring	13.298407	Spring	40.72773	Spring	-2
Summer	18.205875	Summer	47.21879	Summer	0
Winter	12.501850	Winter	38.37056	Winter	-1

4 rows

```
# ALT WAY of plotting same graph with geom_boxplot
ggplot(flt_1, aes(x = Season, y = dep_delay)) +
       geom_boxplot(color = c('springgreen', 'indianred1', 'orange', 'skyblue'), fill =
 c('springgreen', 'indianred1', 'orange', 'skyblue')) +
       scale_y_continuous(name = "Departure Delay [Min]",
                          breaks = seq(0, 1400, 200),
                          limits=c(0, 1400)) +
       scale_x_discrete(name = "Season") +
       ggtitle("Departure delay by Season") +
       labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
       theme(axis.line.x = element_line(size = 0.5, colour = "black"),
             axis.line.y = element_line(size = 0.5, colour = "black"),
             axis.line = element_line(size=1, colour = "black"),
             panel.grid.major = element_blank(),
             panel.grid.minor = element_blank(),
             panel.border = element_blank(),
             panel.background = element_rect(size = 0.5, linetype = "solid"),
             axis.title.x = element_text(color="blue", size=14, face="bold"),
             axis.title.y = element_text(color="#993333", size=14, face="bold"),
             plot.title = element_text(size = 20, hjust = 0.5, color = "red", face="bol
d.italic"),
             text=element_text(size = 16),
             plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color
="#1E1E20", face="bold.italic"),
             axis.text.x=element_text(colour=c('springgreen', 'indianred1', 'orange',
'skyblue'), size = 12),
             axis.text.y=element_text(colour="black", size = 12))
```

```
## Warning: Removed 183135 rows containing non-finite values (stat_boxplot).
```



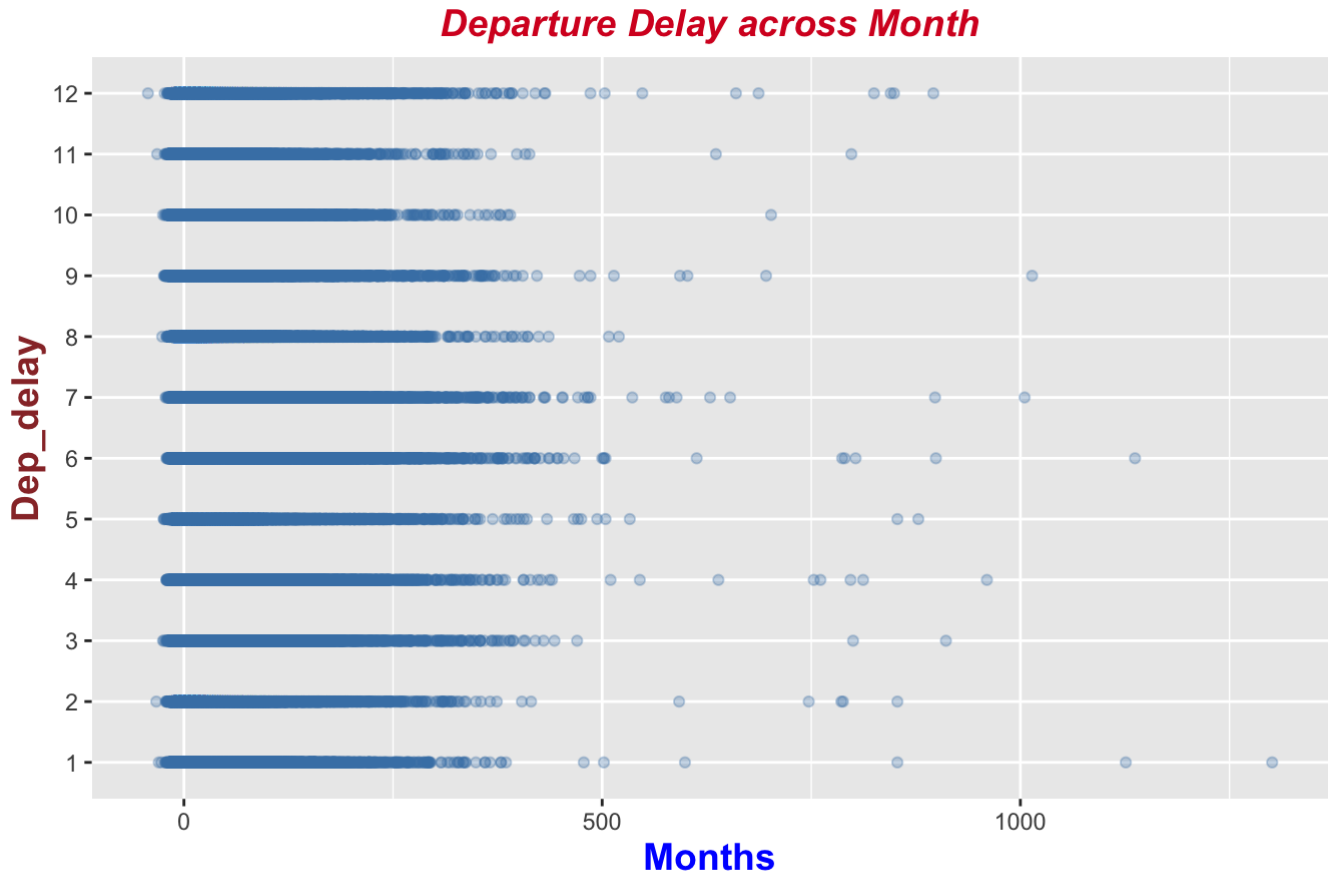
Source: NYC-FLIGHTS datasets | @ Pankaj Shah

ANALYSIS

As we can see from above boxplot that Delays are consistent across the season. During the Winter month, we can see there are some flights which are delayed longer compare to summer, fall and spring. During the Spring the delays are a lot smaller. But overall it's not Seasonal, there are various factors that cause delays besides the weather in winter months. One of the interesting observation we could take from the above graph is why there are delays in the Summer month. When the East Coast Weather is so perfect, no snow storm and weather is getting better. Why are there delays? Let's break down our Seasonality effect to month so that we can see if there is any particular month that has more delays than others.

Q4. Departure Delay across Month in all three airports.

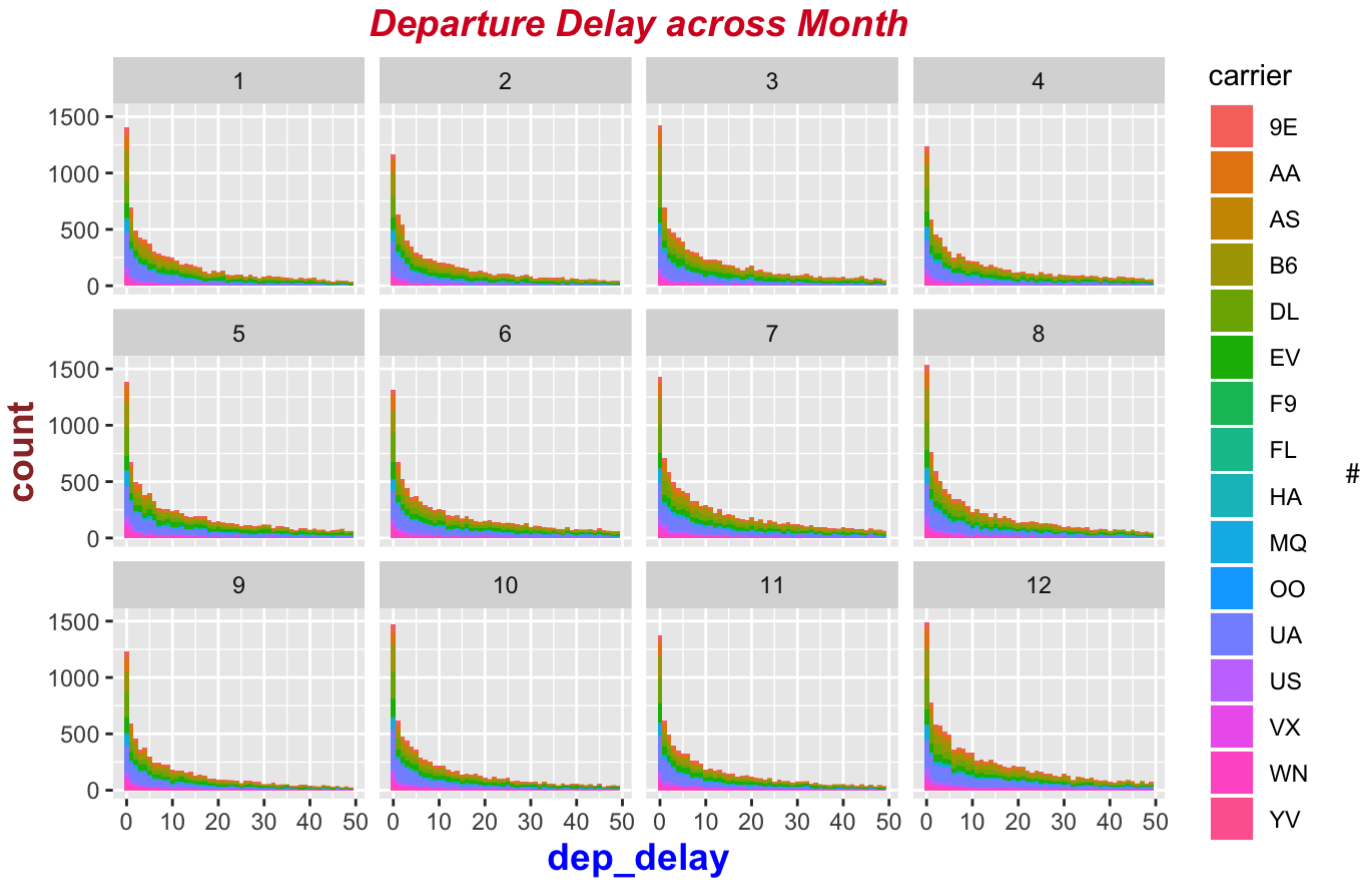
```
ggplot(flt_1, aes(x = month, y = dep_delay) ) +
  geom_point(alpha = 0.3, color = 'steelblue') +
  labs(x="Dep_delay", y="Months",
       caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  ggtitle("Departure Delay across Month") +
  theme(
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#1E1E20",f
ace="bold.italic" ),
  plot.title = element_text(color="#D70026", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))+
  coord_flip()
```



ANALYSIS

Departure delay is equally distributed among all the months. In these plot its really hard to see a huge difference between the months. We can see in some months there are huge delays because of the presence of many outliers but in some months there are less. In the month of January, we can see some outliers are data are fairly spread out. It seems like month June(6th) July (7th) is more densely blue than rest of month. Let's plot another plot where we can see these difference much clear way.

```
t_subset_flight <- subset(flt_1, !is.na(dep_delay), !is.na(arr_delay))
ggplot(aes(x = dep_delay), data = subset(t_subset_flight, dep_delay >= 0 & dep_delay <=
 quantile(dep_delay, .90) ) ) +
  geom_histogram(aes(fill = carrier), binwidth = 1) +
  facet_wrap(~month)+ # break down by carrier.
  ggtitle("Departure Delay across Month") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#1E1E20",f
ace="bold.italic" ),
  plot.title = element_text(color="#D70026", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Q5.Departure Delay across months Statistical Analysis

```
flt_1 %>%
  group_by(month) %>%
  summarise(mean_dd = mean(dep_delay)) %>%
  arrange(desc(mean_dd))
```

month	mean_dd
<fctr>	<dbl>
7	21.522179

month <fctr>	mean_dd <dbl>
6	20.725614
12	16.482161
4	13.849187
3	13.164289
5	12.891709
8	12.570524
2	10.760239
1	9.985491
9	6.630285

1-10 of 12 rows Previous **1** 2 Next

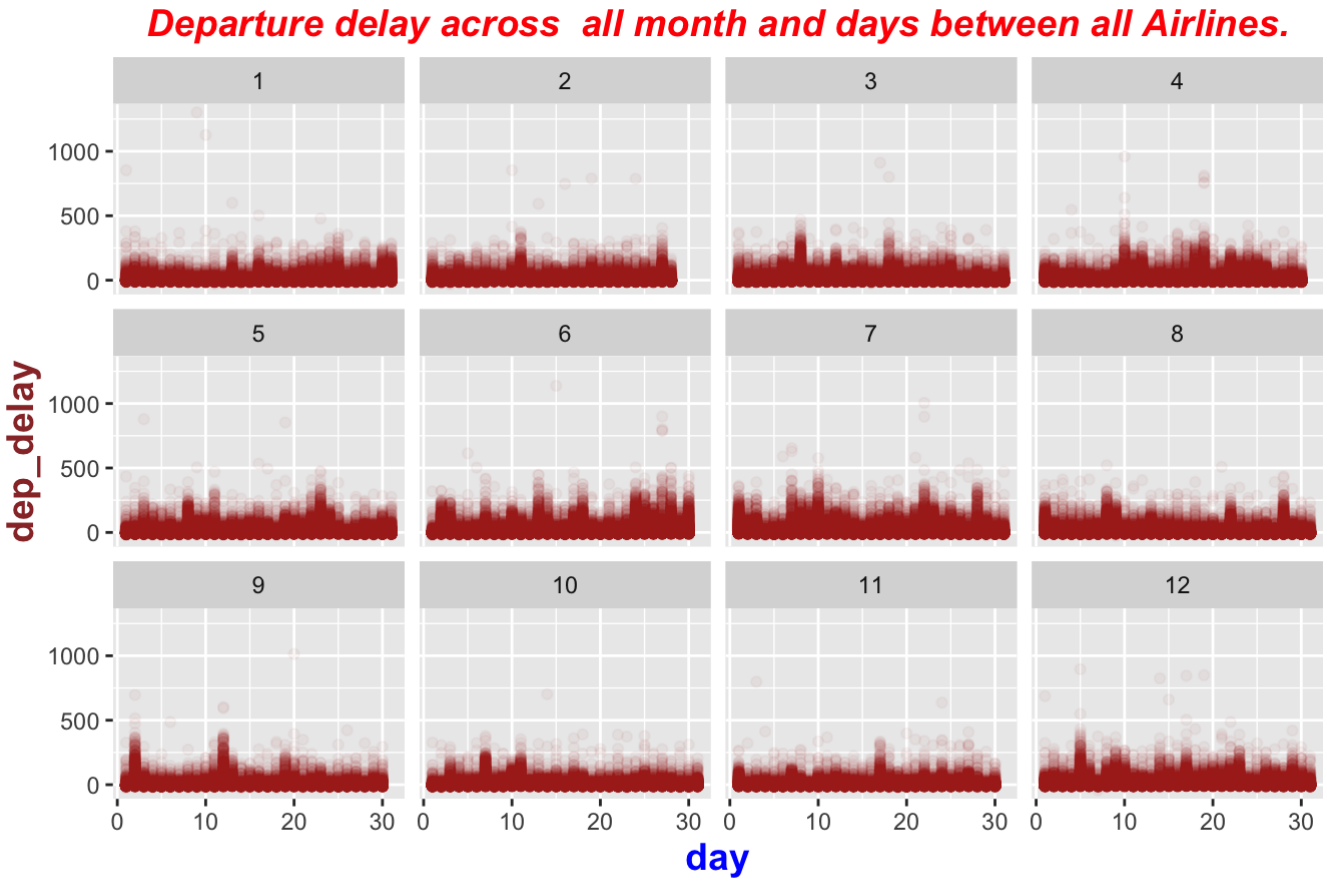
Q6. Which month has the highest average departure delay from an NYC airport?

- July
- June
- December

July, followed by June is the month with the highest average delay of flights departing from an NYC airport. A high average mean of delay has also observed in December, suggesting that the problem lies in the number of flights during the Holidays. The months with the lowest average of departure delays are September to November.

Q7. Departure Delay Across Month and days.

```
subset_flight <- subset(flt_1, !is.na(dep_delay), !is.na(arr_delay))
ggplot(aes(x = day, y = dep_delay), data = subset_flight ) +
  geom_point(alpha = 0.04, color = 'brown') +
  ggtitle("Departure delay across  all month and days between all Airlines.") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  plot.title = element_text(color="red", size=14, face="bold.italic", hjust = 0.5),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#1E1E20",
 face="bold.italic"),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))+
  facet_wrap(~month)
```

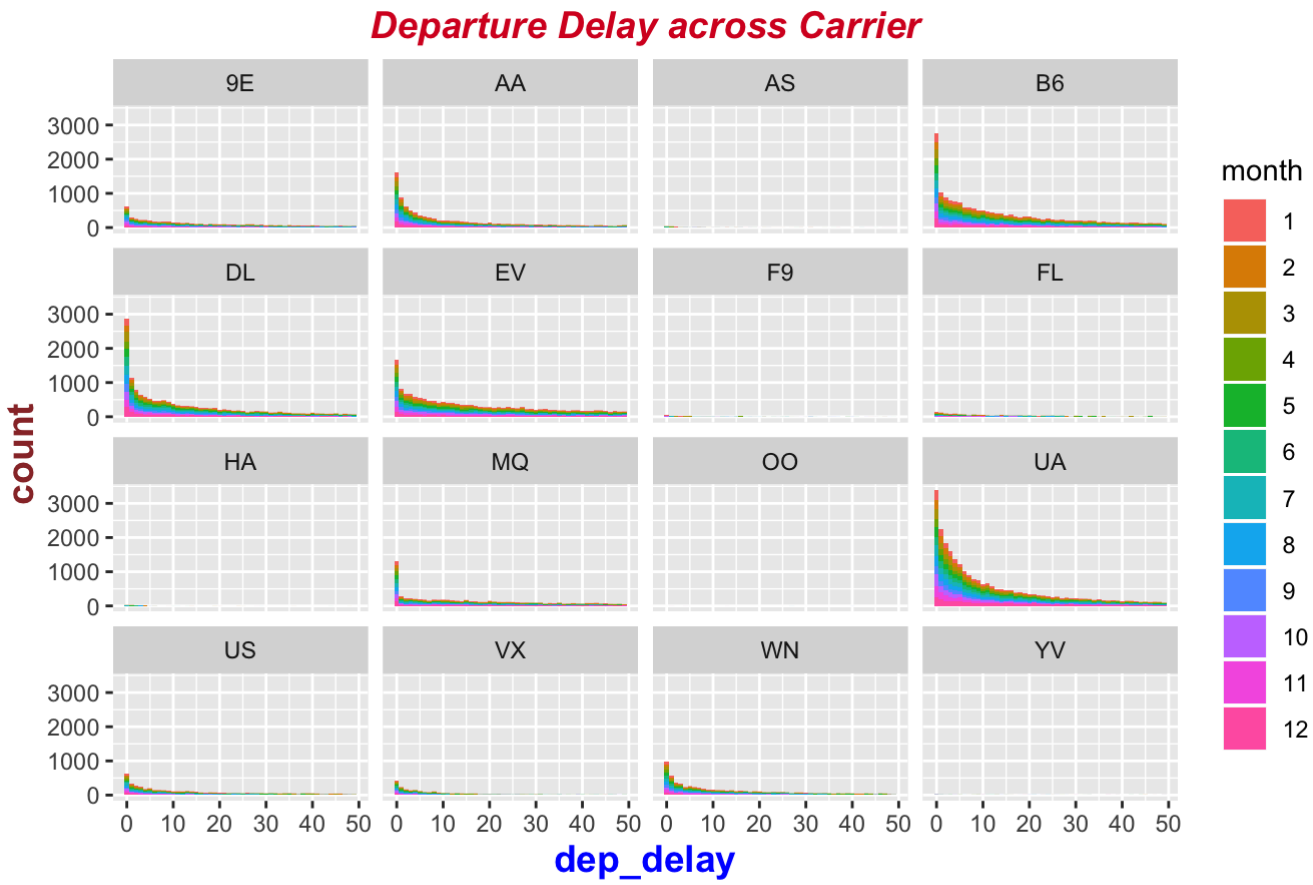


Q8.

Departure Delay across Carrier

Lets also breakdown the analysis by carrier so that we can see if these is across all airlines.

```
ggplot(aes(x = dep_delay), data = subset(subset_flight, dep_delay >= 0 & dep_delay <= qu
antile(dep_delay, .90) ) ) +
  geom_histogram(aes(fill = month), binwidth = 1) +
  facet_wrap(~carrier)+ # break down by carrier.
  ggtitle("Departure Delay across Carrier") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#1E1E20",f
ace="bold.italic" ),
  plot.title = element_text(color="#D70026", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

ANALYSIS

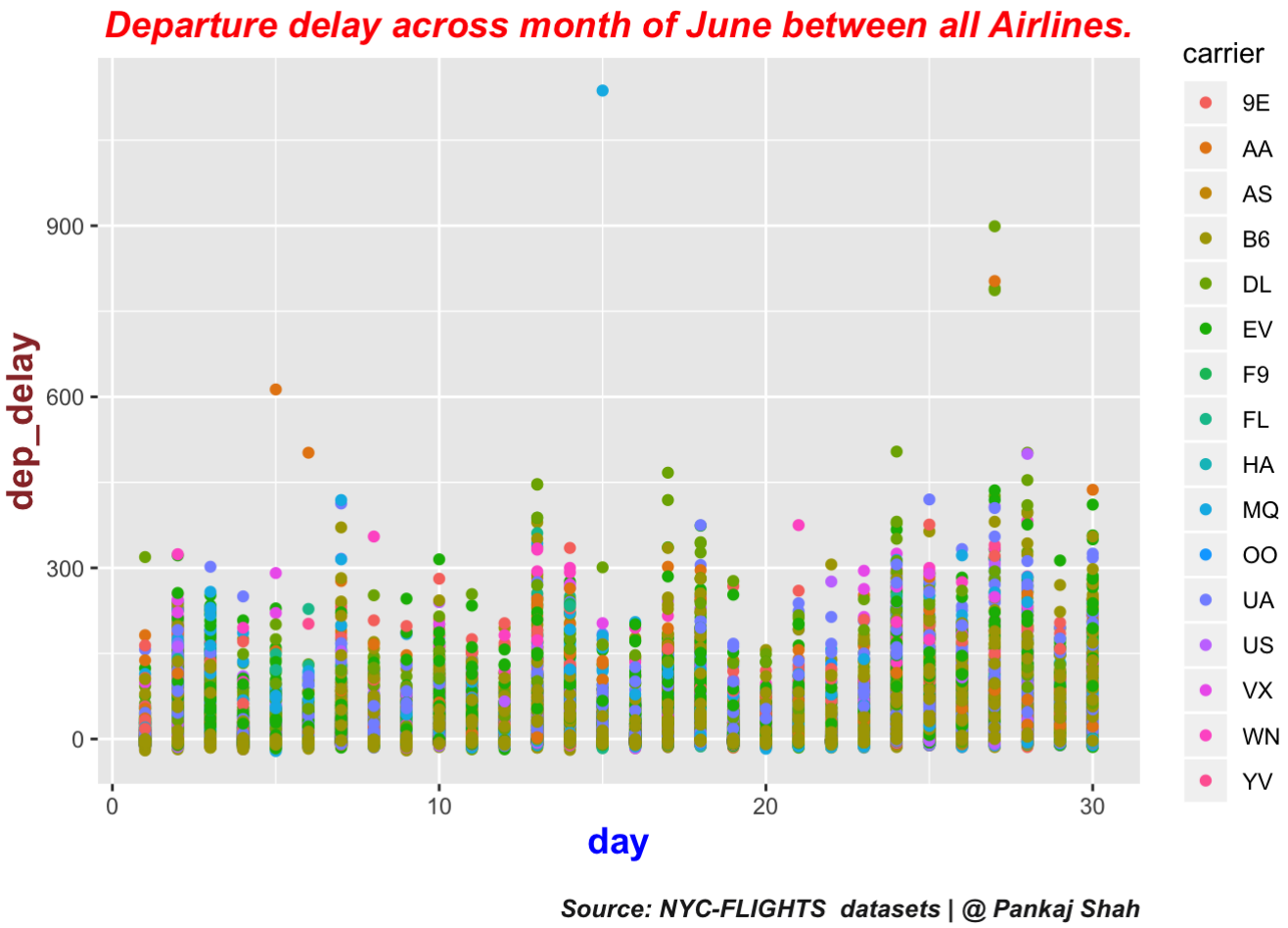
- Major contributor :
- AA : American Airliness.
- B6 : Jet Blue Airliness.
- DL : Delta Airliness.
- EV : Express Jet lines
- UA : United Airliness (Biggest Contributor)

  Seems reasonanble as it does more flights than others.

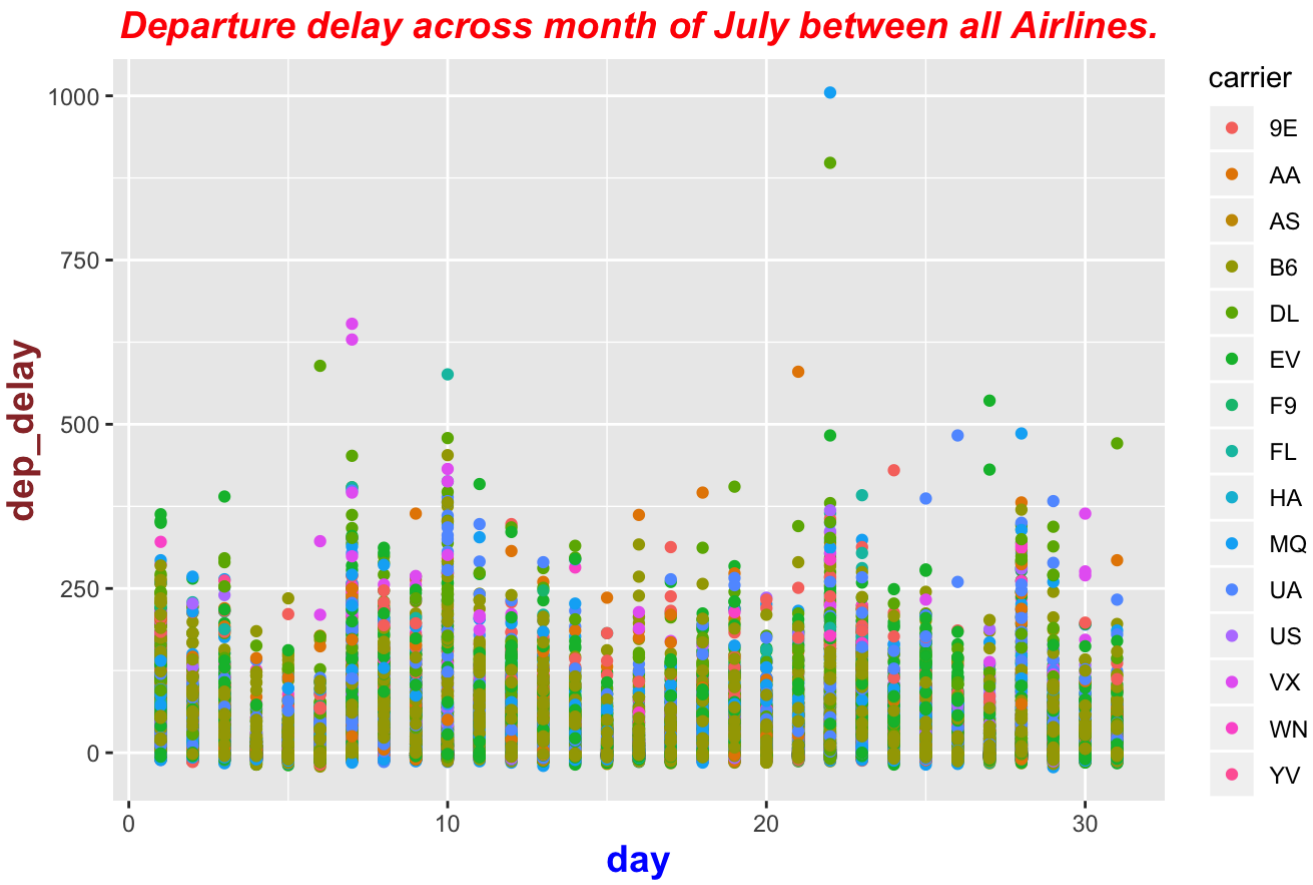
As it is clear from the above graph that we are seeing more delays in the month of June and July. We also can see which carrier is playing the major role in causing such delays. Below I will plot the Histogram plot to see the break down of those two months.

Q9. Departure Delay in June & July

```
# Histogram plot for June
June <- subset(flt_1, month == 6)
June <- subset(June, !is.na(dep_delay), !is.na(arr_delay))

ggplot(aes(x = day, y = dep_delay), data = June ) +
  geom_point(aes(color = carrier))+
  ggtitle("Departure delay across month of June between all Airlines.") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  plot.title = element_text(color="red", size=14, face="bold.italic", hjust = 0.5),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#1E1E20",
 face="bold.italic"),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



```
# Histogram plot for July
July <- subset(flt_1, month == 7)
July <- subset(July, !is.na(dep_delay), !is.na(arr_delay) )

ggplot(aes(x = day, y = dep_delay), data = July ) +
  geom_point(aes(color = carrier))+
  ggtitle("Departure delay across month of July between all Airlines.") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  plot.title = element_text(color="red", size=14, face="bold.italic", hjust = 0.5),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#1E1E20",
 face="bold.italic"),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



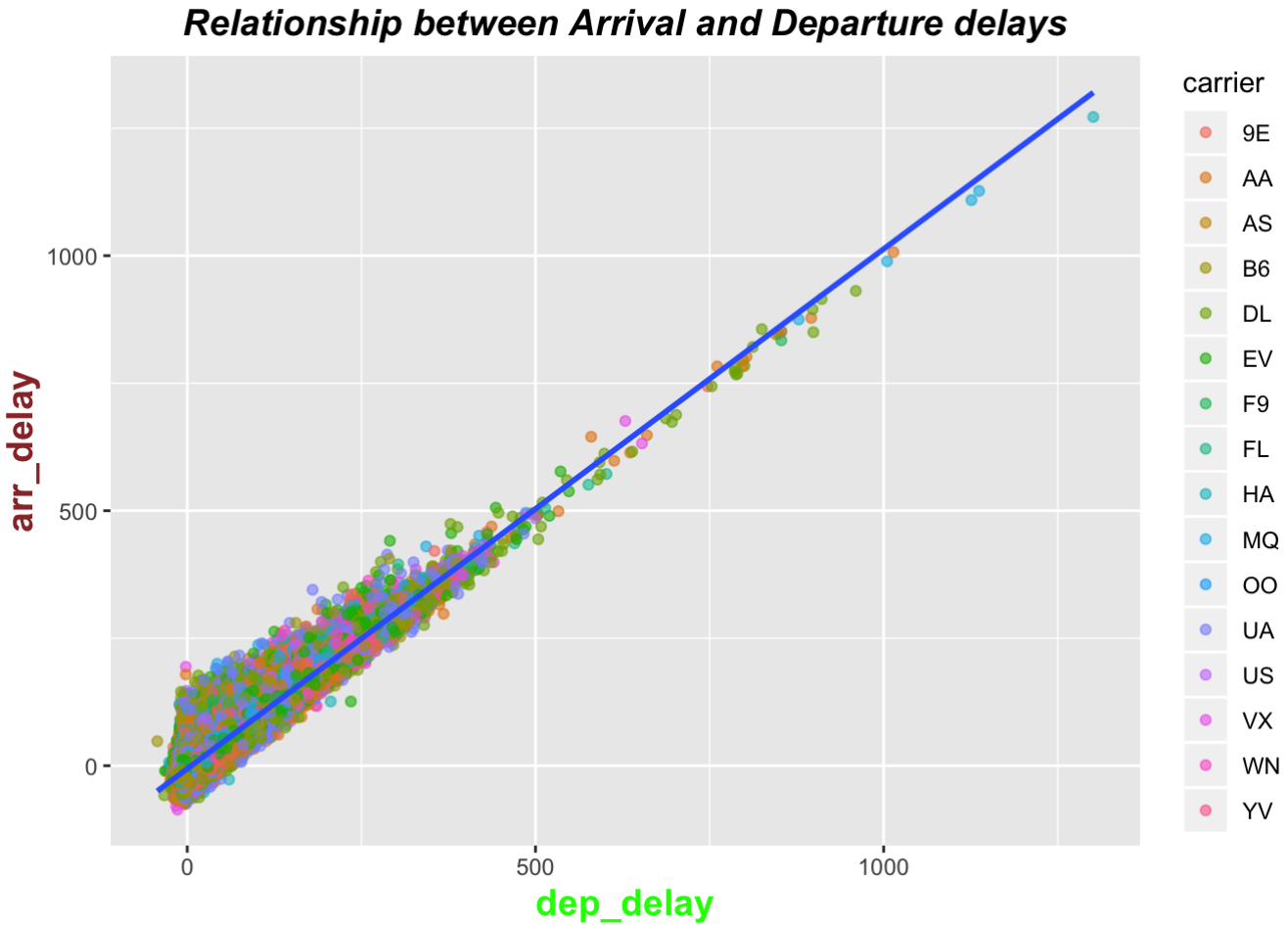
ANALYSIS

It is clear from both charts that delay was not specific in one day but across the month. Still, these datasets we have is not just enough to make any prediction or statements but after looking at the datasets we can surely say that between June and July there are huge delays.

Q10. Relationship between arr_delay and dep_delay.

Lets see if there is any relationship between arr_delay and dep_delay in our datasets.

```
ggplot(subset_flight,aes(x = dep_delay, y = arr_delay)) +
  geom_point(aes(color = carrier), alpha = 0.6)+
  geom_smooth(method= lm)+
  ggtitle("Relationship between Arrival and Departure delays") +
  theme(
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="green", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



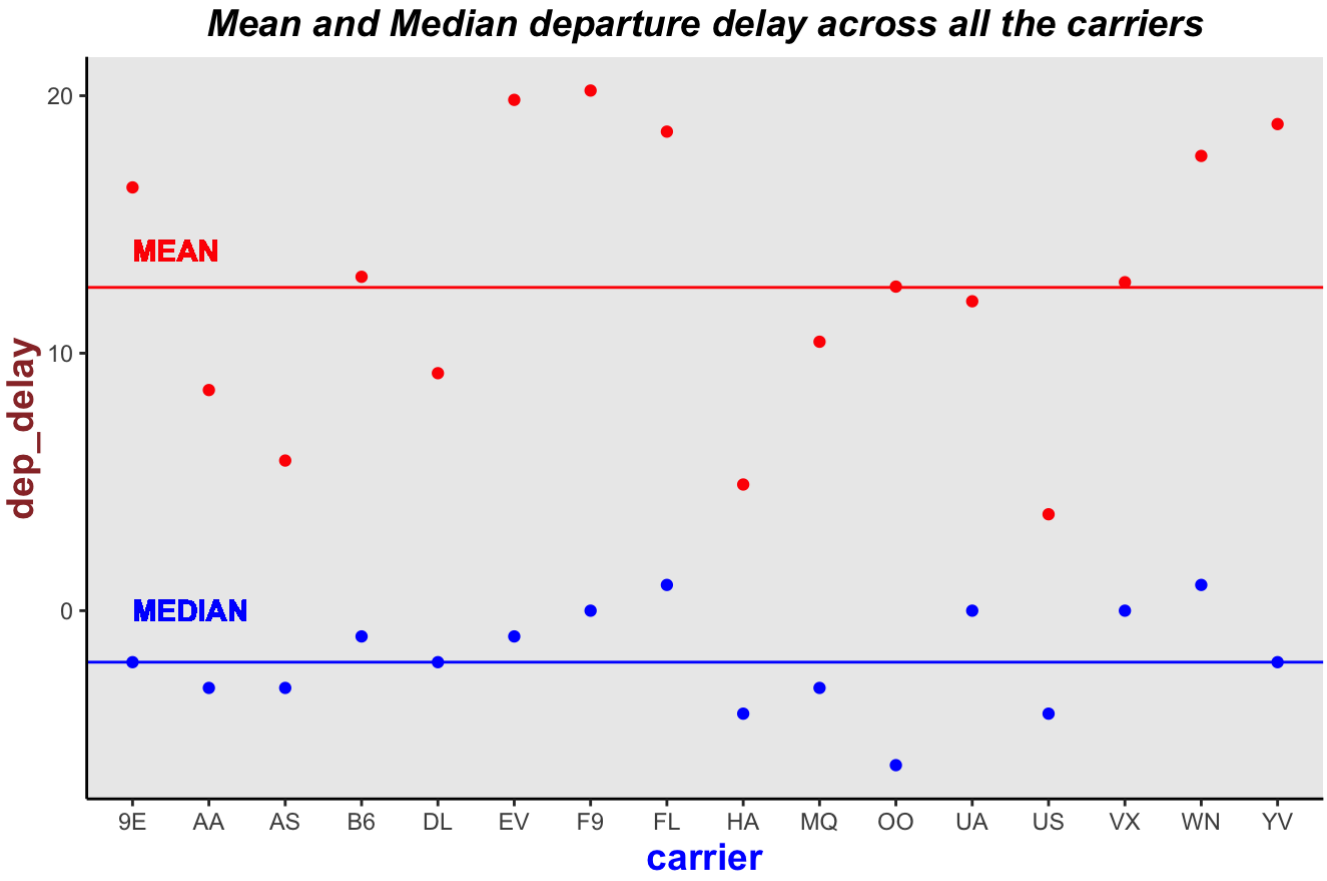
ANALYSIS

From the above plot, we can see there is a linear relationship between arrival delay and departure delays among all the carriers and across all the airports.

After quick and dirty analysis we will tweak our model a little bit and see if the delays are not marginal. Till now if the arrival and departure of the airlines are not on schedule then we are simply calling it delays but if we normalize our datasets we can make our analysis much better. So to do that let's calculate mean and median of departure delays across all the carriers in one plot.

Q11. Mean and Median departure delay across all the carriers

```
# It will take atleast 5 minutes to run this code. [ BE PATIENCE !!! ]

ggplot(subset_flight,aes(x = carrier, y = dep_delay))+
  geom_point(color = 'red', stat = 'summary', fun.y = mean) +  # MEAN
  geom_point(color = 'blue', stat = 'summary', fun.y = median)+ # MEDIAN
  geom_hline(aes(yintercept = mean(dep_delay, na.rm = TRUE)), color = 'red')+
  geom_hline(aes(yintercept = median(dep_delay, na.rm = TRUE)), color = 'blue')+
  ggtitle("Mean and Median departure delay across all the carriers") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  geom_text(aes (x = 1, y = 14), label = "MEAN", hjust = 0, color = 'red', fontface = "b
old")+
  geom_text(aes (x = 1, y = 0), label = "MEDIAN", hjust = 0, color = 'blue', fontface =
"bold")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

ANALYSIS

After looking at the variance of the datasets we can see that there is quite a difference between Mean and Median across all the airlines.

Lets calculate mean delay:

```
mean(flt_1$dep_delay, na.rm = TRUE)
```

```
## [1] 12.55516
```

What is our cut off point. What should be consider late or early delays

We can pick based on average mean which cames out to be 12.55 as a baseline. As it is not standard anything beyond its scheduled arrival is delayed the flight. But in our datasets, we can wiggle a little bit to pick one cut offline and we can bin the flights into respective variables, I think this would be the way to do it.

In Summary, we will Create a categorical variable for departure delay (dep_delay), which we consider to be departure status. Departure delay is a continuous variable capturing the difference in minutes between the expected and actual departure times. The new variable which we generated classifies the delay into 3 discrete levels: Early (up to 0 minutes of delay), On time (up to 13 minutes of delay), and Late (above 13 minutes of delay).

```
flt_1$dep_status <- flt_1$dep_delay # slicing dep_delay to dep_status. I can do mutate.
flt_1$dep_status<- ifelse(flt_1$dep_status < 0,'Early', # Recoding
                          ifelse(flt_1$dep_status < 13,'On Time','Late'))
flt_1$dep_status<-factor(flt_1$dep_status) # type conversion
table(flt_1$dep_status) # Better if we have prop table.
```

```
##
##   Early    Late On Time
##  183135   77076   67135
```

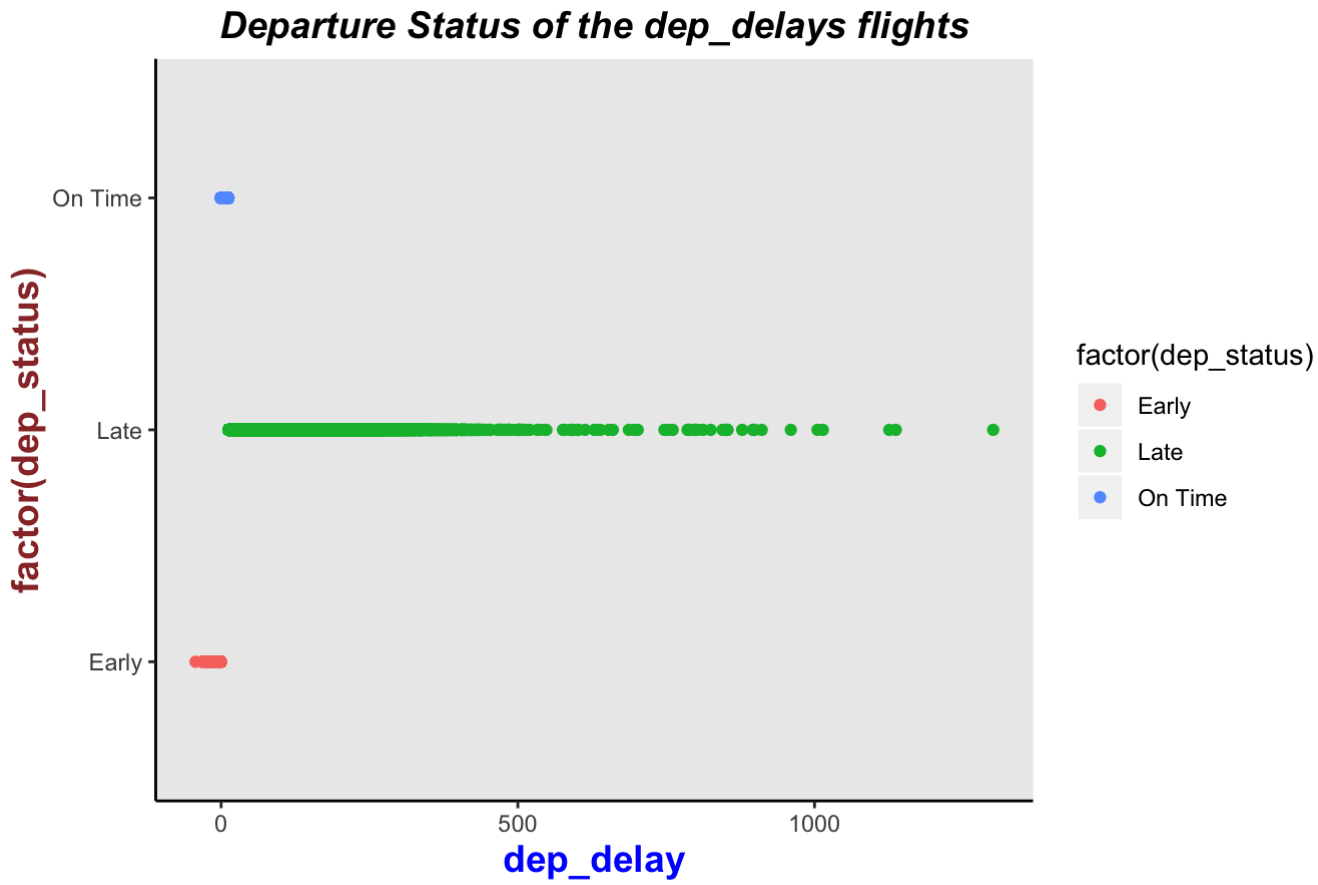
ANALYSIS

There were more early flights compare to Late and On- time flights. From the above table, we can say that must flights seems to be early if they have an early departure. We will dive into Late flights further down but we can see that when we change our baseline we see the huge improvement in Early and On-time departure delay airlines.

Q12. Spotting the Outliers in Late Flights.

The graph below was generated to ensure that the new **gexi ksvgep** variable matches the original **jegxsv** variable:**Also we can visually spot some outliers in late arrival column which are more than 1000 Minutes** We will diagnose all those variables later to see the casual inferences. We can see that most of the late flights are dense between 0 to 500 Minutes and then it starts to disperse and then we see some heavy outliers around 1300 Minutes.

```
ggplot(flt_1, aes(x=dep_delay, y=factor(dep_status))) +
  geom_point(aes(color=factor(dep_status)))+
  ggtitle("Departure Status of the dep_delays flights")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),  color="#D70026",
face="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

Let's look at one more perspective if flights are arriving in the airport on time. We can take mean to get the sense of the data.

```
mean(flt_1$arr_delay, na.rm = TRUE)
```

```
## [1] 6.895377
```

Similarly, as we wiggle our baseline for departure delays we can do similarly for the arrival delays. We will create a categorical variable (arrival status) for arrival delay (arr_delay) with levels: Early, On time, and Late. In creating this variable, we followed the same approach as that used for the departure delay variable above.

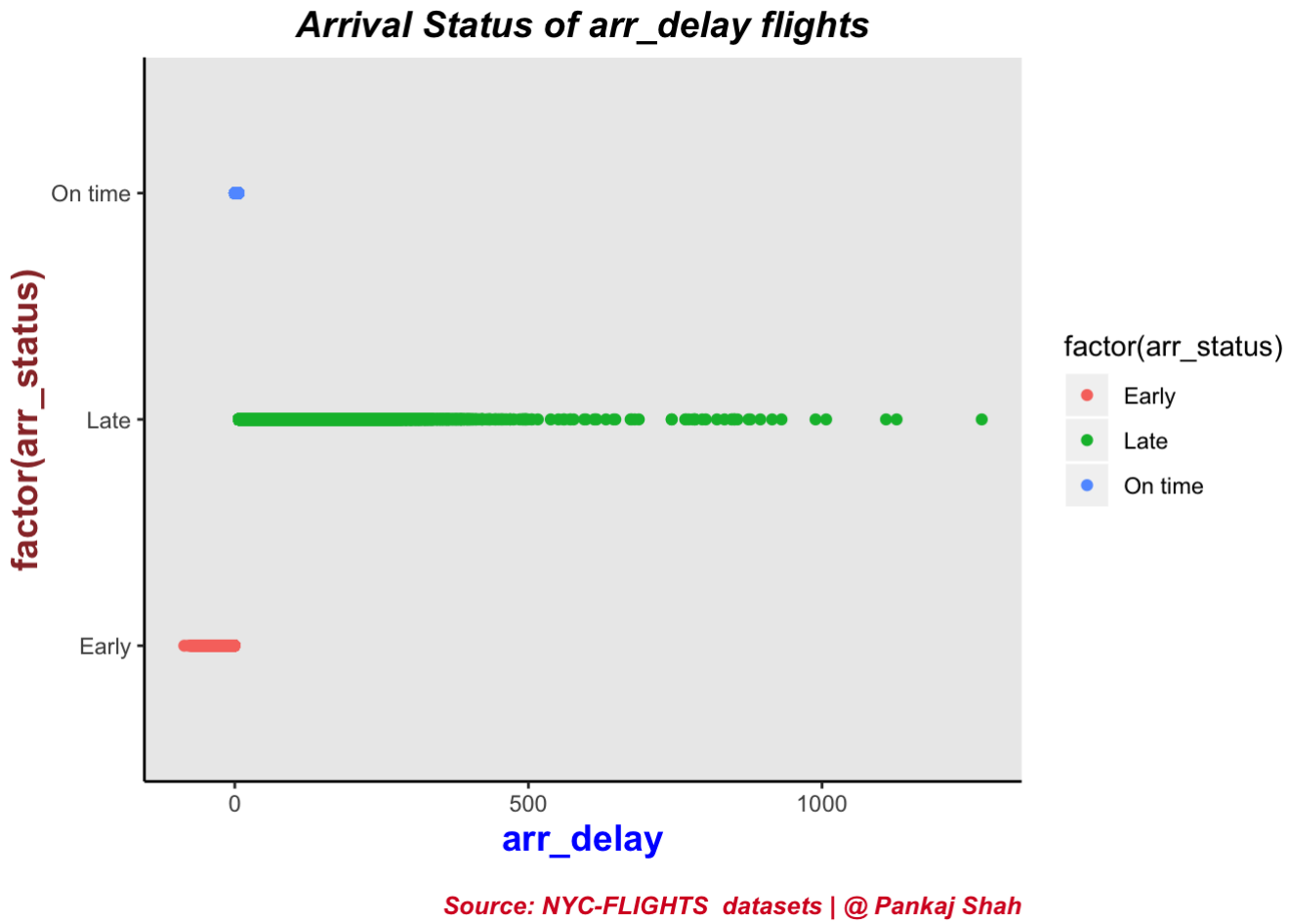
```
flt_1$arr_status <- flt_1$arr_delay
flt_1$arr_status<- ifelse(flt_1$arr_status < 0,'Early',
                          ifelse(flt_1$arr_status < 6.9,'On time','Late')) # taking Mean
 as On_time baseline.
flt_1$arr_status<-factor(flt_1$arr_status)
table(flt_1$arr_status) # prop.table to do analysis.
```

```
##
##   Early    Late On time
##  188933  105827   32586
```

ANALYSIS

The graph below was generated to ensure that the new **gexi ksvrgep** variable matches the original **jegxsv** variable as we did above for the arrival. Keeping it consistent. We can see a small fraction of planes arrive on time, there are more planes that arrive early.

```
ggplot(data=flt_1, aes(x=arr_delay,
                       y=factor(arr_status))) +
  geom_point(aes(color=factor(arr_status)))+
  ggtitle("Arrival Status of arr_delay flights")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Q13. Variable Summaries

A statistical summary of all the variables within our dataset was generated below using the 'summary' command in R:

```
summary(flt_1)
```

```
##       year          month             day           dep_time
##  Min.   :2013   8      : 28756   Min.   : 1.00   Min.   :   1
##  1st Qu.:2013   10     : 28618   1st Qu.: 8.00   1st Qu.: 907
##  sched_dep_time   dep_delay          arr_time    sched_arr_time
##  Min.   : 500   Min.   : -43.00   Min.   :   1   Min.   :   1
##  1st Qu.: 905   1st Qu.:  -5.00   1st Qu.:1104   1st Qu.:1122
##    arr_delay          carrier              flight        tailnum
##  Min.   : -86.000   Length:327346      Min.   :   1   N725MQ :   544
##  1st Qu.: -17.000   Class :character   1st Qu.: 544   N722MQ :   485
##  origin           dest              air_time        distance
##  EWR:117127   Length:327346      Min.   : 20.0   Min.   :  80
##  JFK:109079   Class :character   1st Qu.: 82.0   1st Qu.: 509
##       hour           minute        time_hour
##  Min.   : 5.00   Min.   : 0.00   Min.   :2013-01-01 05:00:00
##  1st Qu.: 9.00   1st Qu.: 8.00   1st Qu.:2013-04-05 06:00:00
##                    carrier_name
##  United Air Lines Inc.   :57782
##  JetBlue Airways         :54049
##                           arrival_airport       lat.x
##  Hartsfield Jackson Atlanta Intl   : 16837   Min.   :21.32
##  Chicago Ohare Intl                : 16566   1st Qu.:32.90
##      lon.x             alt.x             tz.x            dst.x
##  Min.   :-157.92   Min.   :   3.0   Min.   :-10.000   Length:327346
##  1st Qu.: -95.34   1st Qu.:  26.0   1st Qu.: -6.000   Class :character
##    tzone.x                    departure_airport      lat.y
##  Length:327346      John F Kennedy Intl:109079   Min.   :40.64
##  Class :character   La Guardia         :101140   1st Qu.:40.64
##      lon.y            alt.y            tz.y       dst.y
##  Min.   :-74.17   Min.   :13.00   Min.   :-5   Length:327346
##  1st Qu.:-74.17   1st Qu.:13.00   1st Qu.:-5   Class :character
##    tzone.y             Season            dep_status       arr_status
##  Length:327346      Length:327346      Early  :183135   Early  :188933
##  Class :character   Class :character   Late   : 77076   Late   :105827
##  [ reached getOption("max.print") -- omitted 5 rows ]
```

While the information above provides a high-level introduction to the data set, the sections below shall focus on specific analyses of the 'flights' data set.

Q14. Analysis by flight volume:

A high-level view of the flight volumes coming out of the NYC area is displayed below:

```
print("Number of flights flown away from given three airports:")
```

```
## [1] "Number of flights flown away from given three airports:"
```

```
sort(xtabs(formula = ~ departure_airport, data = flt_1), decreasing = TRUE)
```

```
## departure_airport
## Newark Liberty Intl John F Kennedy Intl          La Guardia
##              117127              109079               101140
```

ANALYSIS

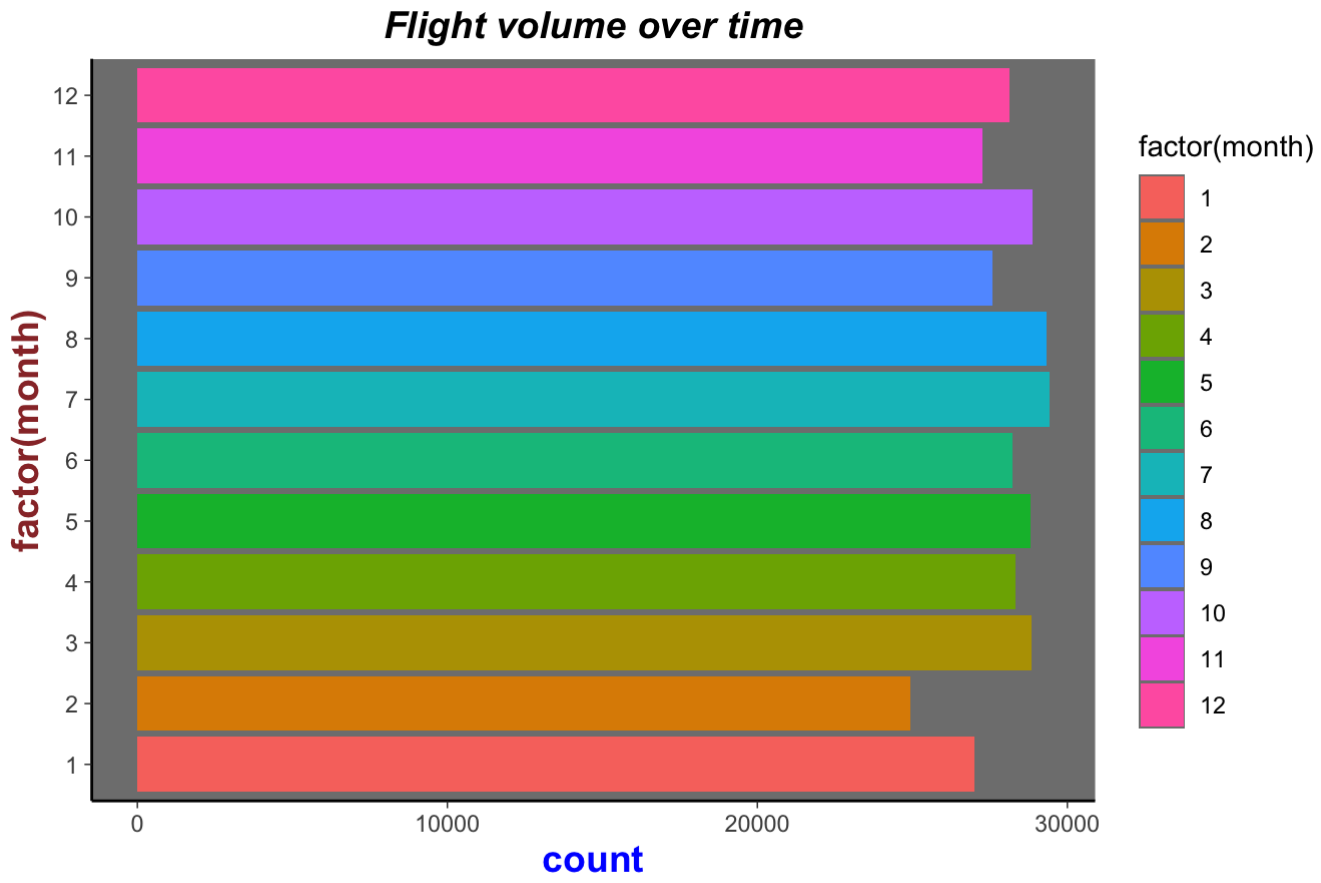
It seems like an almost equal number of flights flew away from all these three airports as there are small marginal changes between these three airports. As La Guardia airport seems to have fewer flights than other two maybe because of being domestic Airport compare to two other International Airports with international flights.

While the three airports manage similar volumes, Newark Liberty international airport is in the lead. We shall now drill down into the volume data, looking at flight traffic from different perspectives.

Q15. Flight volume over time

The flight count by month is displayed below:

```
ggplot(flights) +
     aes(x=factor(month)) +
     geom_bar(aes(fill=factor(month))) +
     scale_colour_brewer(palette = "Set1")+
     theme_dark()+
  ggtitle("Flight volume over time")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))+
  coord_flip()
```



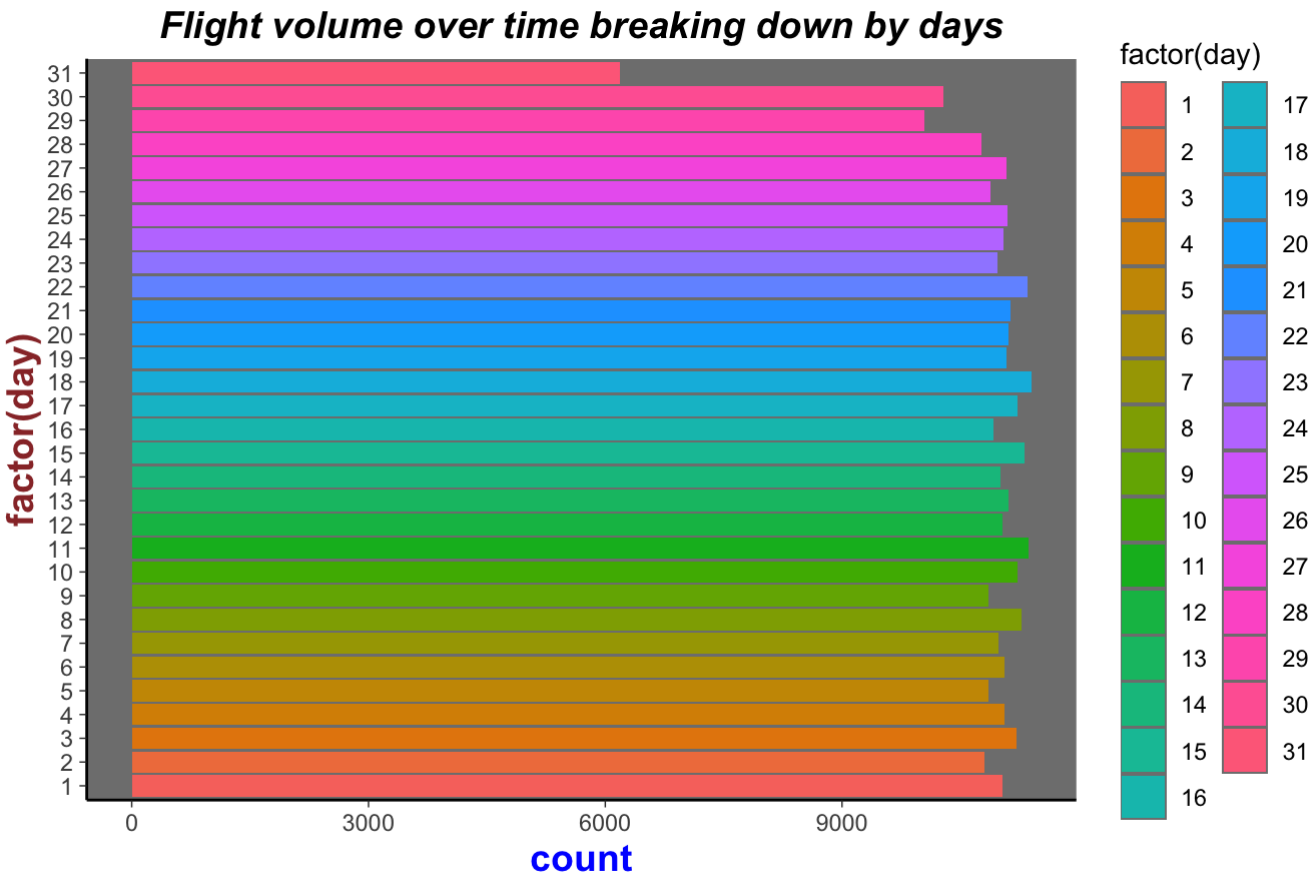
Source: NYC-FLIGHTS datasets | @ Pankaj Shah

ANALYSIS

What we observe is flights throughout the year remains consistent with a little dip in the month of February(2nd) which I believe because of Northeaster Weather. We will dig into that components later when we once merge our datasets with Weather data and see the number of cancellation in each month of the year and by airlines. Most of the flights are during the summer season where the cancellation due to weather seems less. But that is another can of worms which we will dig into it later on.

The output shows that the number of flights each month appears to be consistent and closely follows the number of days in each month. It indicates that the number of flights stays consistently within days. Our finding will be underscored by the output of counting a number of flights within days (ranging from 1 to 31) which is shown below:

```
ggplot(flights) +
  aes(x=factor(day)) +
  geom_bar(aes(fill=factor(day)))+
  scale_colour_brewer(palette = "Set1")+
  theme_dark()+
  ggtitle("Flight volume over time breaking down by days")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))+
  coord_flip()
```



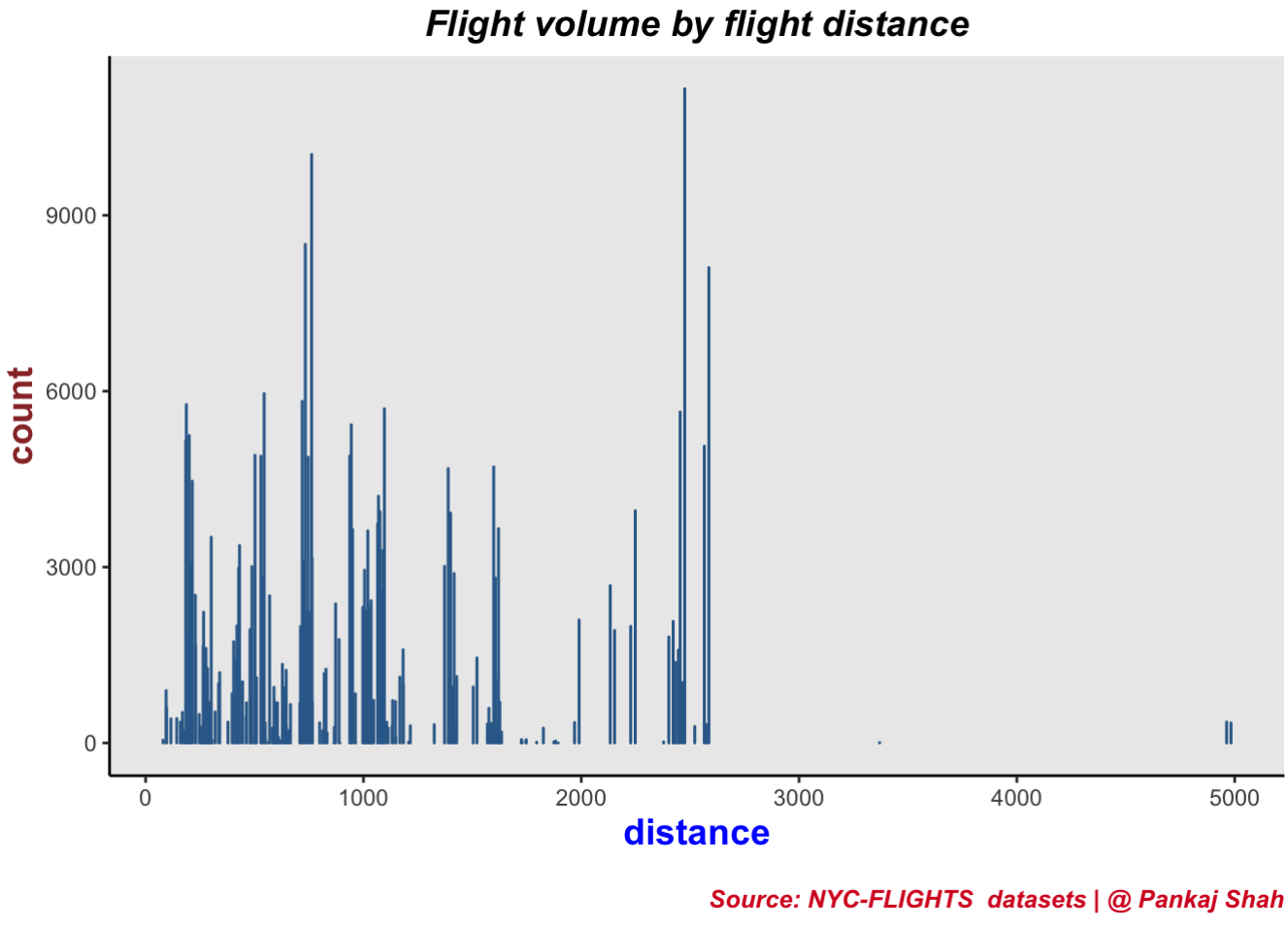
ANALYSIS

29th and 30th seem less flown date. 31st makes total sense as we can have 31st in calendar only half of time. Also, We don't have a dataset of our tickets so it will be a little hard to draw a conclusion just based on these bar plots. But in the ideological world, if there was an equal number of flights offered from all these three airports

across the given days. People tend to fly less towards the end of the month but by not much. If we pay close attention we can see that from 23 it starts to dip but these could be totally due to the availability of flights and other unseen factors. # Q16.Flight volume by flight distance.

The following bar chart shows the number of flights grouped by flying distance, which helps indicate the type of flights departing from NYC(Short-distance domestic, Long-distance domestic, International).

```
ggplot(flt_1, aes(x=distance)) +
  geom_bar(aes(color=1)) + guides(color=FALSE)+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  ggtitle("Flight volume by flight distance")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



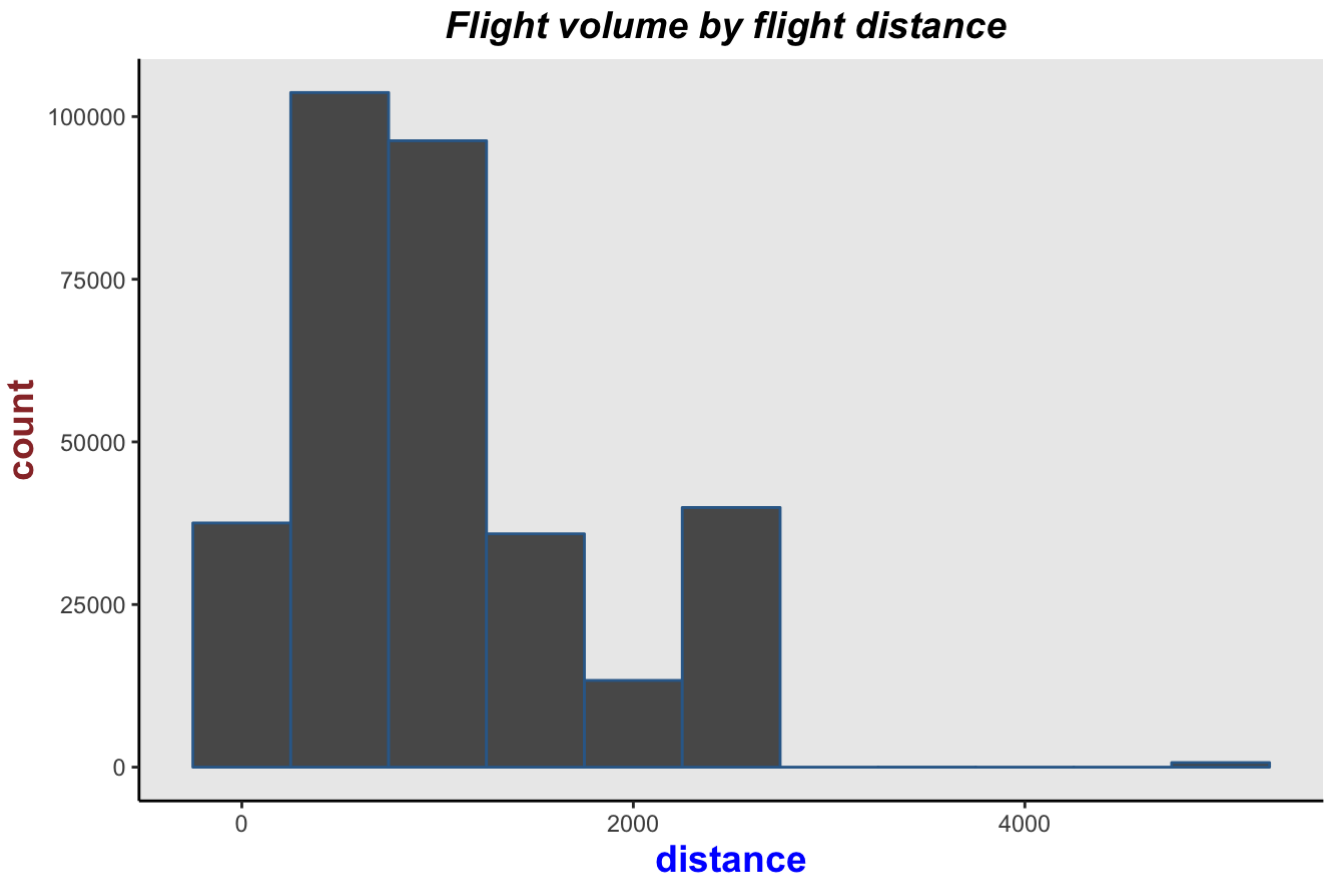
As we can see most of the flights are from range 0 to 2800 Miles. There are some long-haul flights around 5000 Miles. We can bin the flights based on good model assumption as short, mid-distance and long-haul flight category. To do that let's take the summary of our flight datasets.

```
summary(flt_1$distance, na.rm = TRUE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      80     509     888    1048    1389    4983
```

```
# table(flt_1$distance) # Uncomment to see all the flights for specific distance.
```

```
ggplot(flt_1, aes(x=distance)) +
  geom_histogram(aes(color=1), binwidth = 500) + guides(color=FALSE) +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  ggtitle("Flight volume by flight distance ")+ # Using binwidth to split the flights in
to short,mid & long.
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

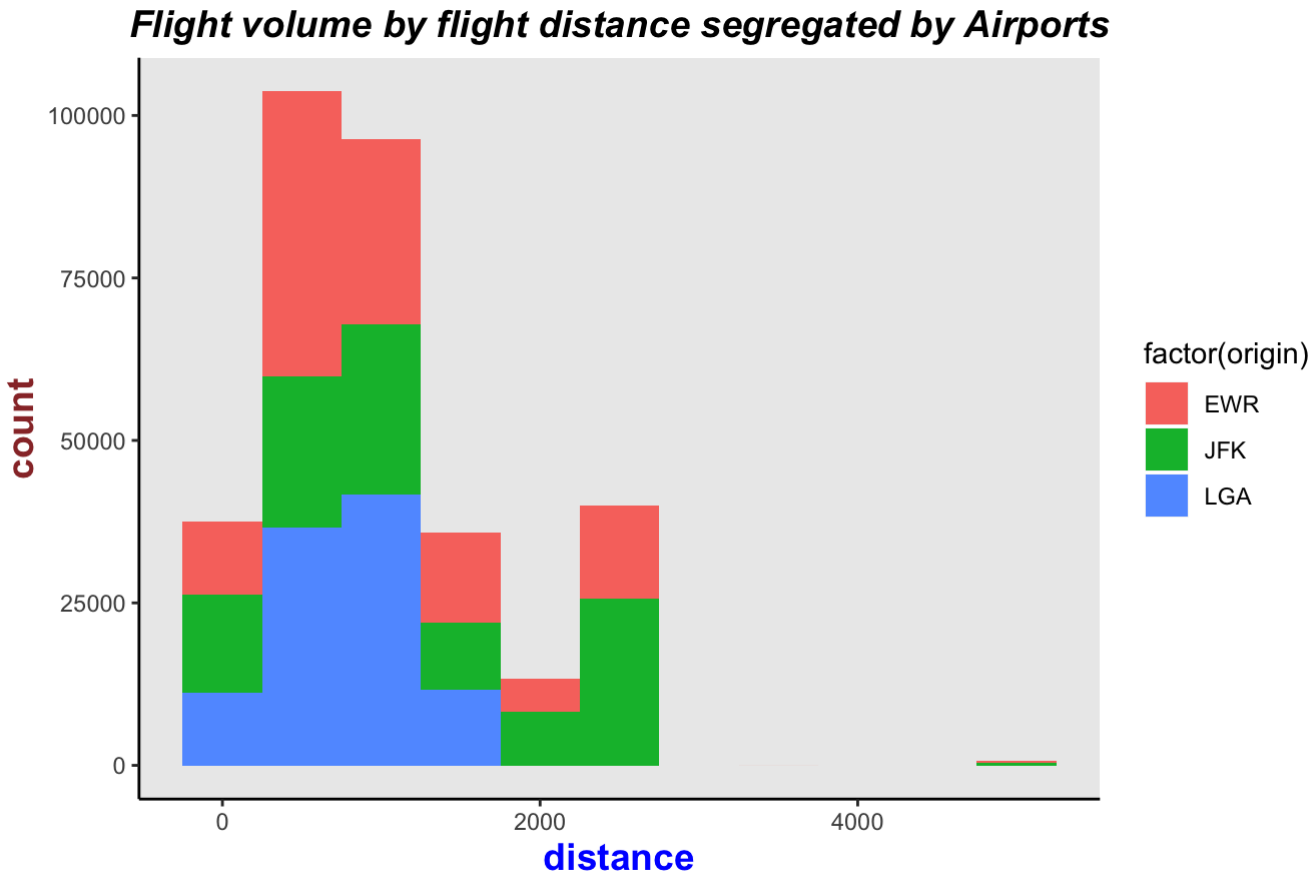
ANALYSIS

More flights were flown 2nd and 3rd bin. We also see some flights that were flown for more than 4000 Miles(Outliers). These most be international flights or long haul flights. The distance between East coast to West Coast is around 2500-3000 Miles. It must have been flights from New York to Hawaii which is around 5000 Miles.

We set 500 miles as the bin width for this report. The output shows that the highest proportion of flights have distances less than 1500 miles. It indicates that most of the flights are traveling short or mid-distance domestic routes. There's also a considerable number of flights with distances around 1500-2500 miles. It indicates the second most common group of flights is long-distance domestic flights / short distance international flights (The flight distance from NYC to Los Angeles is 2500 miles). Last but not least, there's a little number of flights fall into the range from 4500 to 5000 miles, which indicates that the long-distance international flights only count for a small proportion of the total flights departing from NYC. Generating the same chart, while taking the different departure airports into account, results in the following output:

```
ggplot(flt_1, aes(x=distance)) +
geom_bar(aes(fill=factor(origin)), binwidth = 500) +
  guides(color=FALSE)+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  ggtitle("Flight volume by flight distance segregated by Airports")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

```
## Warning: `geom_bar()` no longer has a `binwidth` parameter. Please use
## `geom_histogram()` instead.
```



ANALYSIS

- EWR : RED

- JFK : GREEN
- LGA : BLUE

A higher proportion of short and mid-distance flights. A very small proportion of long distance but hard to visualize as they stack on top of one another. LGA is not international terminal.LaGuardia is the busiest airport in the United States without any non-stop service to Europe. Laguardia doesn't have any International arrivals because the runways are shorts.

Q17. Flight volume by carrier

Below is a table showing flight volume by departure airports and carrier airlines.

```
xtabs(formula = ~ carrier_name + departure_airport, data = flt_1)
```

```
##                              departure_airport
## carrier_name                  John F Kennedy Intl La Guardia
##   AirTran Airways Corporation                   0       3175
##   Alaska Airlines Inc.                          0          0
##   American Airlines Inc.                    13600      14984
##   Delta Air Lines Inc.                      20559      22804
##   Endeavor Air Inc.                         13742       2359
##   Envoy Air                                  6838      16102
##   ExpressJet Airlines Inc.                   1326       8225
##   Frontier Airlines Inc.                        0        681
##   Hawaiian Airlines Inc.                      342          0
##   JetBlue Airways                           41666       5911
##   Mesa Airlines Inc.                            0        544
##   SkyWest Airlines Inc.                         0         23
##   Southwest Airlines Co.                        0       5988
##   United Air Lines Inc.                      4478       7803
##   US Airways Inc.                            2964      12541
##   Virgin America                             3564          0
##                              departure_airport
## carrier_name                  Newark Liberty Intl
##   AirTran Airways Corporation                   0
##   Alaska Airlines Inc.                        709
##   American Airlines Inc.                     3363
##   Delta Air Lines Inc.                       4295
##   Endeavor Air Inc.                          1193
##   Envoy Air                                  2097
##   ExpressJet Airlines Inc.                  41557
##   Frontier Airlines Inc.                        0
##   Hawaiian Airlines Inc.                        0
##   JetBlue Airways                            6472
##   Mesa Airlines Inc.                            0
##   SkyWest Airlines Inc.                         6
##   Southwest Airlines Co.                     6056
##   United Air Lines Inc.                     45501
##   US Airways Inc.                            4326
##   Virgin America                             1552
```

** Observations ** - Airtran, Mesa ,Frontier only fly from LaGuardia.
- Alaska only flew from Newark Liberty Intl
- Hawaiian only flew from JFK.
- Skywest only flew 6 times through newark that is the lowest and 23 from LGA both are lowest.
- United airways flew most 45,501 planes from Newark Liberty Intl

- Frequencies of airports for each airline companies to depart from.
- UA (United Airlines) is the biggest number of airlines in 2013.
- Lots of flight departed from Newark Liberty International Airport (EWR).

Also remember JFK and Newark are International terminal whereas LGA is domestic airport

```
colnames(flt_1)
```

```
##  [1] "year"              "month"             "day"
##  [4] "dep_time"          "sched_dep_time"    "dep_delay"
##  [7] "arr_time"          "sched_arr_time"    "arr_delay"
## [10] "carrier"           "flight"            "tailnum"
## [13] "origin"            "dest"              "air_time"
## [16] "distance"          "hour"              "minute"
## [19] "time_hour"         "carrier_name"      "arrival_airport"
## [22] "lat.x"             "lon.x"             "alt.x"
## [25] "tz.x"              "dst.x"             "tzone.x"
## [28] "departure_airport" "lat.y"             "lon.y"
## [31] "alt.y"             "tz.y"              "dst.y"
## [34] "tzone.y"           "Season"            "dep_status"
## [37] "arr_status"
```

Q18. Flight volume by destination

The table below shows the top 10 destination airports for flights departing from the NYC area:

```
flt_by_dest <- flt_1 %>%
                  group_by(arrival_airport) %>%
                     summarise(dest_count = n()) %>%
                     arrange(desc(dest_count)) %>% top_n(10)
```

```
## Selecting by dest_count
```

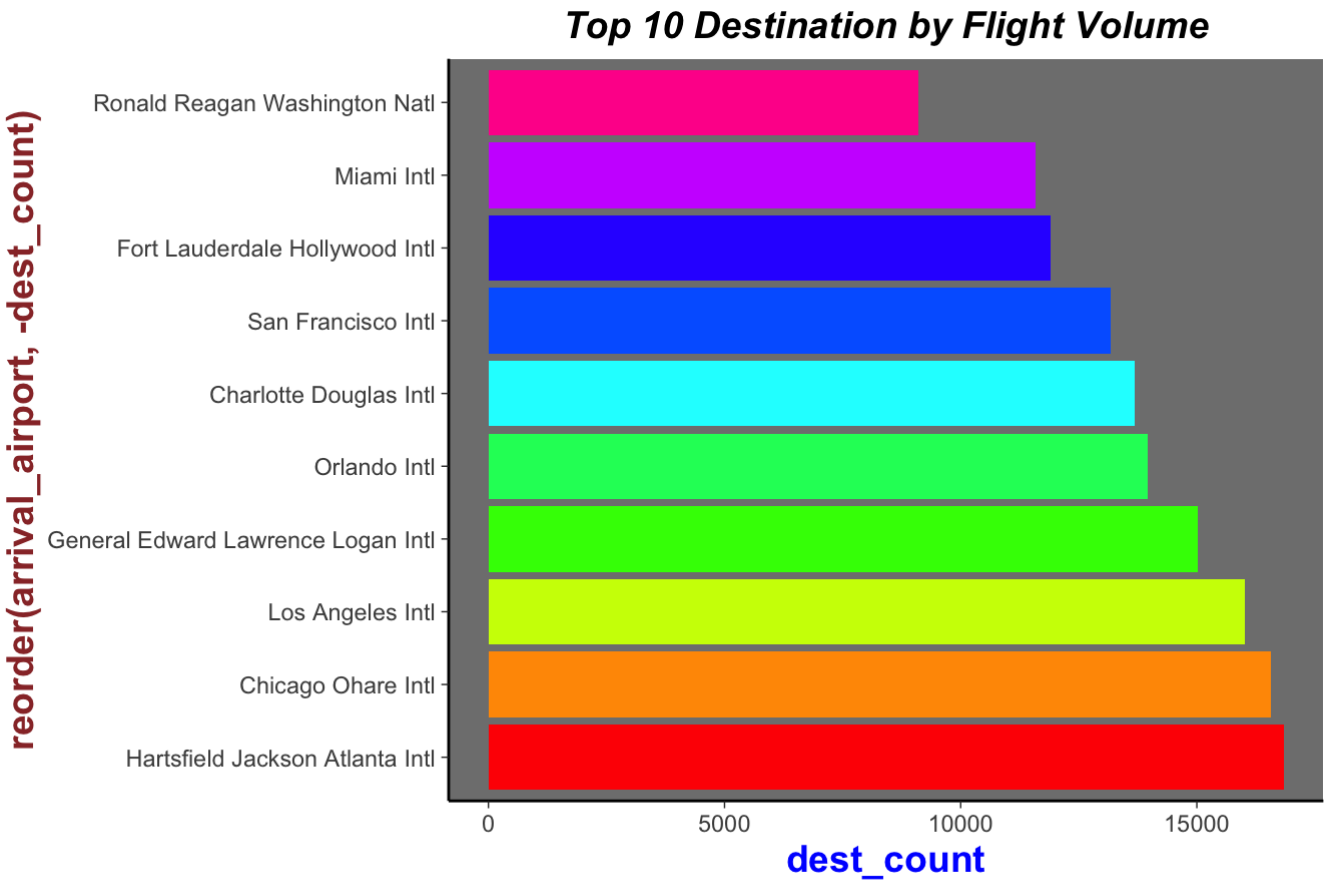
```
flt_by_dest
```

arrival_airport <fctr>	dest_count <int>
Hartsfield Jackson Atlanta Intl	16837
Chicago Ohare Intl	16566
Los Angeles Intl	16026

arrival_airport <fctr>	dest_count <int>
General Edward Lawrence Logan Intl	15022
Orlando Intl	13967
Charlotte Douglas Intl	13674
San Francisco Intl	13173
Fort Lauderdale Hollywood Intl	11897
Miami Intl	11593
Ronald Reagan Washington Natl	9111

1-10 of 10 rows

```
ggplot(flt_by_dest, aes(x=reorder(arrival_airport,-dest_count), y=dest_count)) +
  geom_bar(stat="identity", fill = rainbow(10)) +
  ggtitle ("Top 10 Destination by Flight Volume") +
  theme(legend.position="top") +
  theme_dark() +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))+
  coord_flip()
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

ANALYSIS

- ATL : Hartsfield Jackson Atlanta Intl : Hub connecting many other flights so it makes sense.
- ORD : Chicago Ohare Intl : Hub connecting to Lots of Midwest and the West coast flights so make sense.
- LAX : Los Angeles Intl : West Coast direct flight so make a sense.
- BOS : General Edward Lawrence Logan Intl
- MCO : Orlando Intl : Tourism Spot
- CLT : Charlotte Douglas Intl : Interesting why Charlotte.
- SFO : San Francisco Intl : West Coast flights same as people leaving for toursism. Direct flight is advantage.
- FLL : Fort Lauderdale Hollywood Intl : Florida Tourism spot
- MIA : Miami Intl : Tourism Spot
- DCA : Ronald Reagan Washington Natl : Tourism as well as political (Washigton DC)

Largest number of flights are on domestic routes (Boston being one of them, Chicago can be included but i might say it might be connecting hub to west coast as well as Midwest flights)

Q19. Flight volume by departure airport

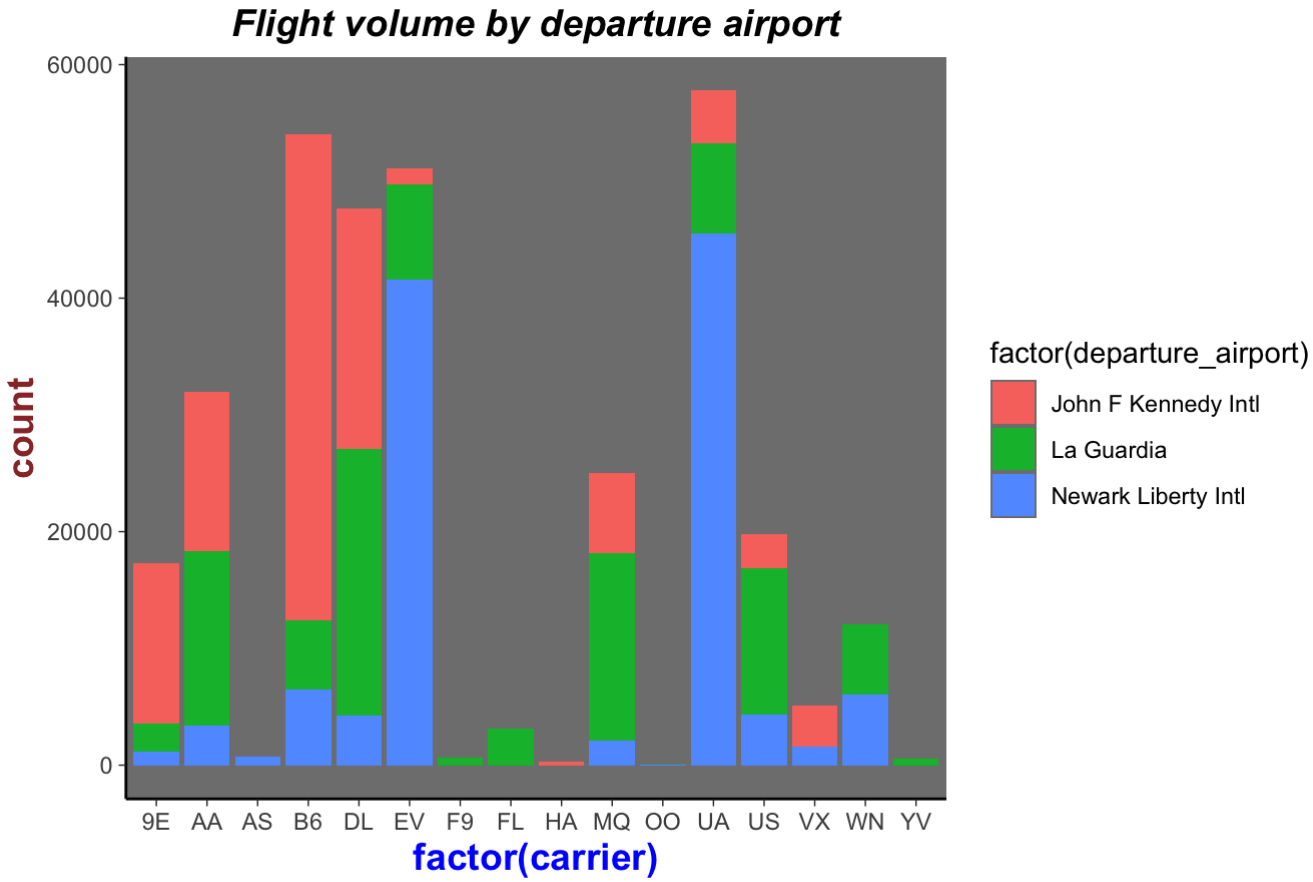
- Analysis by carrier

Below is a summary showing the relationship between departure airports and carrier airlines.

```
xtabs(formula =~ carrier_name + origin, data = flt_1)
```

```
##                             origin
## carrier_name                   EWR   JFK   LGA
##   AirTran Airways Corporation    0     0  3175
##   Alaska Airlines Inc.         709     0     0
##   American Airlines Inc.      3363 13600 14984
##   Delta Air Lines Inc.        4295 20559 22804
##   Endeavor Air Inc.           1193 13742  2359
##   Envoy Air                   2097  6838 16102
##   ExpressJet Airlines Inc.   41557  1326  8225
##   Frontier Airlines Inc.         0     0   681
##   Hawaiian Airlines Inc.         0   342     0
##   JetBlue Airways             6472 41666  5911
##   Mesa Airlines Inc.             0     0   544
##   SkyWest Airlines Inc.          6     0    23
##   Southwest Airlines Co.      6056     0  5988
##   United Air Lines Inc.      45501  4478  7803
##   US Airways Inc.             4326  2964 12541
##   Virgin America              1552  3564     0
```

```
ggplot(flt_1,
  aes(x=factor(carrier))) +
  geom_bar(aes(fill= factor(departure_airport)))+
  ggtitle ("Flight volume by departure airport") +
  theme(legend.position="top") +
  theme_dark() +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

ANALYSIS

It coorelates to our findings as well where the flight volume by flight destination that we observed above. - Newark is Hub for UA as well as EV. - HA only flew to JFK. - F9, FL only to La Guardia.

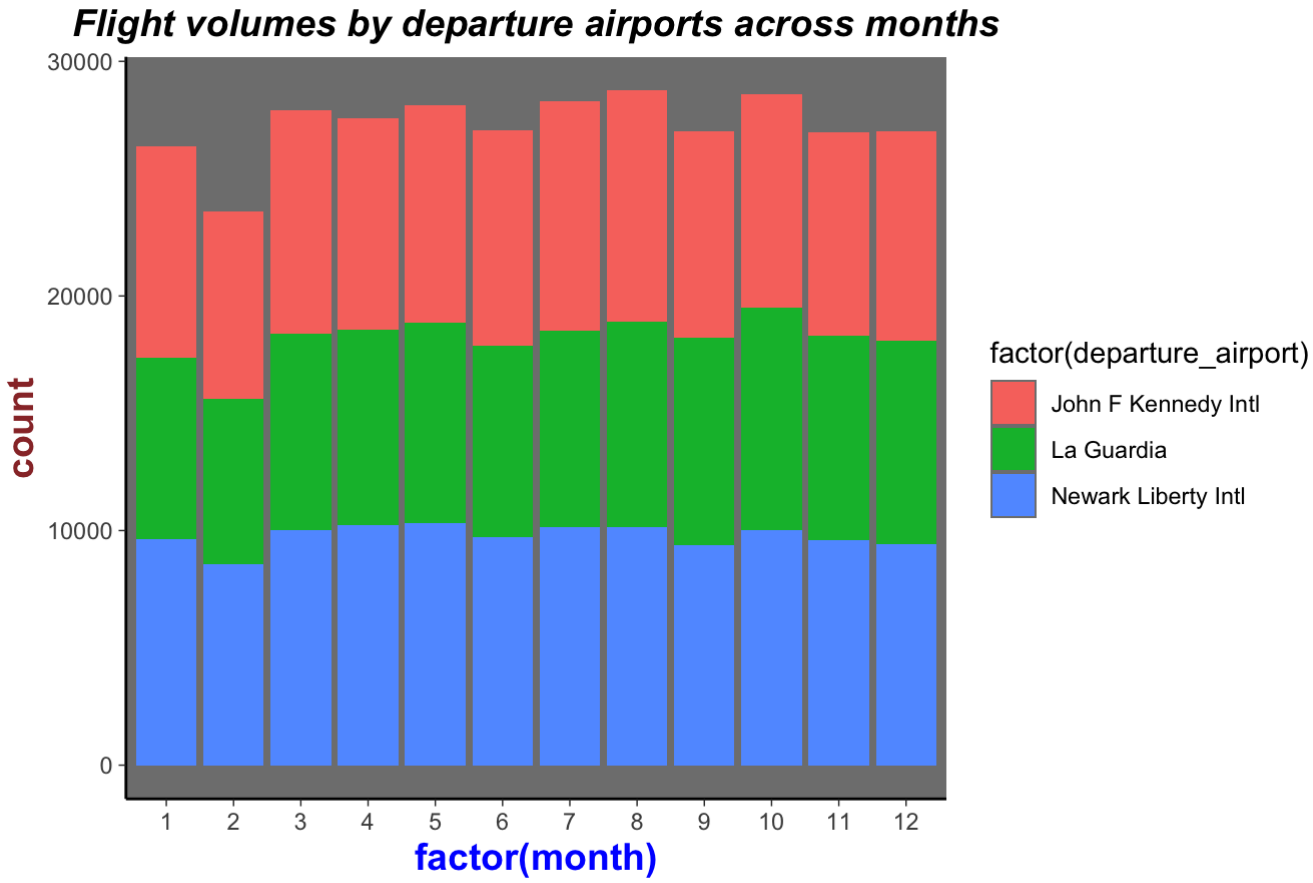
UA has departed the biggest number of airlines in 2013. And major flights departed from EWR

Q20. Summary showing the flight volumes by departure airports across months:

```
xtabs(formula = ~ month + departure_airport, data = flt_1)
```

```
##      departure_airport
## month John F Kennedy Intl La Guardia Newark Liberty Intl
##    1                 9031       7751                9616
##    2                 8007       7029                8575
##    3                 9497       8390               10015
##    4                 9013       8320               10231
##    5                 9270       8555               10303
##    6                 9182       8157                9736
##    7                 9757       8410               10126
##    8                 9870       8742               10144
##    9                 8788       8860                9362
##    10                9096       9516               10006
##    11                8645       8723                9603
##    12                8923       8687                9410
```

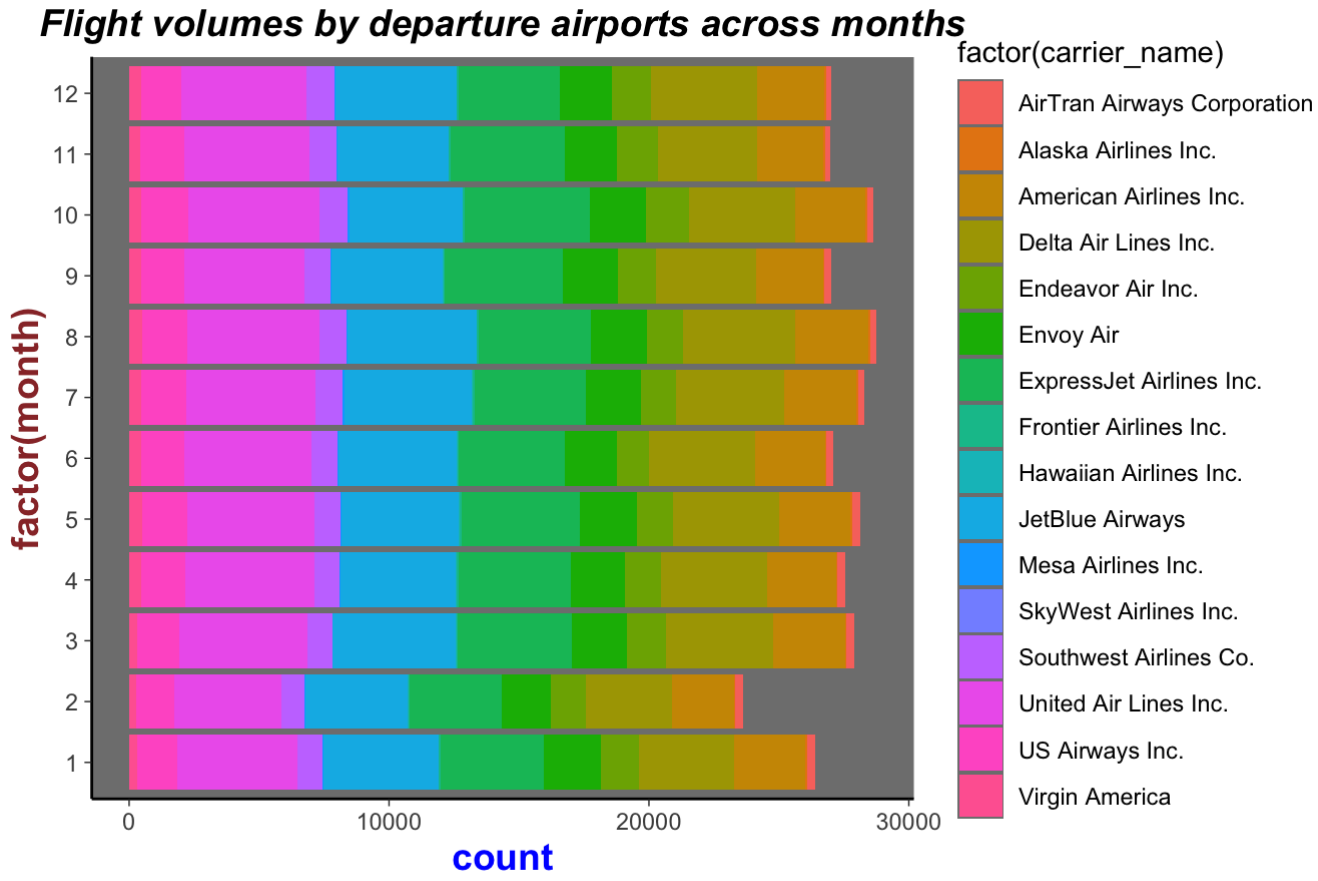
```
ggplot(flt_1) +
  aes(x=factor(month))+
  geom_bar(aes(fill= factor(departure_airport)))+
  ggtitle ("Flight volumes by departure airports across months") +
  theme(legend.position="top") +
  theme_dark() +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



ANALYSIS

The output indicates that the flight volumes among different airports by month generally remain consistent, however, La Guardia seems to have increased its share of traffic over the other airports in the second half of the year.

```
ggplot(data=flt_1,
       aes(x= factor(month))) +
  geom_bar( aes(fill= factor(carrier_name)))+
  ggtitle ("Flight volumes by departure airports across months") +
  theme(legend.position="top") +
  theme_dark() +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))+
  coord_flip()
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

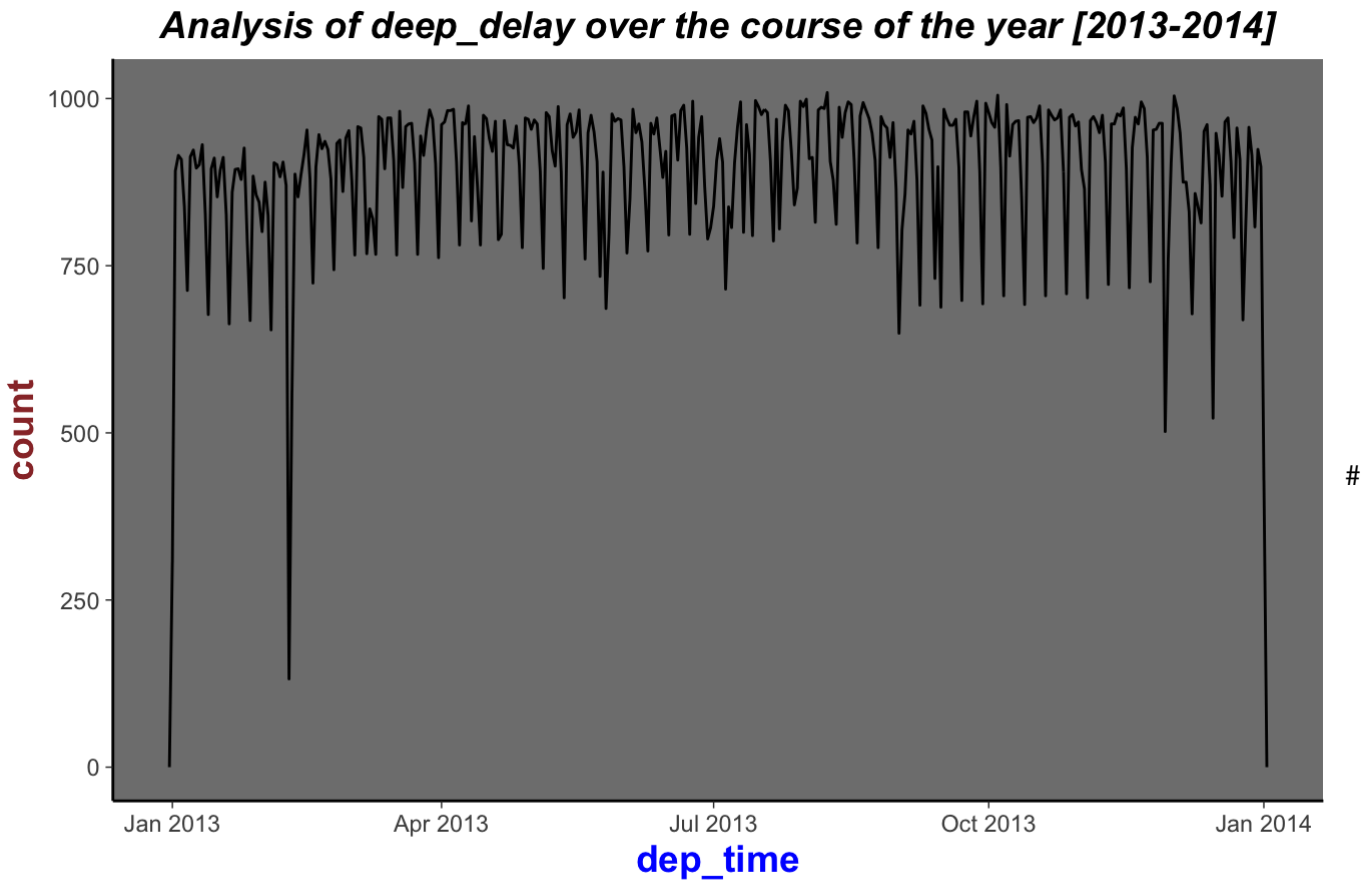
ANALYSIS

As we see in the analysis that the carrier spread is consistent across month and days. We can see

- variability in the number of Flights among different airlines
- But No clear difference in terms of the proportion of flights across the months.

Q21. Overall Analysis of deep_delay over the course of the Year [2013-2014].

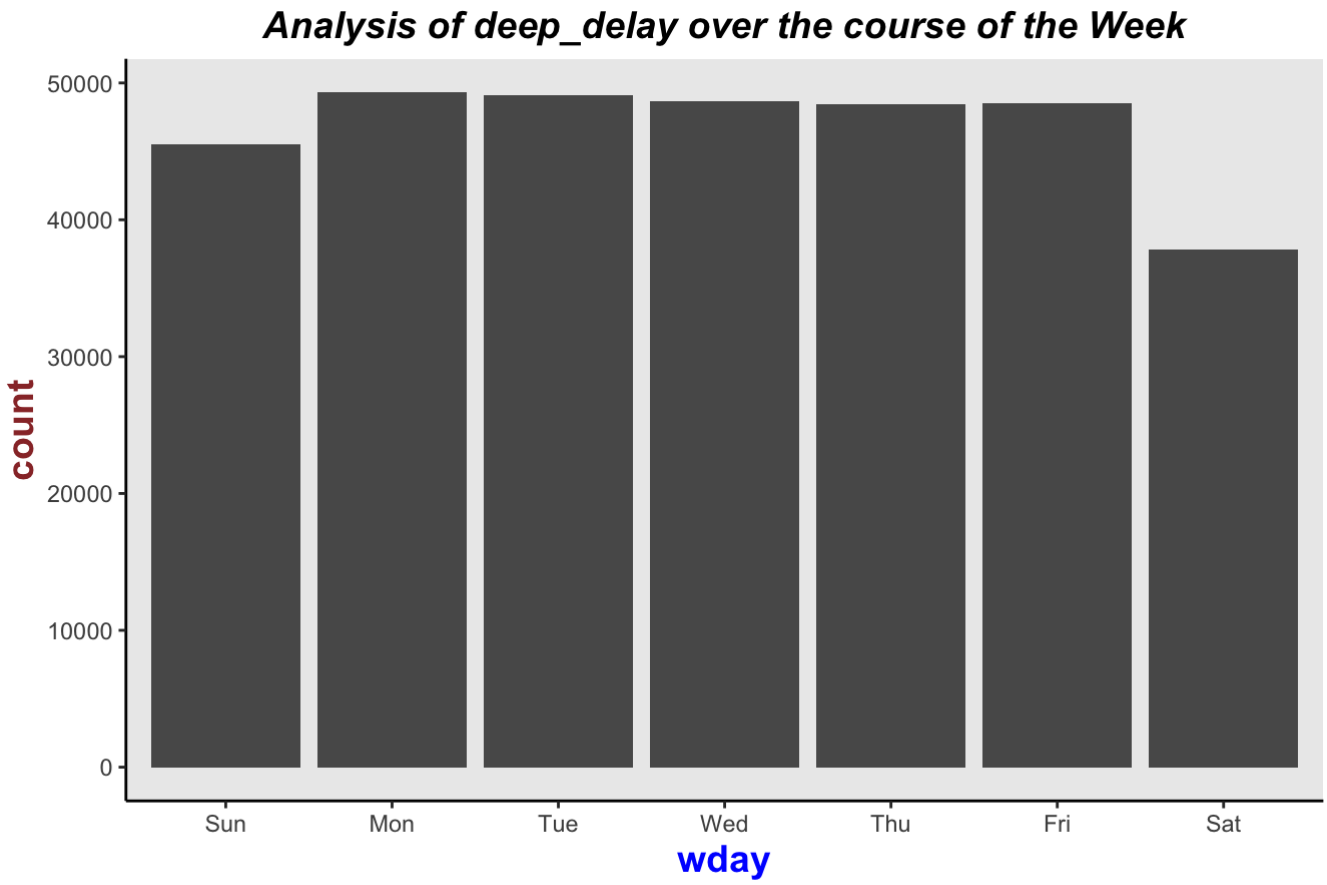
```
# Write a function to extract using make_datetime.
datetime_func <- function(year, month, day, time) {
  make_datetime(year, month, day, time %/% 100, time %% 100)
}

# Filter
flights_dt <- flt_1 %>%
  filter(!is.na(dep_time), !is.na(arr_time)) %>%
  mutate(
    dep_time = datetime_func(year, month, day, dep_time),
    arr_time = datetime_func(year, month, day, arr_time),
    sched_dep_time = datetime_func(year, month, day, sched_dep_time),
    sched_arr_time =datetime_func(year, month, day, sched_arr_time)) %>%
  select(origin, dest, ends_with("delay"), ends_with("time"))

# Plot
flights_dt %>%
  ggplot(aes(dep_time)) +
  geom_freqpoly(binwidth = 86400)+ # 86400 seconds = 1 day
  ggtitle ("Analysis of deep_delay over the course of the year [2013-2014]") +
  theme(legend.position="top") +
  theme_dark() +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



#

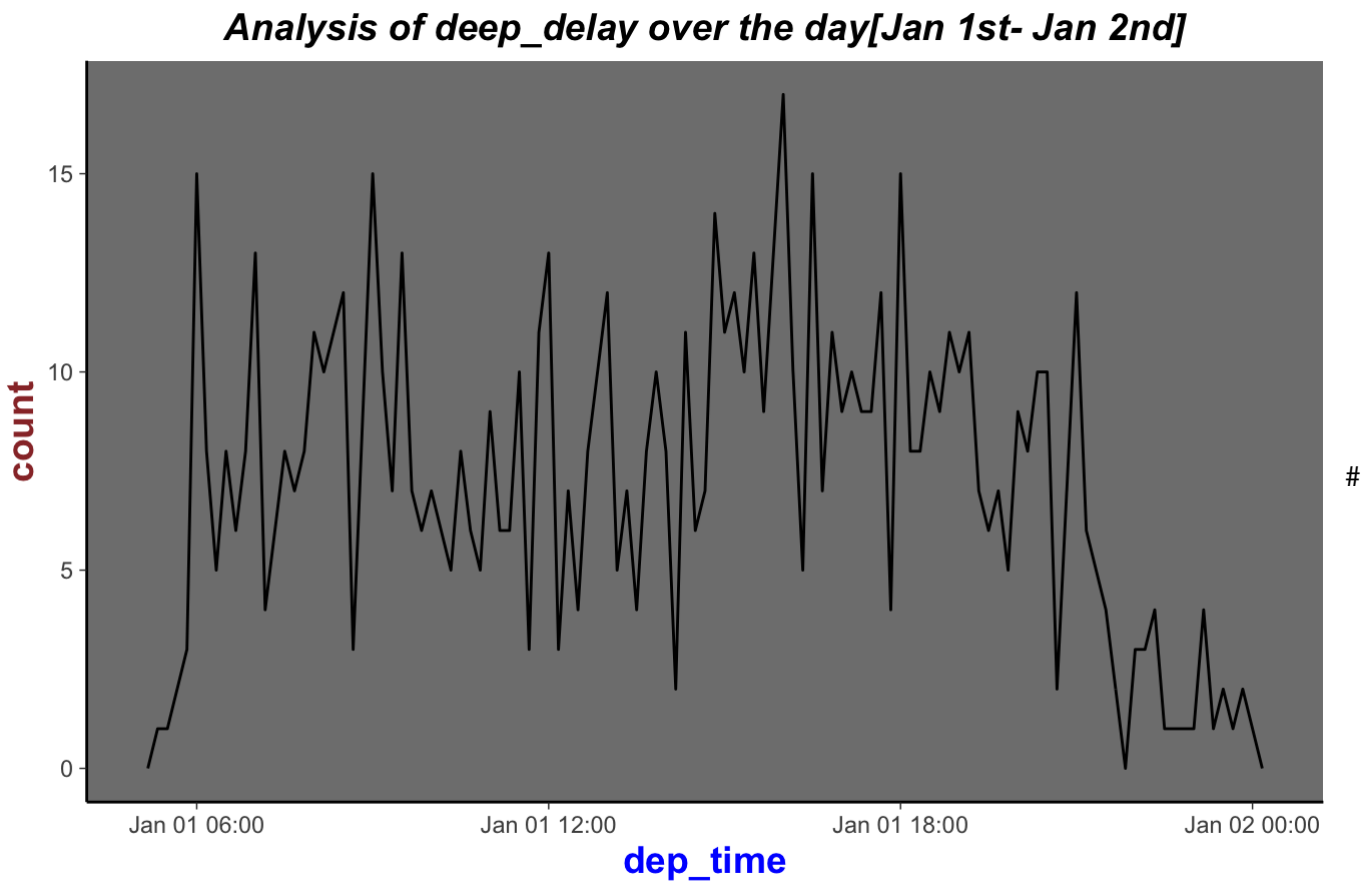
Q22. Overall Analysis of deep_delay over the course of the Week.

```
flights_dt %>%
  mutate(wday = wday(dep_time, label = TRUE)) %>%
  ggplot(aes(x = wday)) +
  geom_bar(stat="count") +
  ggtitle ("Analysis of deep_delay over the course of the Week") +
  theme(legend.position="top") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Q23. Overall Analysis of deep_delay over the course of the day.

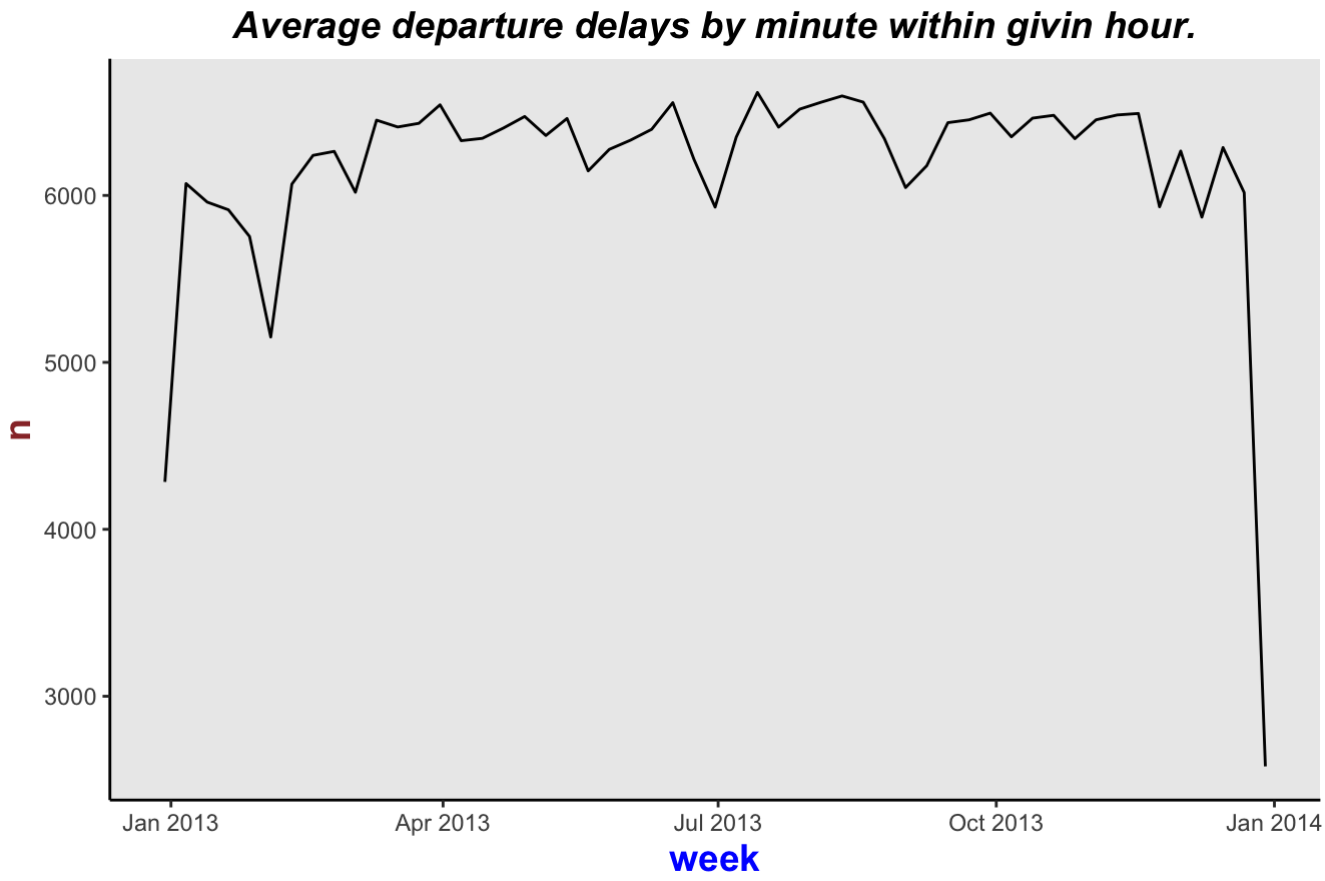
```
flights_dt %>%
  filter(dep_time < ymd(20130102)) %>%  # On 2nd Jan 2013
  ggplot(aes(dep_time)) +
  geom_freqpoly(binwidth = 600)+ # 600 s = 10 minutes
  ggtitle ("Analysis of deep_delay over the day[Jan 1st- Jan 2nd]") +
  theme(legend.position="top") +
  theme_dark() +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



#

Q24. Overall Analysis of deep_delay over the hour [2013-2014]

```
flights_dt %>%
  count(week = floor_date(dep_time, "week")) %>%
  ggplot(aes(week, n)) +
  geom_line()+
  ggtitle ("Average departure delays by minute within givin hour.") +
  theme(legend.position="top") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

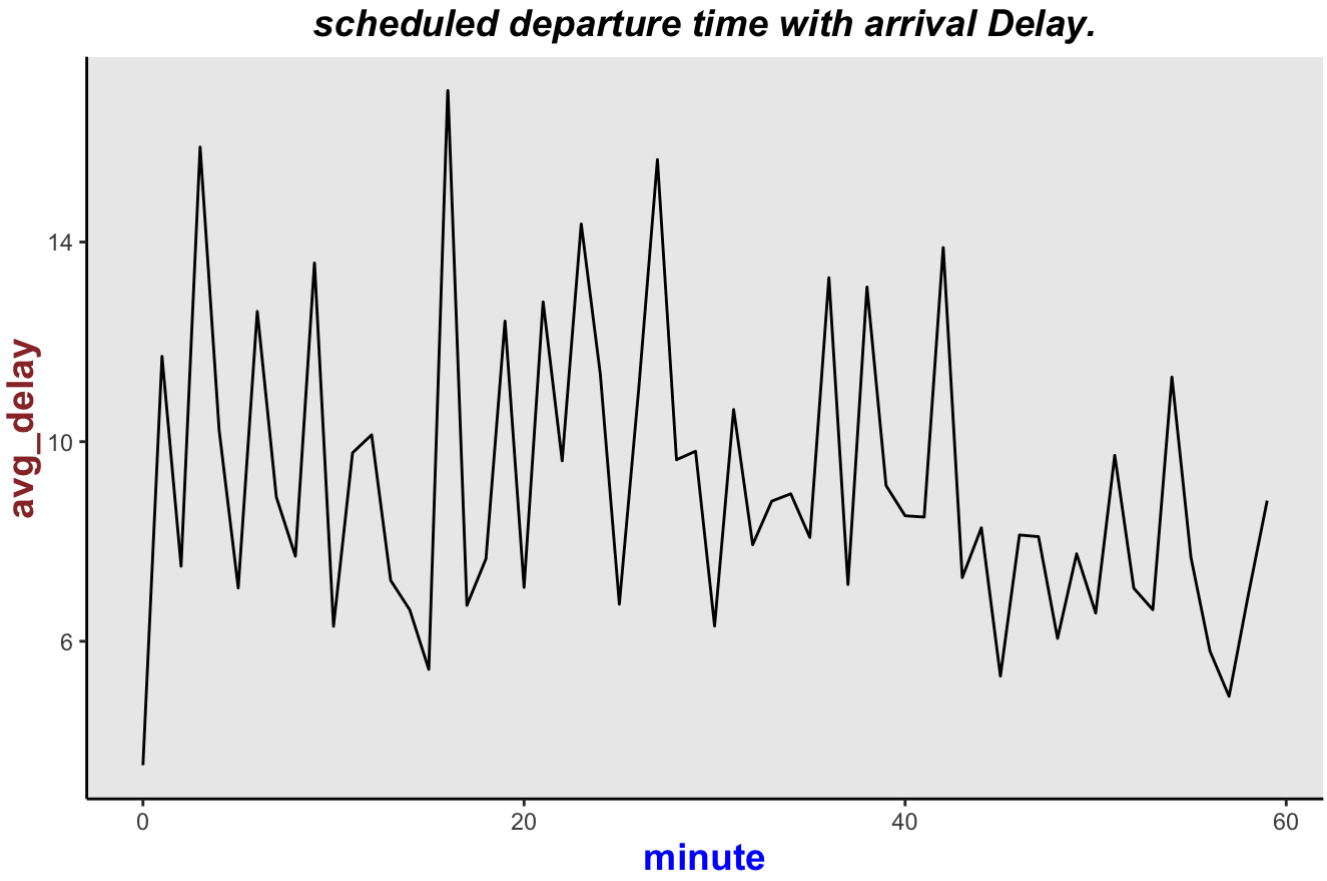


ANALYSIS

Average departure delay by minute within the hour. It looks like flights leaving in minutes 20-30 and 50-60 have much lower delays than the rest of the hour.

Q25. How is the flights scheduled for the given airports?

```
sched_dep <- flights_dt %>%
  mutate(minute = minute(sched_dep_time)) %>%
  group_by(minute) %>%
  summarise(
    avg_delay = mean(arr_delay, na.rm = TRUE),
    n = n())

ggplot(sched_dep, aes(minute, avg_delay)) +
  geom_line()+
  ggtitle ("scheduled departure time with arrival Delay.") +
  theme(legend.position="top") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

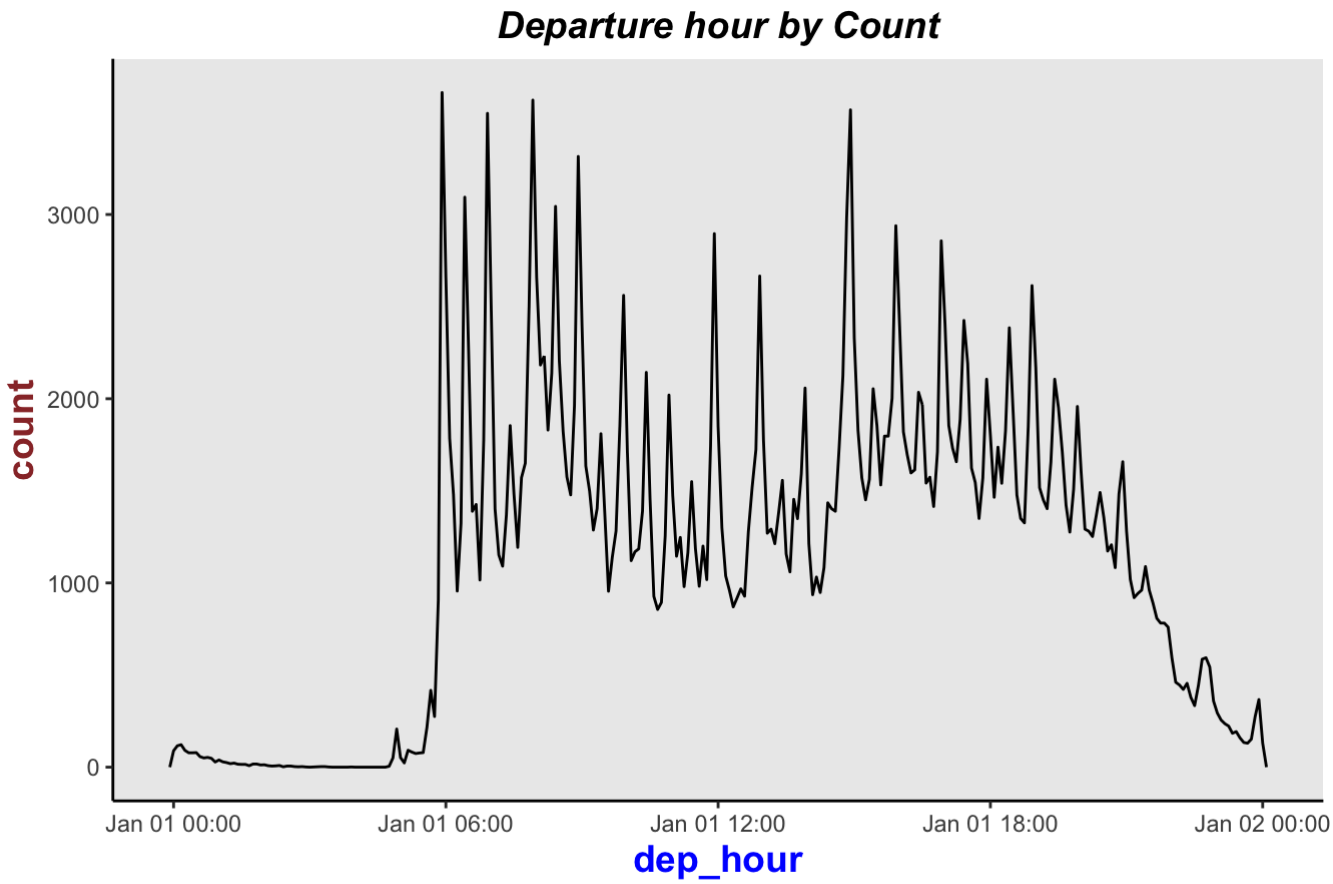


ANALYSIS

If we look at the scheduled departure time we don't see such a strong pattern.

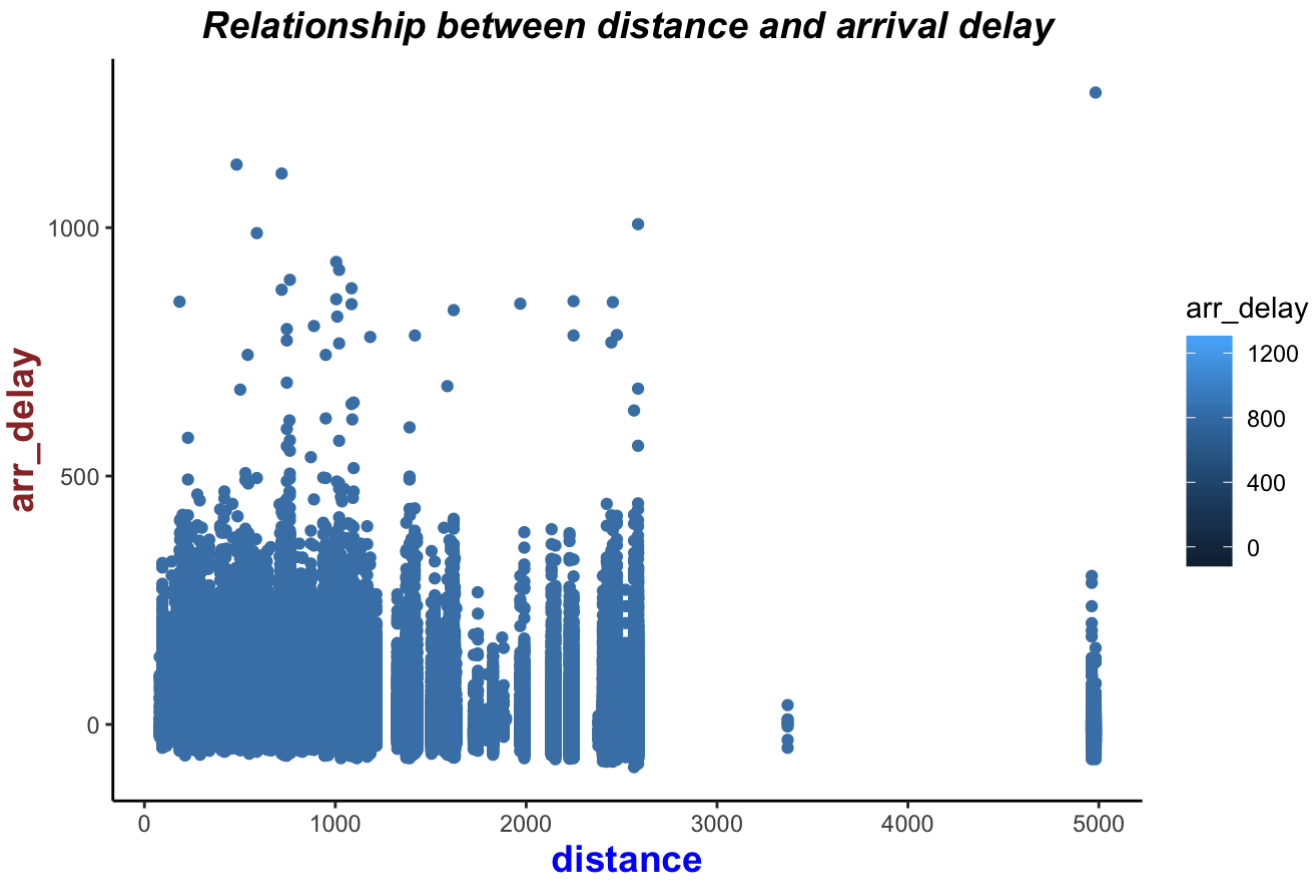
Q26. Departur hour by Count

```
flights_dt %>%
  mutate(dep_hour = update(dep_time, yday = 1)) %>%
  ggplot(aes(dep_hour)) +
  geom_freqpoly(binwidth = 300)+
  ggtitle ("Departure hour by Count") +
  theme(legend.position="top") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15), color="#D70026",f
ace="bold.italic" ),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Q27. Relationship between distance and arrival delay

```
ggplot(flt_1, aes(distance, arr_delay, fill = arr_delay))+
  geom_point(color = "steelblue") +
  ggtitle ("Relationship between distance and arrival delay") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme_bw()+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

```
# breakdown time_hour to year, month, day, hour,min, sec. I will show you both ways of c
onverting date into Weekdays and the Week numbers.
date <- as.Date(flt_1$time_hour, format = "%Y-%m-%d %H:%M:%S" ) # This is old way of doi
ng
flt_1$weekday <- weekdays(date)
unique(flt_1$weekday)
```

```
## [1] "Tuesday"   "Wednesday" "Thursday"  "Friday"    "Saturday"  "Sunday"
## [7] "Monday"
```

```
flt_1$wdays<-ymd_hms(flt_1$time_hour) %>% wday() # This is new way of coding
unique(flt_1$wdays) # 1 : Sunday by default and 7 : Saturday
```

```
## [1] 3 4 5 6 7 1 2
```

```
flt_1<- flt_1 %>% mutate(week_d = ifelse(wdays %in% c(1,7), "weekend","weekdays")) # con
verting to wdays & weekend
table(flt_1$week_d) # sanity check
```

```
##
## weekdays  weekend
##   244046    83300
```

Q28. Was there any difference between Arrival delays on weekends & during the week?

```
print(" For Arrival Delays")
```

```
## [1] " For Arrival Delays"
```

```
# Aggregate to see how many arr_delay happen during weekdays and weekends.
aggregate(flt_1$arr_delay ~ flt_1$week_d,
          FUN = mean,
          na.rm = T)
```

flt_1$week_d <chr>	flt_1$arr_delay <dbl>
weekdays	8.574576
weekend	1.975786

2 rows

```
print(" T-test For Arrival Delays")
```

```
## [1] " T-test For Arrival Delays"
```

```
t.test(arr_delay~ week_d, data = flt_1)
```

```
##
##  Welch Two Sample t-test
##
## data:  arr_delay by week_d
## t = 39.135, df = 160780, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  6.268304 6.929275
## sample estimates:
## mean in group weekdays  mean in group weekend
##               8.574576               1.975786
```

```
print(" For Departure Delays ")
```

```
## [1] " For Departure Delays "
```

```
# Aggregate to see how many dep_delay happen during weekdays and weekends.
aggregate(flt_1$dep_delay ~ flt_1$week_d,
          FUN = mean,
          na.rm = T)
```

flt_1$week_d <chr>	flt_1$dep_delay <dbl>
weekdays	13.52435
weekend	9.71569

2 rows

```
print(" T-test For Departure Delays")
```

```
## [1] " T-test For Departure Delays"
```

```
t.test(dep_delay~ week_d, data = flt_1)
```

```
##
##  Welch Two Sample t-test
##
## data:  dep_delay by week_d
## t = 25.381, df = 164010, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  3.514546 4.102770
## sample estimates:
## mean in group weekdays  mean in group weekend
##               13.52435                9.71569
```

ANALYSIS

Comparison done by T-test.

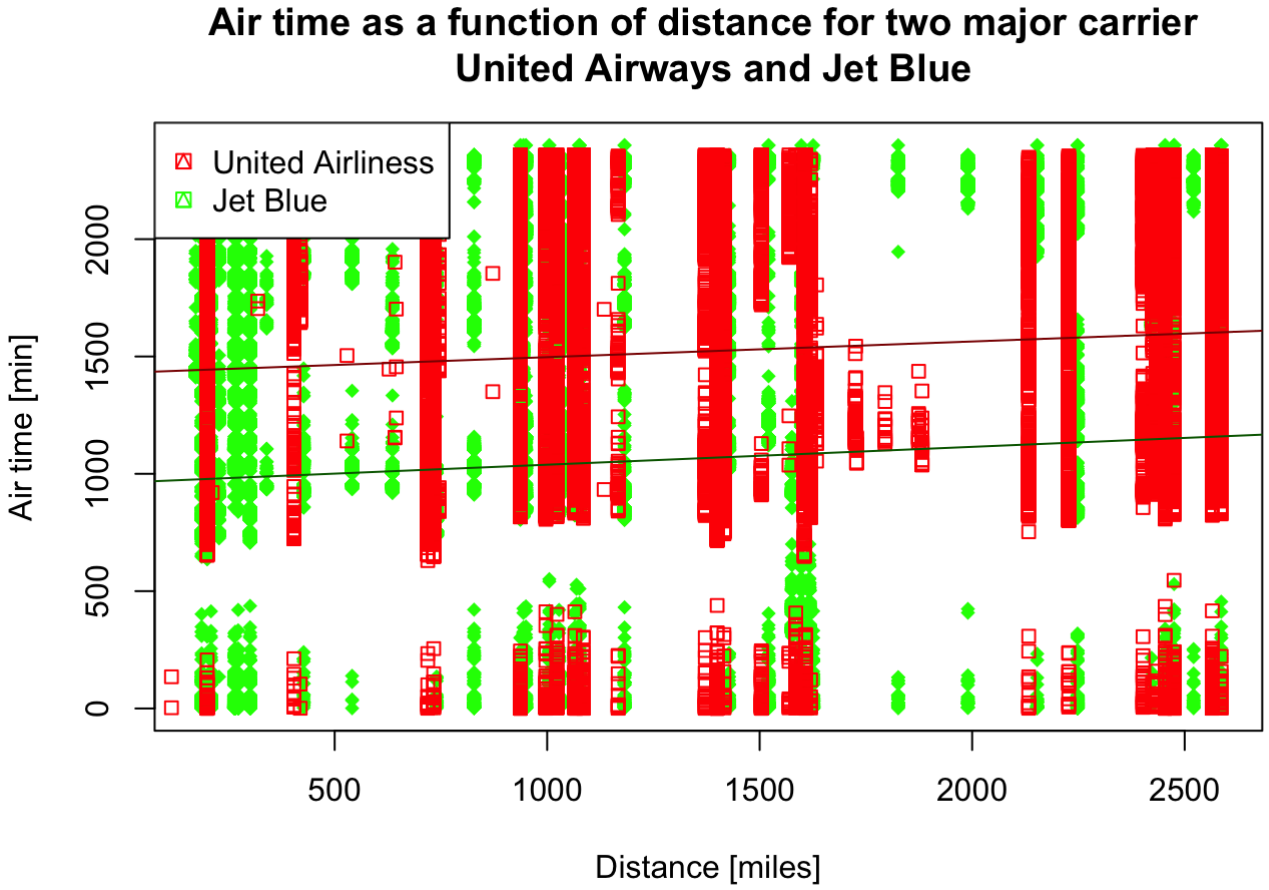
Q29. Linear Regression Analysis between two major airliness.

Linear Regression Model for Air time as a function of distance for two major carrier United Airways and Jet Blue.

```
flights_per_carrier <- cbind (Frequency  = table(unique(flt_1$carrier)),
                              RelFreq = prop.table (table(unique(flt_1$carrier))))

head(flights_per_carrier, 10) # Will show top 10 table
```

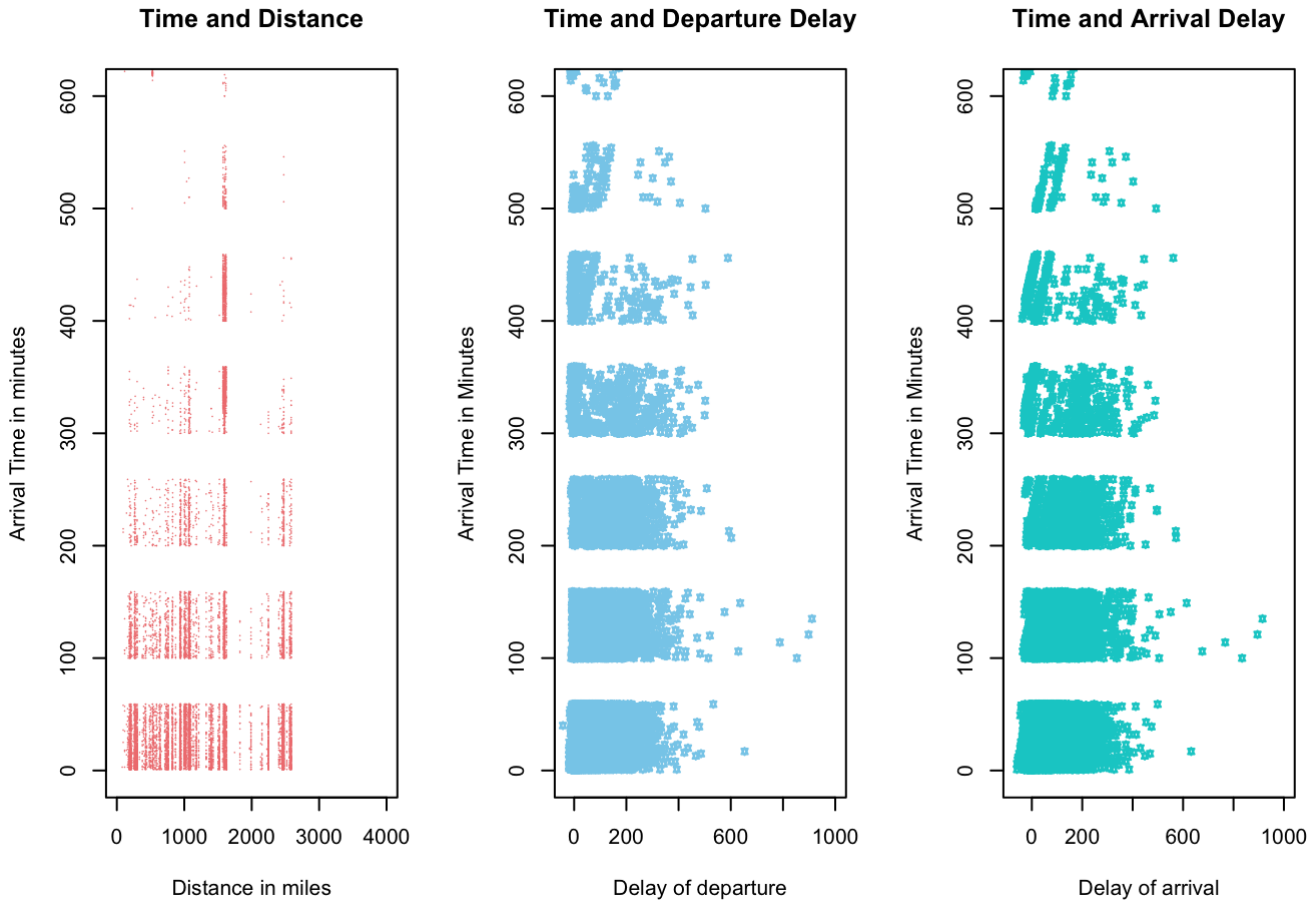
```
##    Frequency RelFreq
## 9E         1  0.0625
## AA         1  0.0625
## AS         1  0.0625
## B6         1  0.0625
## DL         1  0.0625
## EV         1  0.0625
## F9         1  0.0625
## FL         1  0.0625
## HA         1  0.0625
## MQ         1  0.0625
```

```
jet_blue <- subset(flt_1, carrier == 'B6')
united_air <- subset(flt_1, carrier == 'UA')

# Linear Regression
air_time.lm.jet_blue <- lm(distance ~ arr_time, data = jet_blue)
air_time.lm.united_air <- lm(distance ~ arr_time, data = united_air)

plot (x = jet_blue$distance,y = jet_blue$arr_time,xlab = 'Distance [miles]',ylab = 'Air
 time [min]',main = 'Air time as a function of distance for two major carrier \n United
 Airways and Jet Blue',pch=18,col='green')
points (x = united_air$distance, y = united_air$arr_time,pch=22,col='red')
abline (air_time.lm.jet_blue , col = 'darkgreen')
abline (air_time.lm.united_air, col = 'darkred')

legend ('topleft',legend = c('United Airliness', 'Jet Blue'),col = c('red', 'green'),pch
 = 14)
```



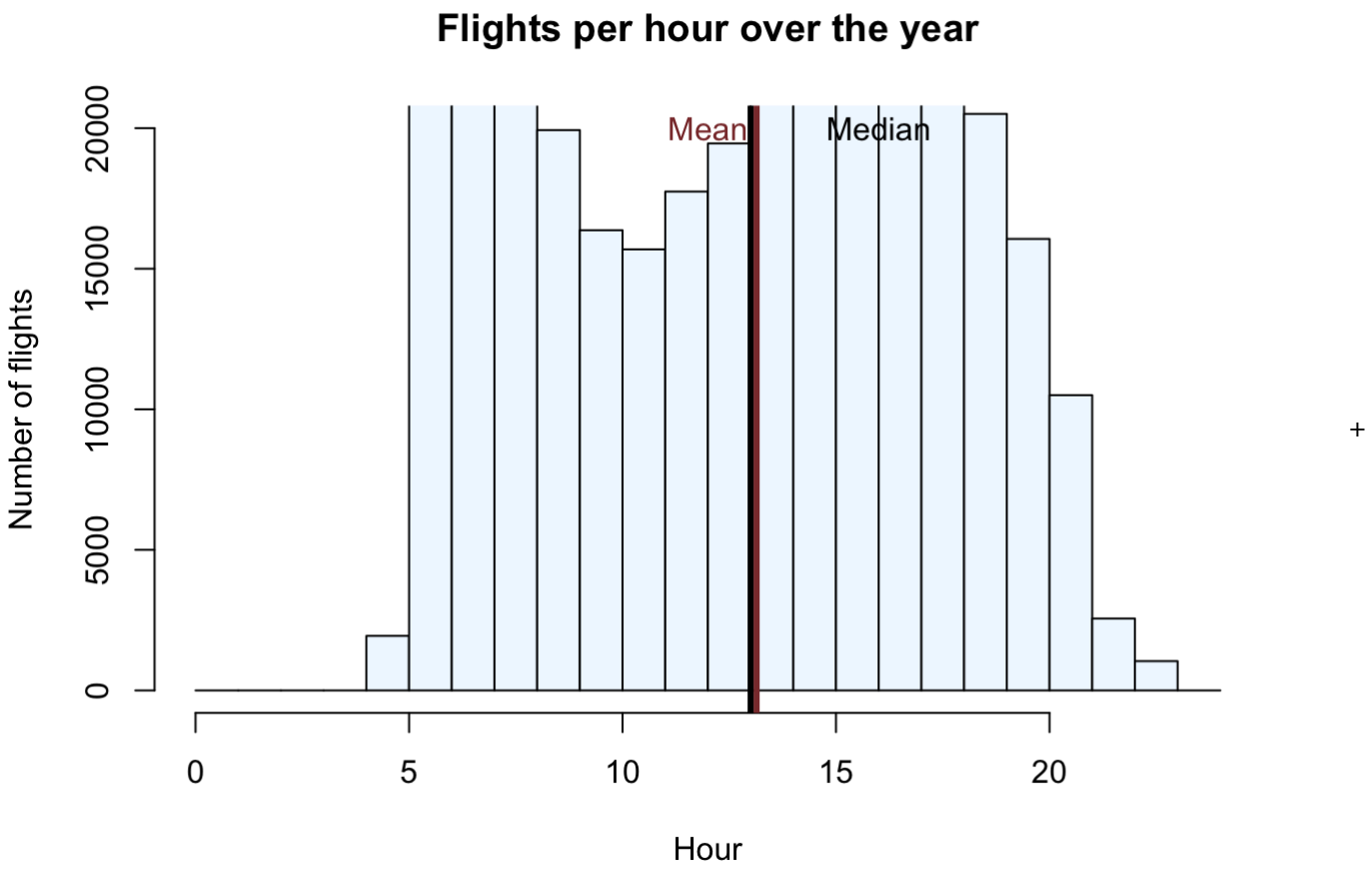
Q30. Relationship of Time with 3 other dependent variables.

```
#Put the plots next to each other
par(mfrow = c(1, 3))

#First Plot - Distance
plot(x = flt_1$distance,y = flt_1$arr_time,main = "Time and Distance",xlab = "Distance i
n miles", ylab = "Arrival Time in minutes",xlim = c(0, 4000),ylim = c(0, 600), col = "li
ghtcoral",pch = 18, type = "p",cex = 0.2)
abline(lm(flt_1$arr_time ~ flt_1$distance),col = "black",lty = 1)

# Second Plot - dep_delay
plot(x = flt_1$dep_delay,y = flt_1$arr_time,main = "Time and Departure Delay",xlab = "De
lay of departure",ylab = "Arrival Time in Minutes",xlim = c(-33, 1000),ylim = c(0, 600),
col = "skyblue", pch = 11,type = "p",cex = 0.5)
abline(lm(flt_1$arr_time ~ flt_1$dep_delay),col = "black",lty = 1)

# Third Plot - arr_delay
plot(x = flt_1$arr_delay,y = flt_1$arr_time,main = "Time and Arrival Delay",xlab = "Dela
y of arrival",ylab = "Arrival Time in Minutes",xlim = c(-70, 1000),ylim = c(0, 600), col
 = "cyan3",pch = 11, type = "p",cex = 0.5)
abline(lm(flt_1$arr_time ~ flt_1$arr_delay),col = "black",lty = 1)
```



Below We will create a histogram which will show how many flights were taken in each hour throughout the year.

```
hist(x = flt_1$hour,main = "Flights per hour over the year", xlab = "Hour",ylab = "Numbe
r of flights",col = "aliceblue",border = "black",xlim = c(0,24),ylim = c(0,20000),breaks
 = seq(0,24, by = 1))
abline(v = median(flt_1$hour, na.rm = T),col = "black",lwd = 3,lty = 1)
text(x = 16,y = 20000,labels = "Median", lwd = 2)
abline(v = mean(flt_1$hour, na.rm = T),col = "indianred4",lwd = 3, lty = 1)
text(x = 12,y = 20000, labels = "Mean",lwd = 2,col = "indianred4")
```



Choose top three Carriers based on number from three airports interms of flights.

a. United Airliness (UA)
b. JetBlue(B6)
c. Express Jet Airliness (EV)

Q31. Dive deep into three specific Airliness.

In order to look deeper into our findings, we chose 3 airline companies: - United Airlines (UA) - JetBlue(B6)) - Express Jet Airlines (EV)

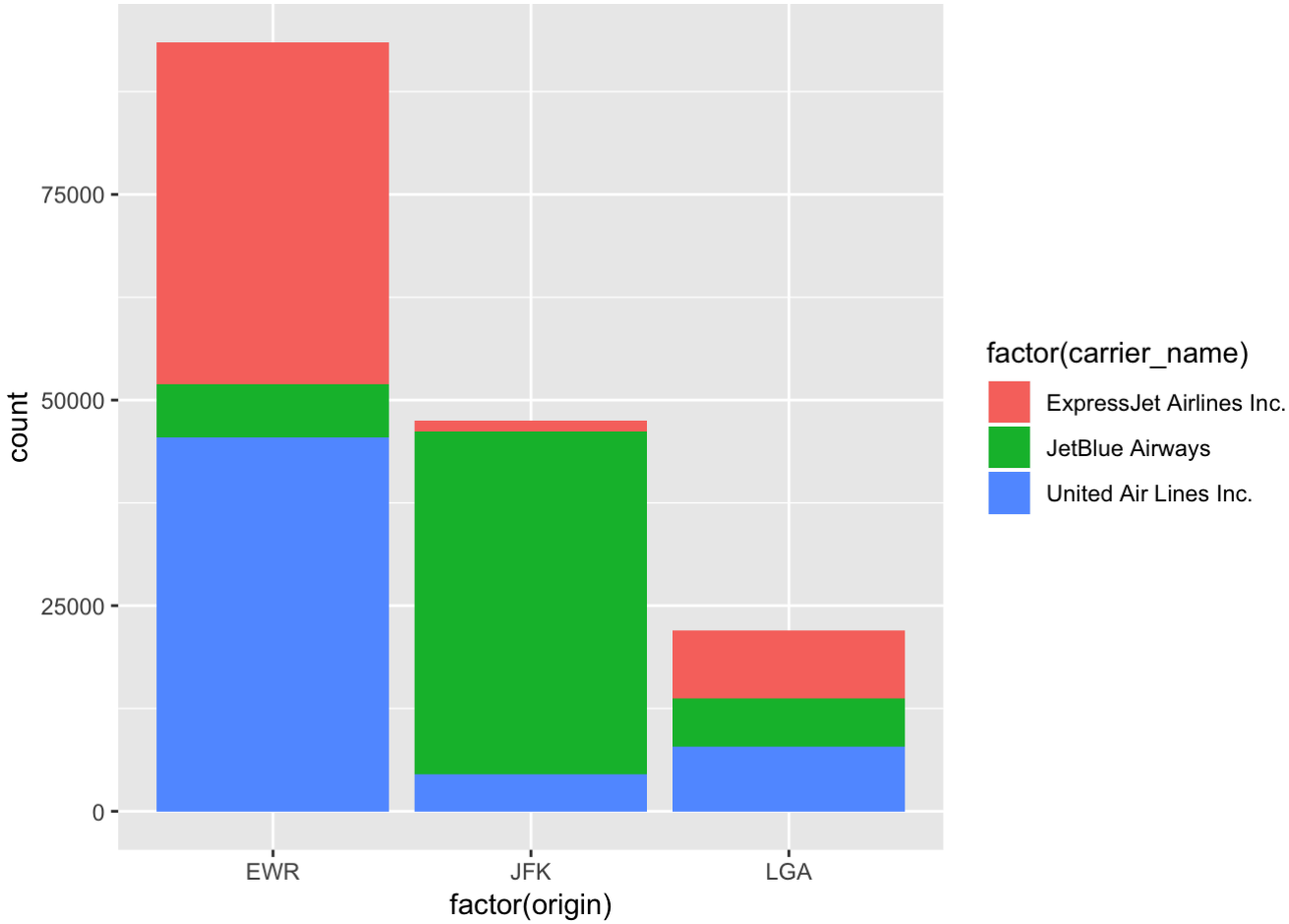
out of the 16 carriers that fly out of NYC. The 3 airlines selected are typically categorized into the same market positon: budget airlines, which allows us to meaningfully compare and contrast their performance. Let's make a subset of budget airlines where we will choose the top three carriers.

```
budget_flights <- flt_1 %>%
                  filter(carrier == 'UA'|carrier == 'B6'|carrier == 'EV')
# head(budget_flights) # Uncomment to see the top of the budget airliness.
```

Q32. Summaries within airports from chosen three flights.

A horizontal comparison among budget airlines within different airports displays the following results:

```
ggplot(data = budget_flights) +
  aes(x= factor(origin)) +
  geom_bar(aes(fill= factor(carrier_name)))
```



ANALYSIS

- We can see Jetblue has large share in JFK Airport.
- Delta has roughly similar number of flights in JFK and LGA but very less from Newark Liberty Airport.

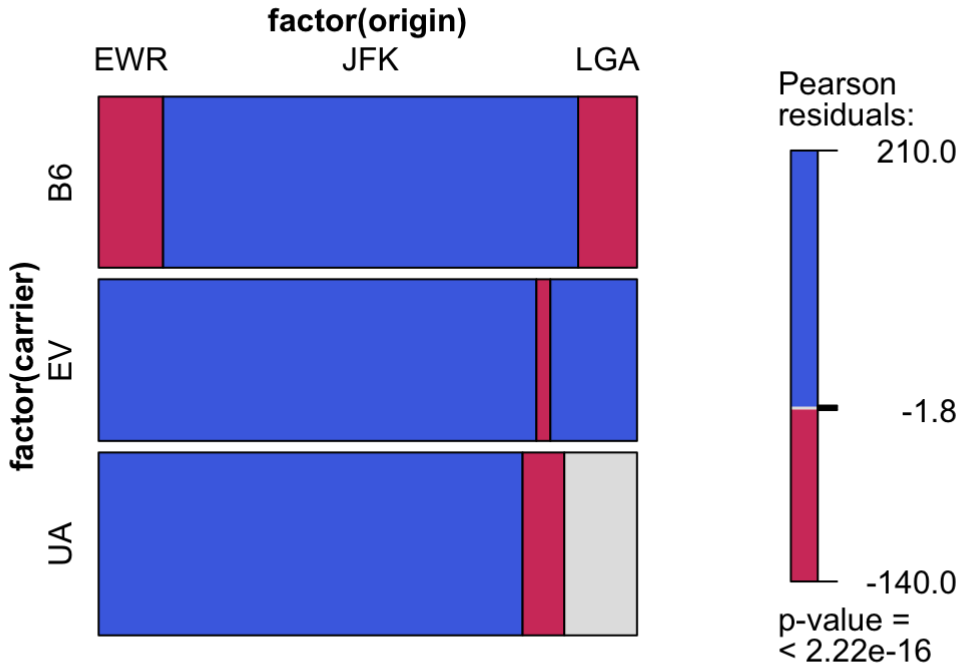
Q33. Overall Share in Mosiac plot

The share of flights among airports can be presented more easily in the output below:

```
xtabs(data=budget_flights,
      formula=~ factor(origin) +
        factor(carrier))
```

```
##               factor(carrier)
## factor(origin)    B6    EV    UA
##            EWR  6472 41557 45501
##            JFK 41666  1326  4478
##            LGA  5911  8225  7803
```

```
mosaic(formula= factor(origin)~ factor(carrier),
       data=budget_flights, gp = shading_hcl, gp_args = list(interpolate = c(1, 1.8)))
```

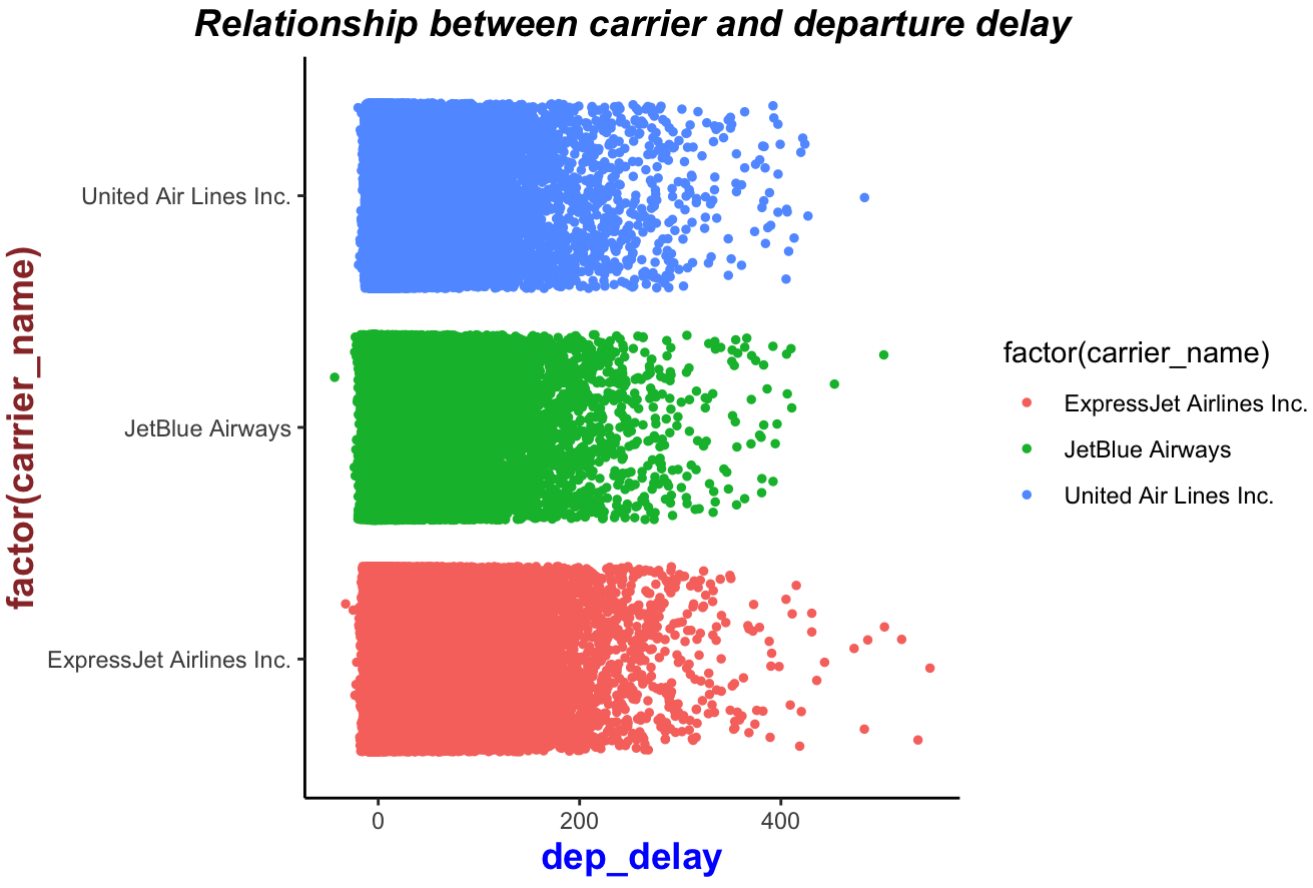


ANALYSIS

- JetBlue has a larger share of its flights in JFK.
- Delta has most of its flights in JFK and LGA. (Roughly equal distribution)
- Summaries among delays

The following strip plot shows a comparison output of the three airlines on their performance of depature delay time:

```
ggplot(data=budget_flights,
       aes(x=dep_delay,
           y= factor(carrier_name))) +
  geom_point( aes(color= factor(carrier_name)),
              size=1,
              position= position_jitter(height=0.4))+
  ggtitle ("Relationship between carrier and departure delay") +
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme_bw()+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



We have to spend a lot of Exploratory work on finding out there are delays in arrival, departure not only in one season but every season throughout the year. We tried to correlate it with various factors but without analyzing with weather datasets our reports will be incomplete so let's see the effects our delays with the weather datasets.

Q34. Relationship between Weather and Departure delays.

```
weather_n_flights <- flights %>%
                      mutate(dep_delay_by_hr = dep_delay/60) %>% # Convert it into Hour
 as all our datas are in minute
    select(origin, year, month, day, hour, dep_delay_by_hr) %>% # Select the specfic col
umns
    inner_join(weather, by = c("origin", "year", "month", "day", "hour"))
# glimpse(weather_n_flights) # Uncomment to see the weathe and flights datasets have bee
n merged well.
```

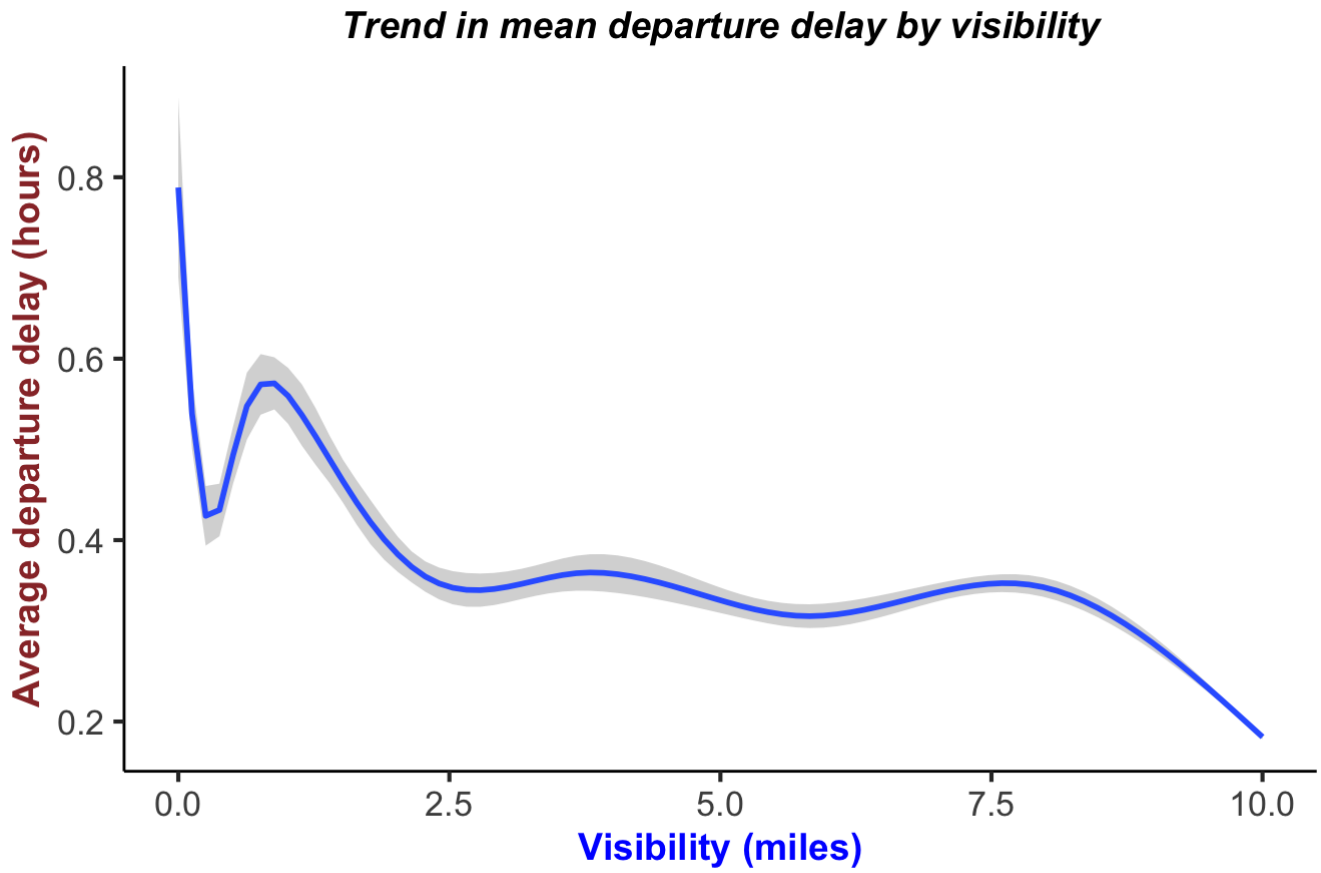
After merging two datasets, let's Look into the visibility factor which is major of airlines to operate smoothly.

- **Visibility:** Visibility is a measure of the distance at which an object or light can be clearly discerned. Visibility may vary according to the direction and angle of view, and the height of the observer. Visibility is affected by the presence of fog, cloud, haze, and precipitation.

Q35. Trend in mean departure delay by visibility

```
weather_n_flights %>% # Inner Join Weather with flights
    select(dep_delay_by_hr, visib) %>% # Select only column of our interest i.e delay an
d visibility
    filter(!is.na(dep_delay_by_hr) & !is.na(visib)) %>% # Filter out the missing datas
    ggplot(aes(x = visib, y = dep_delay_by_hr)) + # Plot the graph
    geom_smooth() +
    theme_bw(base_size = 16) +
    xlab("Visibility (miles)") + # X label
    ylab("Average departure delay (hours)") + # Y- label
    ggtitle("Trend in mean departure delay by visibility") + # Title
    theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Source: NYC-FLIGHTS datasets | @ Pankaj Shah

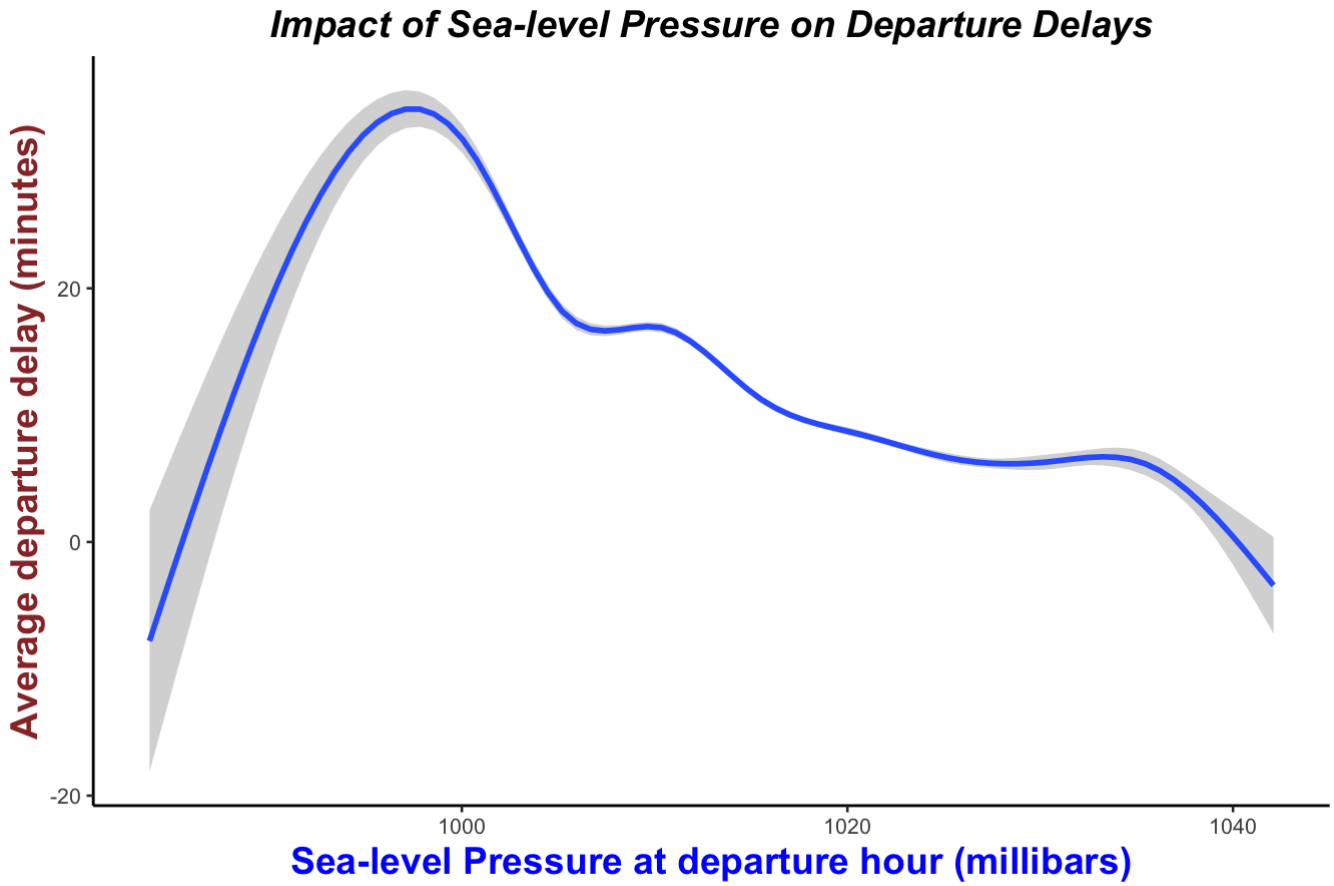
ANALYSIS

Aircraft departure and arrival is limited by the visibility (or RVR) to an extent that depends on the sophistication of ground equipment, the technical equipment fitted to the aircraft and the qualification of the flight crew. Many aerodromes and aircraft are fitted with equipment that makes possible a landing in very low visibility conditions provided the flight crew is suitably qualified; however, in very low visibility, it may prove impossible for the pilot to navigate the aircraft along the runway and taxiways to the aircraft stand.

Q36. Relationship between Pressure and Departure delays

```
flights %>%
    select(origin, dest, year, month, day, hour, dep_delay, arr_delay) %>%
    inner_join(weather, by = c("origin", "year", "month", "day", "hour")) %>%
    select(dep_delay, pressure) %>%
    filter(!is.na(dep_delay) & !is.na(pressure)) %>%
    ggplot(aes(x = pressure, y = dep_delay)) +
    geom_smooth() +
    theme_bw(base_size = 10) +
    xlab("Sea-level Pressure at departure hour (millibars)") +
    ylab("Average departure delay (minutes)") +
    ggtitle("Impact of Sea-level Pressure on Departure Delays")+
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah")+
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Let's pick one of the destinations in mid to long-haul flights to see the effects of Weather because on the short flight it will be hard to correlate the true findings. So just for a sake of the impact we can choose San Francisco as a final destination and see the variability in the speed.

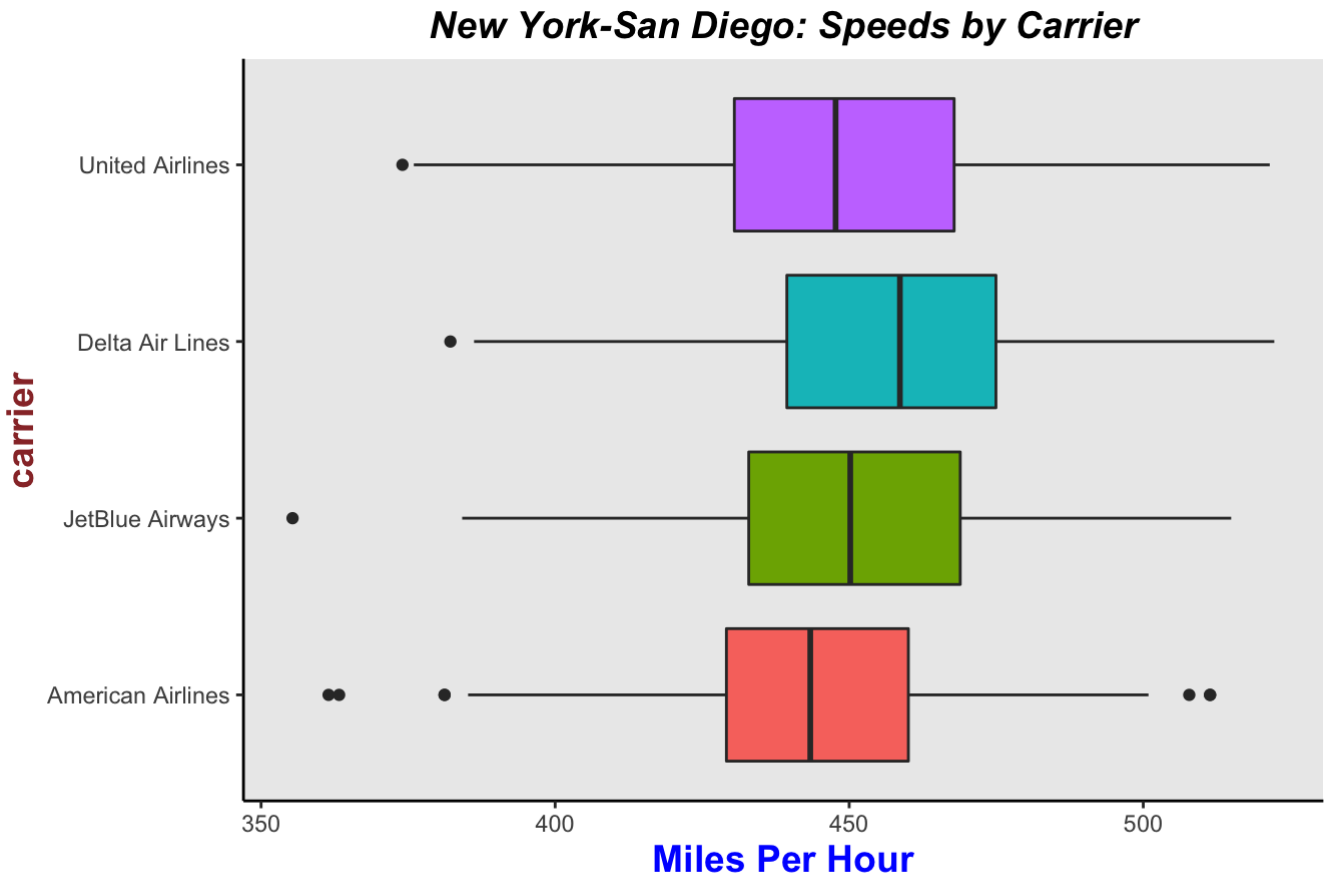
Source: NYC-FLIGHTS datasets | @ Pankaj Shah

Q37. Speeds by carrier

Take an example of flights from east coast NYC to San-diego.

```
flights %>%
    filter(dest == 'SAN') %>%
    left_join(planes, by = "tailnum") %>%
    left_join(airlines, by = "carrier") %>%
    mutate(milesperminute = distance / air_time,
           milesperhour = milesperminute * 60) %>%
    ggplot(aes(x = carrier, y = milesperhour, fill = carrier)) +
    geom_boxplot() +
    guides(fill=FALSE) +
    ggtitle("New York-San Diego: Speeds by Carrier") +
    ylab("Miles Per Hour") +
    coord_flip()+
    scale_x_discrete(breaks=c("AA", "B6", "DL", "UA"),
                     labels=c("American Airlines", "JetBlue Airways", # two are our budg
et airlines & two i choose as a control so that we are not bias in our datasets out of 1
6.
                              "Delta Air Lines", "United Airlines"))+
  theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

```
## Warning: Removed 28 rows containing non-finite values (stat_boxplot).
```



#

Source: NYC-FLIGHTS datasets | @ Pankaj Shah

Q38. Delays in flights to Midwest St. Louis by time of day

Just not to be bias I will choose one more flight somewhere in midwest to see the effects.

```
# Also we choose different airliness as we have choosen for the west coast airliness.
flights %>%
    filter (dest == "STL") %>%
    left_join(airlines) %>%
    mutate(dep_delay_hours = (dep_delay/60)) %>%
    filter(dep_delay_hours < 10 & carrier %in% c("AA", "EV", "WN", "MQ")) %>%
    ggplot(aes(x = hour, y = dep_delay_hours, color = name))+
    facet_wrap( ~ name) +
    geom_point () +
    xlab ("Hour of the Day") +
    ylab ("Departure Delay (Hours)") +
    geom_line(stat = "smooth", method = "loess" , aes(group = name, color = name)) +
    theme_bw() +
    scale_color_manual(name = "Airline",
                       values = c("American Airlines Inc." = "royalblue4",
                                  "DeltaAir Lines Inc." = "blue4",
                                  "Envoy Air" = "springgreen4",
                                  "ExpressJet Airlines Inc." = "yellow",
                                  "Southwest Airlines Co." = "orange",
                                  "United Air Lines Inc." = "skyblue")) +
    guides(color=FALSE) +
    ggtitle("Delays in flights to Midwest St. Louis by time of day")+
    theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

```
## Joining, by = "carrier"
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : pseudoinverse used at 7.95
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : neighborhood radius 7.05
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : reciprocal condition number 3.1227e-16
```

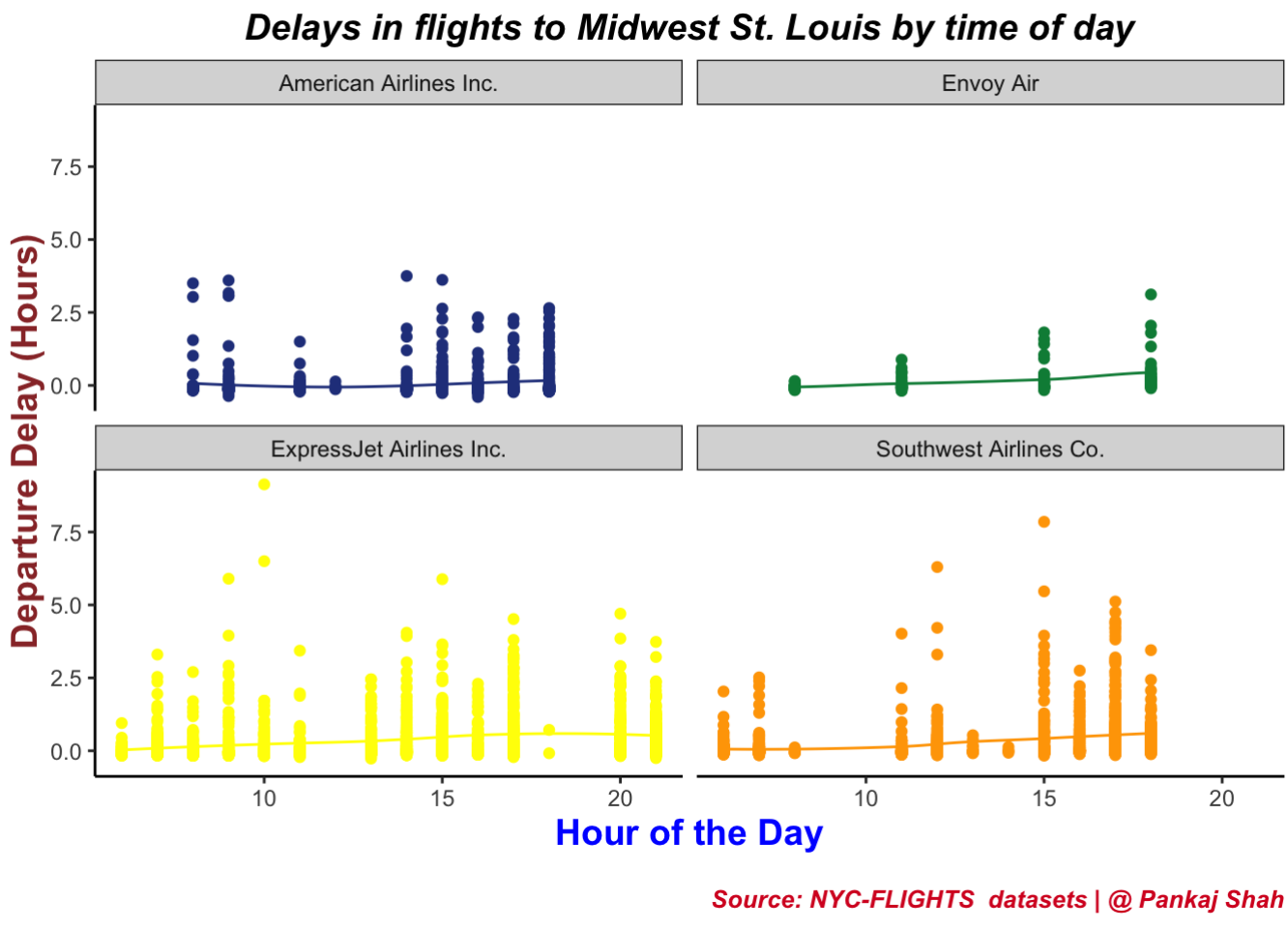
```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : There are other near singularities as well. 49.703
```

```
## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object)), : pseudoinverse used
## at 7.95
```

```
## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object)), : neighborhood radius
## 7.05
```

```
## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object)), : reciprocal
## condition number 3.1227e-16
```

```
## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object)), : There are other
## near singularities as well. 49.703
```



Drilling down to NYC airport for United Airliness.

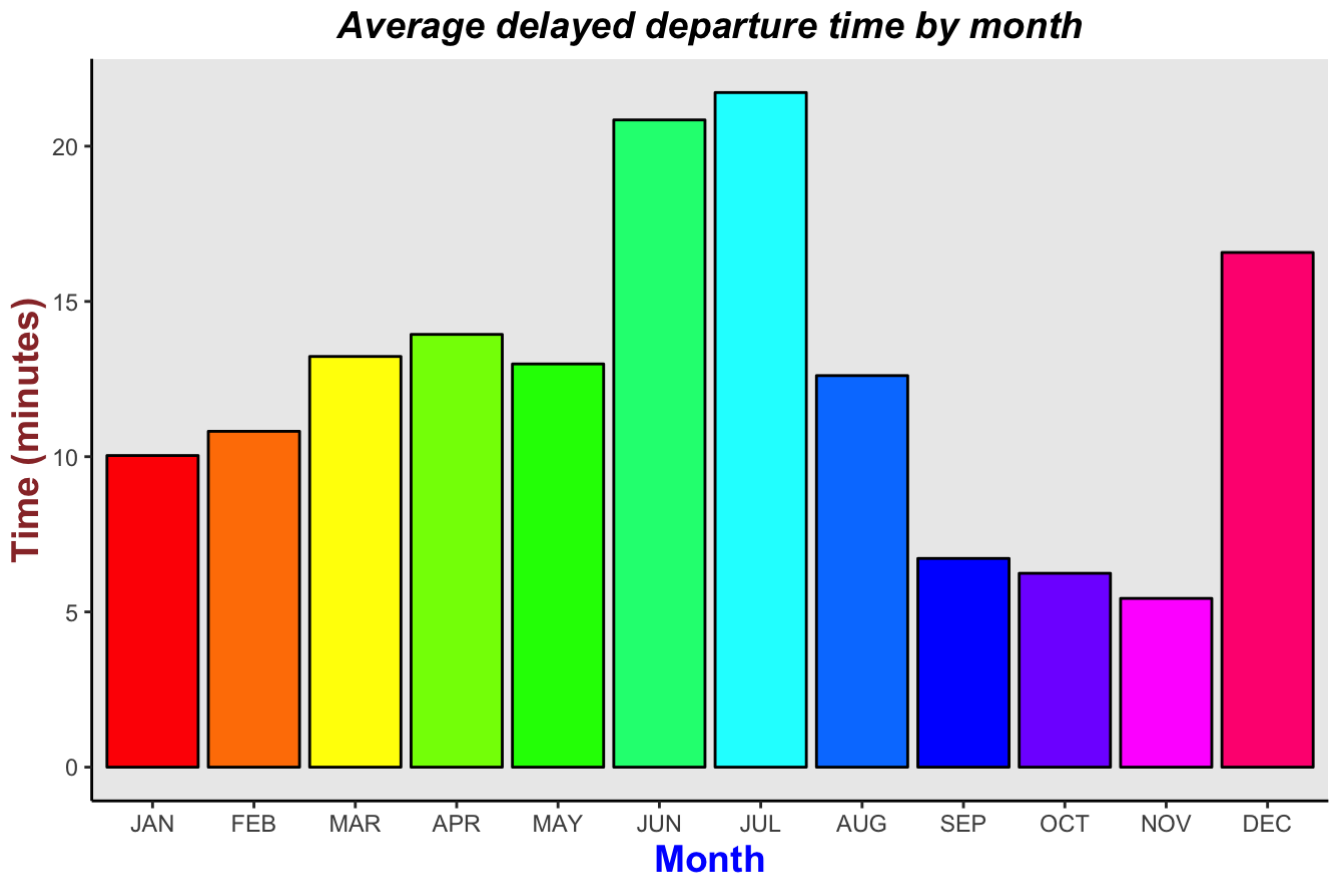
Q39. United's delays by NYC airport.

```
flights %>%
    rename(destination = dest,
           departure_delay = dep_delay,
           arrival_delay = arr_delay,
           time_in_air = air_time) %>%
    select(-year, -dep_time, -arr_time, -tailnum,
           -flight, -hour, -minute, -day) %>%
    mutate(ind_delayed_dep = ifelse(departure_delay > 0, 1, 0),
           ind_delayed_arr = ifelse(arrival_delay > 0, 1, 0)) %>%
    left_join(airlines, by = "carrier") %>%
    group_by(origin, name) %>%
    filter(carrier == "UA") %>%
    summarise(n_obs = n(),
              per_delayed_dep = round(sum(ind_delayed_dep, na.rm=TRUE) / n(),2),
              per_delayed_arr = round(sum(ind_delayed_arr, na.rm=TRUE) / n(),2)) %>%
    rename(Airport = origin,
           `Carrier Name` = name,
           `Number of Flights` = n_obs,
           `Proportion with Departure Delays` = per_delayed_dep,
           `Proportion with Arrival Delays` = per_delayed_arr) %>%
    pander(style = "rmarkdown", split.tables = 200)
```

Airport	Carrier Name	Number of Flights	Proportion with Departure Delays	Proportion with Arrival Delays
EWR	United Air Lines Inc.	46087	0.49	0.38
JFK	United Air Lines Inc.	4534	0.33	0.37
LGA	United Air Lines Inc.	8044	0.38	0.36

Q40. Departure delays by month

```
months_Str <- c("JAN","FEB","MAR","APR","MAY","JUN",
                "JUL","AUG","SEP","OCT","NOV","DEC")

flights %>%
    filter(!is.na(dep_delay) & !is.na(month)) %>%
    group_by(month) %>%
    summarise(mean_dep_delay = mean(dep_delay)) %>%
    # make month categorical by wrapping factor() around it and telling it the underlyin
g labels
    ggplot(aes(x = factor(month, labels = months_Str),
               y = mean_dep_delay)) +
    # rainbow colors on the bars, but not related to the values so not inside an aes().
 color = "black" does the outline
    geom_bar(stat = "identity", fill = rainbow(12), color = "black") +
    xlab("Month") +
    ylab("Time (minutes)") +
    ggtitle("Average delayed departure time by month")+
    theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

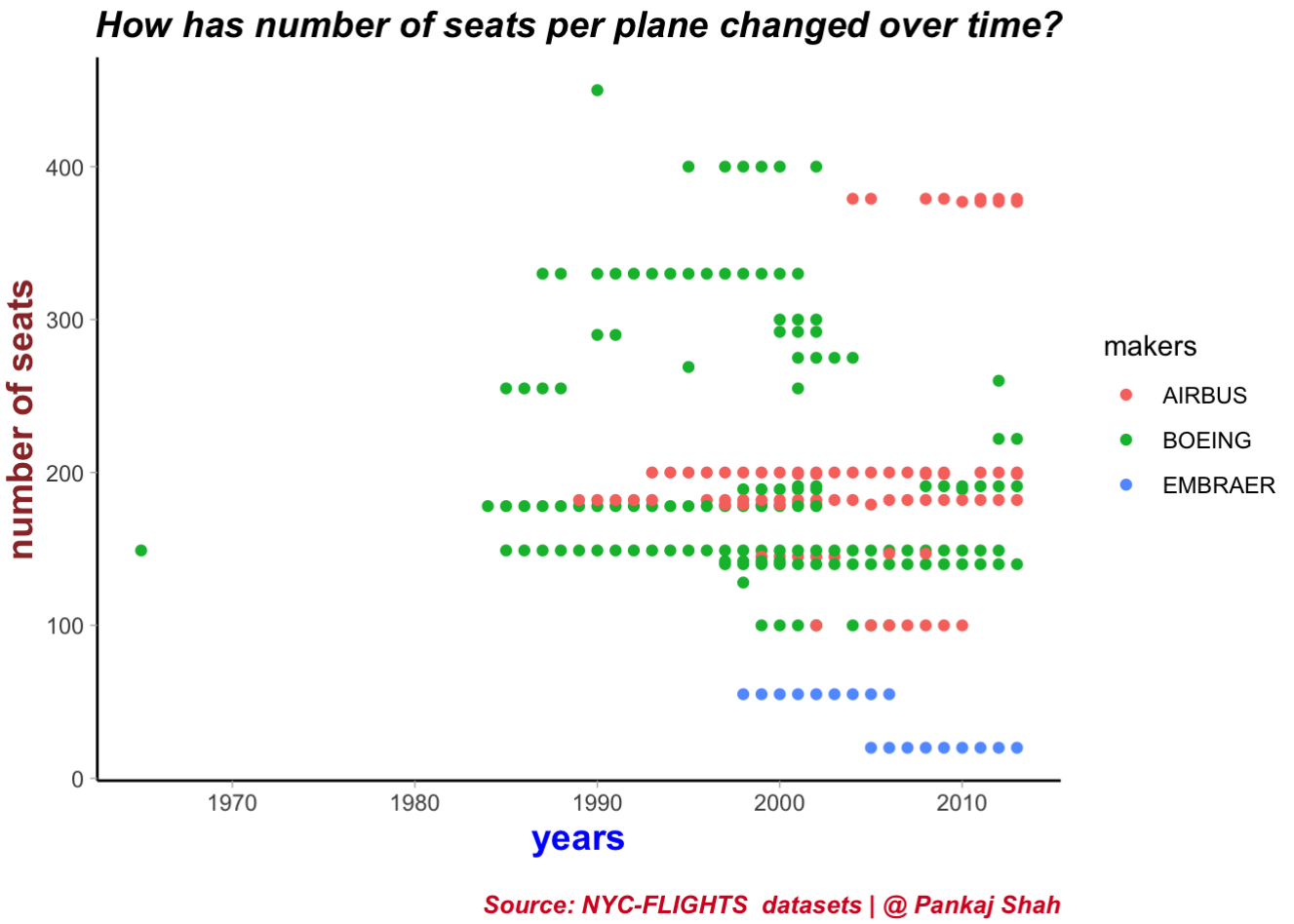


Source: NYC-FLIGHTS datasets | @ Pankaj Shah

Q41. How many seats are on a plane?

```
data_manufacturer <- flights %>%
    select(carrier, tailnum) %>%
    left_join(airlines, by = "carrier") %>%
    inner_join(planes, by = "tailnum") %>%
    # let's group Airbus and Airbus Industries into one category
    mutate(makers = ifelse((manufacturer == "AIRBUS" | manufacturer == "AIRBUS INDUSTRI
E"), "AIRBUS", as.character(manufacturer)))

# set up a vector of the top 3 within a list
list <- c("AIRBUS","BOEING","EMBRAER")

# how has the number of seats per plane changed over time?
ggplot(data = data_manufacturer %>%
           # filter to just the top 3 using the "in" function
           filter(makers %in% list) %>%
           select(makers, year, seats) %>%
           # just need one row per maker-year-seat
           # (many duplicates on the data!)
           distinct(makers, year, seats),
       aes(x = year, y=seats, color = makers)) +
  geom_point() +
  theme_light() + xlab("years") + ylab("number of seats") +
  ggtitle("How has number of seats per plane changed over time?")+
    theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

```
## Warning: Removed 16 rows containing missing values (geom_point).
```



Q42. Ovearll Stastical delays by month

```
by_month <- group_by(nycflights,month)
dep_delay_month <- summarise(by_month, Mean = round(mean(dep_delay),digits = 2),
                                       Median = round(median(dep_delay),digits = 2),
                                       IQR = IQR(dep_delay),
                                       MAX = max(dep_delay)
                                )
arrange(dep_delay_month,desc(Mean))
```

month <int>	Mean <dbl>	Median <dbl>	IQR <dbl>	MAX <dbl>
7	20.75	0	26	392
6	20.35	0	25	803
12	17.37	1	25	849
4	14.55	-2	16	427
3	13.52	-1	17	393
5	13.26	-1	19	351
8	12.62	-1	15	436

month <int>	Mean <dbl>	Median <dbl>	IQR <dbl>	MAX <dbl>
2	10.69	-2	15	319
1	10.23	-2	12	1301
9	6.87	-3	8	473

1-10 of 12 rows Previous **1** 2 Next

By knowing all the statistical numbers like the mean or the median a more reliable measure for deciding which month(s) to avoid flying if you really dislike delayed flights.

Q43. On time departure rate for NYC airports

```
nycflights_on_time <- nycflights %>%
  mutate(dep_type = ifelse(dep_delay < 5, "on time", "delayed"))

nycflights_on_time  %>%
  group_by(origin) %>%
  summarise(ot_dep_rate = sum(dep_type == "on time") / n()) %>%
  arrange(desc(ot_dep_rate))
```

origin <chr>	ot_dep_rate <dbl>
LGA	0.7279229
JFK	0.6935854
EWR	0.6369892

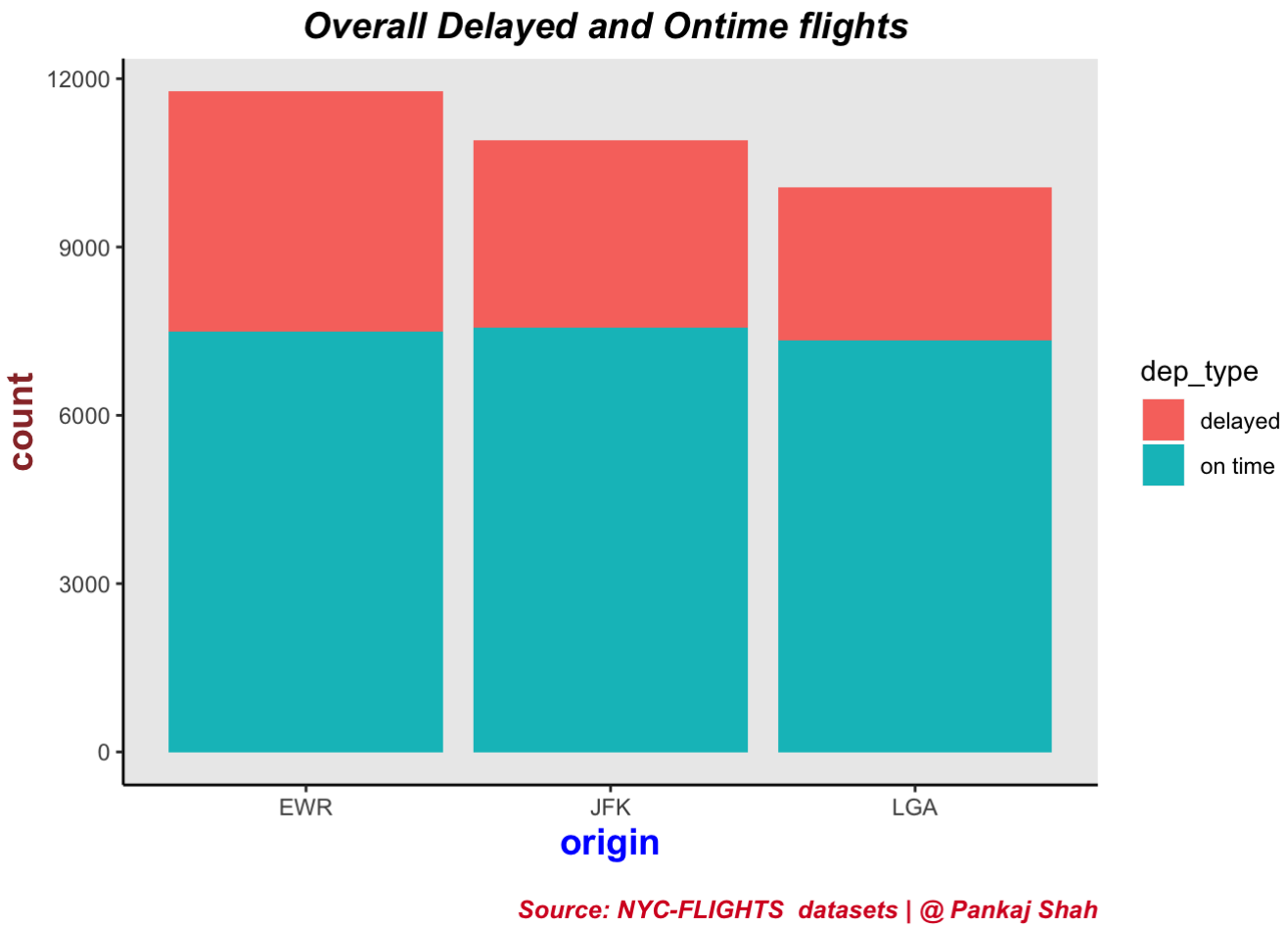
3 rows

If you were selecting an airport simply based on on time departure percentage, LGA airport would be best tof fly out.

```
# nycflights_on_time <- nycflights_on_time %>%
#         mutate(ontime = dep_delay < 5)
#
# nycflights_on_time %>%
#   group_by(origin) %>%
#   summarise(ontime_prop = sum(ontime == TRUE) / n()) %>%
#   arrange(desc(ontime_prop))
```

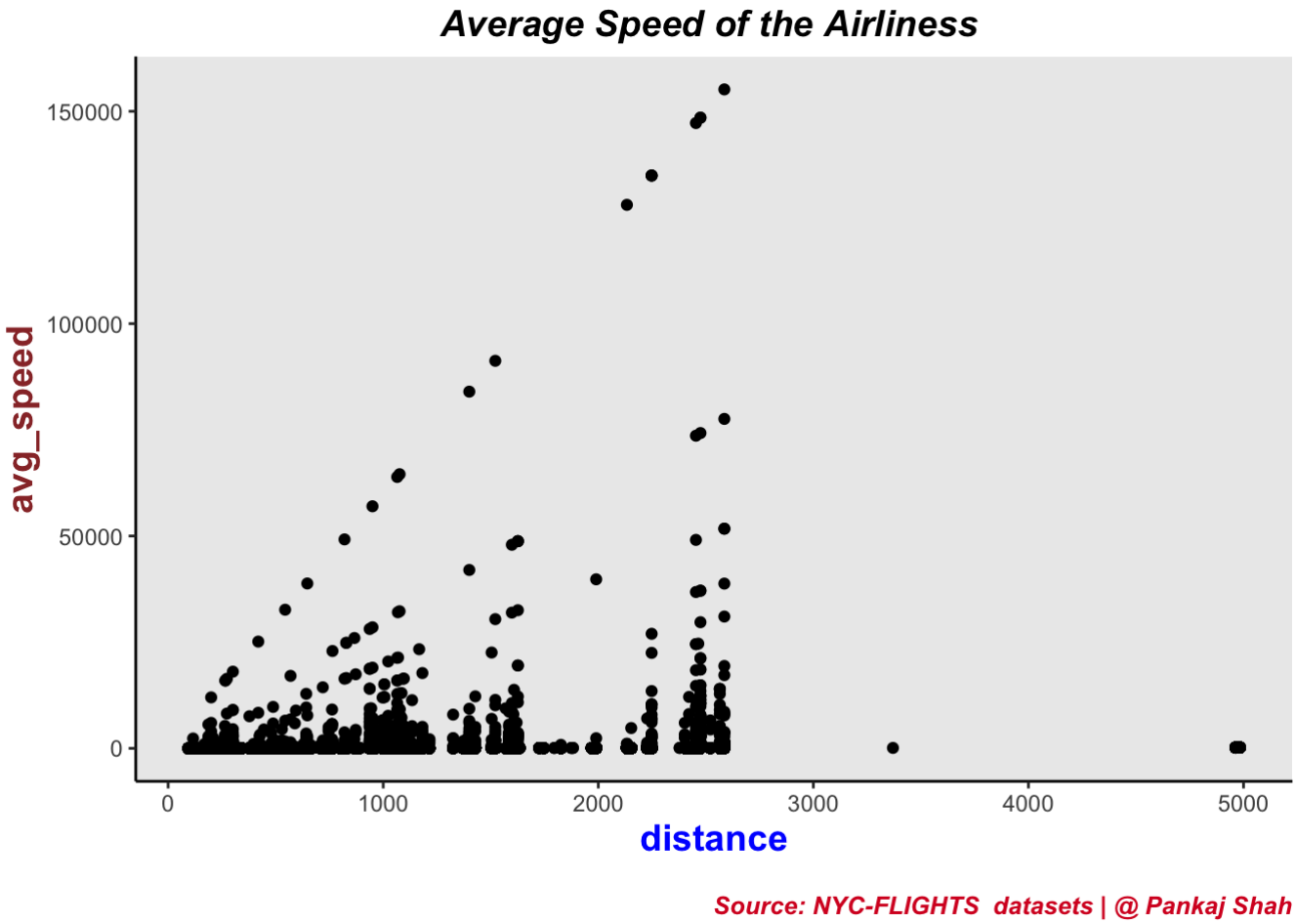
Q44. Overall Delayed and Ontime flights

```
ggplot(data = nycflights_on_time, aes(x = origin, fill = dep_type)) +
  geom_bar()+
  ggtitle("Overall Delayed and Ontime flights ") + # Title
    theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



#Q45. Average Speed of the Airplanes

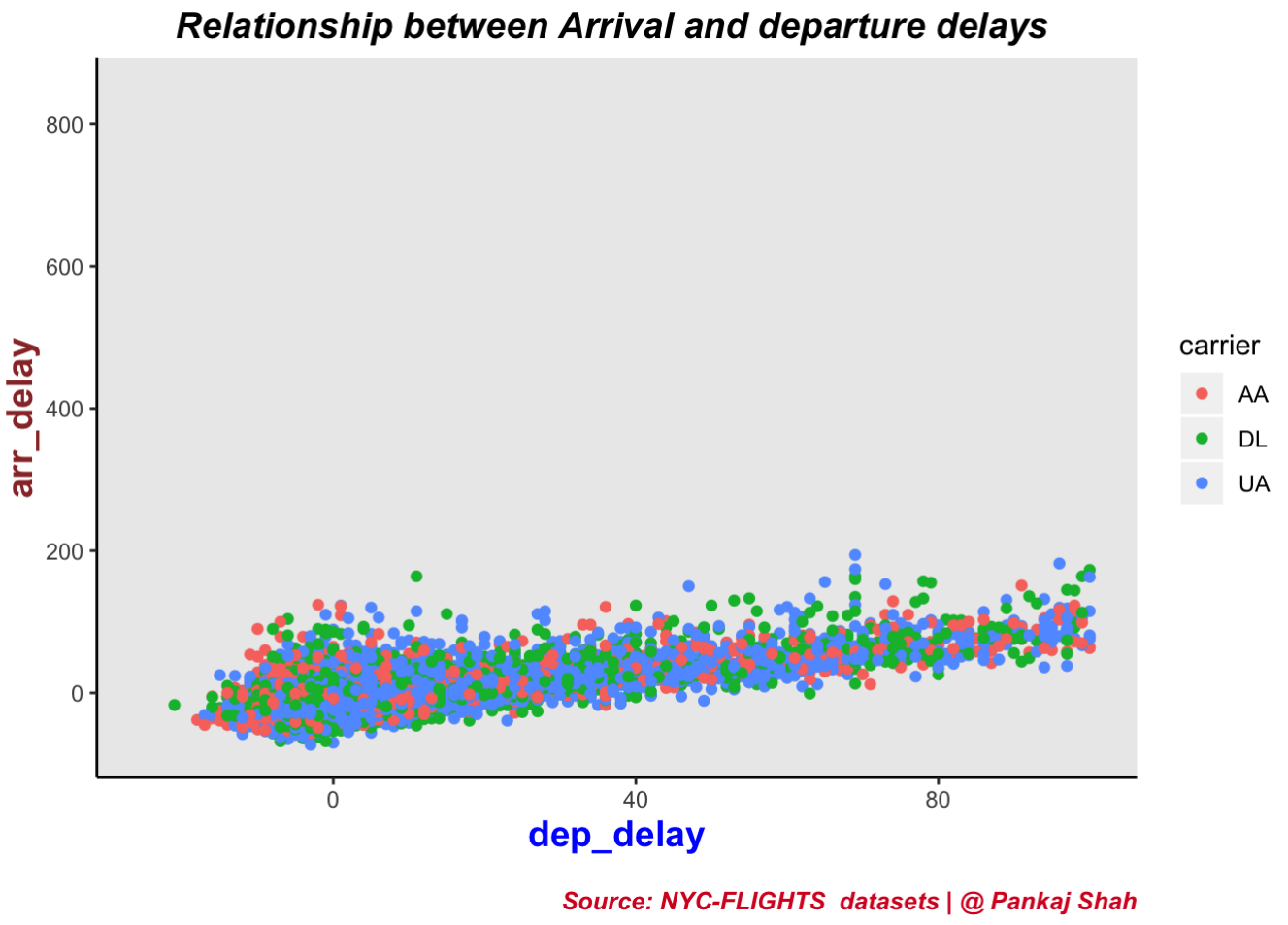
```
nycflights <- nycflights %>% mutate(avg_speed = distance / (arr_time/60))
ggplot(data = nycflights, aes(distance,avg_speed)) + geom_point()+
ggtitle("Average Speed of the Airliness ") + # Title
    theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```



The reason could be that with longer distances the start and landing time does not count so heavy as with short distances. There is one exception fast flight from LaGuardia to Atlanta. The very far flight distances (the points on the 5.000 miles distance rage) are FROM NYC to Honolulu (HNL), the shortest to Philadelphia (PHL).

```
dl_aa_ua <- nycflights %>%
  filter(carrier == "AA" | carrier == "DL" | carrier == "UA")

ggplot(dl_aa_ua, aes(x = dep_delay, y = arr_delay, color = carrier)) +
        xlim(-25, 100) +
        geom_point()+
  ggtitle("Relationship between Arrival and departure delays ") + # Title
    theme(plot.title=element_text(size=12))+ # Size and theme
  labs(caption = "Source: NYC-FLIGHTS  datasets | @ Pankaj Shah") +
  theme(
  axis.line.x = element_line(size = 0.5, colour = "black"),
  axis.line.y = element_line(size = 0.5, colour = "black"),
  axis.line = element_line(size=1, colour = "black"),
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  panel.background = element_rect(size = 0.5, linetype = "solid"),
  plot.caption=element_text(size=9.5, hjust=1.0, margin=margin(t= 15),color="#D70026",fa
ce="bold.italic"),
  plot.title = element_text(color="black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="blue", size=14, face="bold"),
  axis.title.y = element_text(color="#993333", size=14, face="bold"))
```

```
## Warning: Removed 444 rows containing missing values (geom_point).
```



Q46 How many planes were late than 5 Minutes.

```
nycflights <- nycflights %>%
        mutate(ontime = dep_delay < 5)
nycflights <- nycflights %>%
  mutate(arr_type = ifelse(dep_delay < 5, "on time", "delayed"))

nycflights %>% group_by(origin) %>% summarise(ontime_prop_1 = sum(arr_type == 'delayed')
 / n()) %>%
                                                arrange(desc(ontime_prop_1))
```

origin <chr>	ontime_prop_1 <dbl>
EWR	0.3630108
JFK	0.3064146
LGA	0.2720771

3 rows

Q47. Analysing on Single Plane

```
singleplane <- filter(flights, tailnum=="N355NB") %>%
  select(year, month, day, dest, origin, distance)
head(singleplane)
```

year <int>	month <int>	day <int>	dest <chr>	origin <chr>	distance <dbl>
2013	1	7	PIT	LGA	335
2013	1	8	FLL	LGA	1076
2013	1	9	PBI	LGA	1035
2013	1	10	MSP	LGA	1020
2013	1	21	PIT	LGA	335
2013	1	22	FLL	LGA	1076

6 rows

```
sum(singleplane$distance)
```

```
## [1] 106914
```

```
dim(singleplane)
```

```
## [1] 128   6
```

Q48. Looking other components of Weather on flight Datasets.

```
head(weather)
```

origin <chr>	year <dbl>	month <dbl>	day <int>	hour <int>	temp <dbl>	dewp <dbl>	humid <dbl>	wind_dir <dbl>	wind_speed <dbl>
EWR	2013	1	1	1	39.02	26.06	59.37	270	10.35702
EWR	2013	1	1	2	39.02	26.96	61.63	250	8.05546
EWR	2013	1	1	3	39.02	28.04	64.43	240	11.50780
EWR	2013	1	1	4	39.92	28.04	62.21	250	12.65858
EWR	2013	1	1	5	39.02	28.04	64.43	260	12.65858
EWR	2013	1	1	6	37.94	28.04	67.21	240	11.50780

6 rows | 1-10 of 15 columns

```
avgdelay <- flights %>%
  group_by(month, day) %>%
  filter(month < 13) %>%
  summarise(avgdelay = mean(arr_delay, na.rm=TRUE))
precip <- weather %>%
  group_by(month, day) %>%
  filter(month < 13) %>%
  summarise(totprecip = sum(precip), maxwind = max(wind_speed))
precip <- mutate(precip, anyprecip = ifelse(totprecip==0, "No", "Yes"))
merged <- left_join(avgdelay, precip, by=c("day", "month"))
head(merged)
```

month <dbl>	day <int>	avgdelay <dbl>	totprecip <dbl>	maxwind <dbl>	anyprecip <chr>
1	1	12.651023	0	24.16638	No
1	2	12.692888	0	20.71404	No
1	3	5.733333	0	17.26170	No
1	4	-1.932819	0	24.16638	No
1	5	-1.525802	0	20.71404	No

month	day	avgdelay	totprecip	maxwind	anyprecip
<dbl>	<int>	<dbl>	<dbl>	<dbl>	<chr>
1	6	4.236429	0	16.11092	No

6 rows

Take Home Message:

One thing to consider living in east coast during winter month is that Snowstorms are the most frequent instigators of massive flight delays and cancelations at the metro's big three airports, but aren't the only weather nuisance. Situated on Flushing Bay, storm surge from flood LaGuardia's runways, which can push as far as the terminal buildings and jetways. Fortunately, many planes can be moved ahead of time if the weather is known beforehand to other airports as a precaution. Flights will not resume until couple days later which might affect many flights. A line of thunderstorms ahead of an advancing cold front into the Northeast can also trigger significant flight delays on the order of several hours. Fortunately, thunderstorm days (24 to 26 each year, on average), aren't nearly as numerous as, say, Houston, Denver or Atlanta so we didn't see those effect on our datasets.

Of course, it's not just rain or snow that can delay your flight to the Big Apple. Low clouds and fog can trigger big delays. We can analyze further in details if time and situation permit to find more about wind delays at these airports. Two of Newark's three runways are oriented southwest to northeast. Since west to northwest crosswinds are common, this can be a frequent problem, even on sunny days. Winds can create headaches for pilots attempting to land, which is why we see planes circling EWR on lots of news article for delays around the month of June and July. Security accounted for a relatively low percentage —0.1%—of delay minutes for U.S. airlines according to most of the journals. Security reasons range from an evacuation of a terminal or concourse, reboarding of aircraft because of the security breach, inoperative screening equipment or long lines at screening areas in excess of mostly within an hour. Canceling flights due to weather is at the judgment of the carrier. Common causes of delays include tornados, blizzards, and hurricanes. National Aviation System (NAS) delays refer to a broad set of conditions including non-extreme weather, airport operations, heavy traffic volume, and air traffic control. These cancellations or delays are related to circumstances within the airline's control, including maintenance or crew problems, aircraft cleaning, baggage loading, and fueling. The most common cause of delays is late-arriving aircraft, which accounted for 41.9% of total tardiness minutes in 2014. This situation causes a ripple effect on other flights.

THANK YOU!!!