DSC465 Final Report
Group9

Stephen Montgomery
YunTzu Yu
Yifei Guo

A.Introduction

Our group chose to analyze flights & airline data collected and summarized by the Bureau of Transportation Statistics (BTS). The data we explored came from three locations of the BTS website. The three sources of data we explored each led to a focus on our three Part: 1.Location & seasonality of delays, 2. distribution of delays among airlines, and 3.the monetary effects of delays and cancellations.   The primary source of data was the BTS's airport summary data. This data gave information on the amount and type of delays and cancellations per airport, airline, and month. We used this data primarily in our geographic temporal analysis. The second source of data we used was the on time Flights data. This data was a more granular/detailed version of the primary data. The on time flights data gave the same information as the primary data but divided it by every flight that occurred. This allowed us to examine distributions of delay time per airports and airlines. Finally we combined our primary airport summary data with the BTS yearly revenue reports on airlines. This data included a total amount of cancellations and delays each airline had per year and their respective revenue for that year. Exploring this data led us to our third theme of cancellation and delay impacts on airline revenue. These three vignettes were used to tell our story of the impact of covid-19 on the airlines and will be explored in further detail in our exploratory and visualization sections.

B.Exploratory analysis & C.Explanatory Visualizations

Part 1.

Exploratory analysis Geographic

The exploratory analysis done for the geographic and temporal visualizations began with graphing the distribution of delays, and cancellations among all airports. This was done by grouping the data by year and air carriers then plotting the data using density, violin and box plots to visualize the distributions. This however did not work due to the airport with an extreme amount of flights present in the data. To amend this I choose to filter out the outliers. This effectively zoomed in the graphs to where most of the distribution occurred. **figures 1-3** show the distributions of the total amount of canceled flights per airport by month in 2019. **figures 4-6** show the same information as figures 1-3 but for delayed flights. After exploring the different distributions of delayed and canceled flights we decided to view how the amount of total flights per airline changed between 2019 & 2020. To achieve this we aggregated the amount of delayed and canceled flights by month per airline then filtered by the three most popular airlines (Delta, American & United). The resulting graphs of changes in total flight cancellations and delayed flights can be found in **figures 7-9**. To better compare the changes and patterns between the three airlines we decided to change the graphs from total flights per month to percent change from the first month. To do this we took each month's sum and divided it by the first month's sum to get a percent change in the amount of flights/delays/cancellations per month. The results are depicted in **figures 10 & 11**. By applying this technique we saw that the three airlines followed a very similar pattern where delays seem to decrease and cancellations seem to spike during the beginning of the pandemic. Next we looked at how the average Covid-19 case rate and amount of flights interacted with each other. To do this we combined data from the CDC website that depicted cases per month with the original data of flights per month. We then scaled/normalized the cases and total flights variables and plotted the results of the transformation for 2019-2020. Looking at the results in **figure 12** we can see the sharp decline in flights around the start of the pandemic along with the steady increase in cases as 2020 progressed. Lastly we decided to plot the amount of delays/cancellations and total flights per airport on top of a map of the USA. To do this we merged the flight summary statistics that included the amount of total, delayed and canceled flights per airport, month, and airline with another data source that included the latitude and longitude of every airport. This combined dataframe was then used alongside the R data in Statesmap to plot the amount of delays per location on top of a map of the USA. To get the amount of delays per location we grouped the data by year and airport then aggregated by the sum of delays. We then used the sum of delays and encoded it as size in the geo plot to display the relative amount of delays per airport. The results of this depiction can be found in **figure 13**. To go beyond comparing a single variable among all the airports we decided to plot both total flights and total cancellations per airport. To do this we used the same combined dataframe and grouped by year and airport then aggregated the total flights and cancellation columns. We then plotted both of these variables on top of the Statesmap USA depiction. To increase the visibility of the canceled flights we manually encoded the total flights as green and increased the transparency while also manually encoding the canceled flights as a dark red. This led to concentric circles for every airport with the larger green circle representing the total flights and the smaller red circle representing the total cancellations. The purpose of this graph was to visualize on a map the ratio of canceled flights to total flights. This graph can be viewed in **figure 14**. The graph did not achieve its purpose due to scaling issues. The scaling of the green circles led to a distortion that made the smaller airport seem to have a larger proportion of cancellations to overall flights. This phenomenon is known as Weber's law which explains that human perception focuses on percent changes in differences. In our case this visualization scaling makes it hard for the viewer to decode the relative difference of the green and red dots of the smaller airports. This difficulty could lead to the viewer falsely believing that the smaller airports have a higher percent cancellation rate then their larger counterparts. Due to this effect it was decided that only one variable should be encoded and plotted at a time on the map. Dissatisfied with this solution we choose to build an interactive tool that lets the user input the filter criteria to construct a coordinated view. By using interactivity and coordinated views we believed that we could accomplish our goal of showing multiple variables relative to each other (such as time, location, and flight types).

Explanatory Visualizations Geographic

In our geographic explanatory visualization, pictured in figure 21, we utilized interactivity to create a coordinated view that displays our message of how the pandemic affected flights within the US. The interactivity was achieved by using the Shiny App framework. We coded several drop menus along with a date slider which allows the user to select a month, airline, state and flight type. These selections are used to filter the data and display it in several locations within the coordinated view. The leftmost graph of the coordinated view is a timeline showing the trend of the filtered data over 2019 & 2020. The month filter is not applied to this data; instead it is used as a vertical dashed red line to show the user where they are in the overall timeline of the filtered data. The center graph is the geographic representation of the filtered data with the amount of selected flight type by airport. This variabel was double encoded as size and color. The right most graph is a percent distribution of flight types for the given filters. By using this app several patterns of the impact of the pandemic on flights become clear to the user. The first and most evident pattern is the sharp decrease in flights and increase in cancellations near the start of the pandemic. Additionally, if the user compares the flight type distributions in the pie chart from the beginning of the pandemic to the end of 2020 they will see that the percent share of total delayed flights decrease when compared to the beginning of the pandemic.

Our app's interactivity gives the coordinated view a rich layer of customization and personalization. By making the three graphs in the coordinated view fully interactive we have given the user the ability to construct the view that best suits their needs. The default view itself shows a clear message of the impact of Covid-19 on the amount of flights. Interacting with the tool the user can dig deeper into the data and decode more patterns such as the decrease in delayed flights as a percentage of overall flights during the start of the pandemic. By giving the user the ability to explore the data their way we have enabled them to choose and build the visualization that best enriches their understanding of the data. Furthermore, by giving the user a temporal, geographic, and distribution view of the filtered data we have given them the tool to enrich their understanding of the many potential patterns of the data. Additionally, having a coordinated view allows for side by side comparison. This side by side comparison increases their ability to decode and recognize any patterns between the views. Thus, having a coordinated view depicting the datas distribution, changes over time and location goes beyond simple visualizations to deliver an enriched and customizable view to the user.

To pick the color scales for the geoplot and the divergent colors in the pie graph we utilized the tools on the color brewer website. When designing the coordinated view we wanted to make a distinction between all flights Vs. diverted, canceled and delayed flight types. To do this we used green for all flights and red for the diverted, canceled and delayed flight types. We ensured that this color scheme was consistent across all our graphs that used color. We choose these colors because they are on opposite sides of the color wheel. We decided to double encode the size of the flight types because only having size or color would make decoding the differences between airports difficult. When encoding the amount of the flight types on the geoplot visualization we choose to use a continuous color scale as we were plotting a continuous variable. One challenge we ran into was choosing the background color for the geoplot. We had to find a color that would not obscure the light or dark green or red circles. To do this we chose blue because it was the color that was the furthest from both red and green on the color wheel and allowed the least interference with the continuous color scales.

To create the app we had to transform and combine the base BTS data with location data found from a different source. This combined data frame was then used as the basis to be filtered and aggregated based on the inputs from the shiny UI. The server took the inputs form the Shiny UI then filtered the combined data frame and aggregated it according to each view's requirements. The elements of each graph were taken control of and customized to provide the most efficient and easy to use experience. For example, when a user updates the filters the titles of each graph will update displaying exactly what the user has selected. Furthermore, we made changes to the legend location and size to increase the space for each graph in the coordinated view. In keeping with Gestalt's theory of grouping like objects together, we kept all graphs in the coordinated view in a fluid row at the top and all filters at the bottom. To further add to the distinction we added a horizontal black line separating the two rows.

By using our app the user may draw many conclusions from the data. The app allows the user to analyze the distributions of flight type, relative size of flight types per airport and trends of flight types from month to month. Analyzing these patterns it becomes evident that the start of the pandemic dramatically reduced the amount of flights while increasing the amount of cancellations. Additionally, there was a dramatic reduction in delayed flights as a percent of total flights in 2020 when compared to 2019 and prior.

Part 2.

Exploratory Analysis Boxplot and Heatmap

I put the feature of Percentage of delayed aircraft in a day in y- axis and feature of the airline in x-axis , so one point which is circled means that there were 20 % of the United airlines delayed on a certain day in 2019. I use the boxplot so that the user can see the quintile distribution of the delay percentage time. **See Figure 15**

I also created a Heatmap by tile plot,using the Percentage of delayed aircraft in a day as the factor of fill color.I chose the top 5 busiest airports to make the comparison among the airlines, so I put the airports in y axis versus the airlines in x axis. **See Figure 16**

Explanatory Analysis Temporal

Assuming we show the flight history in 2019 and 2020 to someone who is interested in airline performance, so that they could know which airport works the most efficiently or which airline performs best which means when people arrange their trip, they could take our data as reference to decide where to go and which airline to buy the ticket.
We would draw a plotbox to show my target audience which airline has the best performance on punctuality .

We could create a graph to plot the delay times of each airline in 2019 and 2020, so that users can make comparisons among the graphs to make the decision which airline ticket to buy.

If I were a customer, I would have seen that Jetblue Airline had the worst performance in 2019 and compared data in 2019 and 2020, awaring that though it permanfance has been improved a little bit, its performance still was the worst one among the airline. If I was to buy a ticket and on-time is important for me, I would have tried to avoid the Jetblue Airline Airlines.

Explanatory Analysis Heatmap

In addition to the feature of the delay time that the user would like to know, maybe they would take the airport in reference as well.
From the heatmap, I could tell that Delta Airline ＆Southwest Airline have relatively good performance on punctuality in 2019. In 2020, the Percentage of delayed aircraft in a day of All of the Airline has decreased, the reason for that maybe is because fewer people took the flights, the fewer passengers in each flight, so the flight could take off more punctually. Under this condition, Delta Airline ＆Southwest Airline still performed relatively well.
In conclusion, if I were a person who try to buy the flight ticket, Delta Airline ＆Southwest Airline would be my top 2 choice.

Part 3.

Explanatory Analysis Monetary

This part aims to find the financial effect of delay and cancellation through the exploratory analysis by graphing the total revenue variable with delays and cancellations variables in recent years. This process was done by selecting a useful and meaningful variable from the raw data by year and airline companies and then plotting the data mainly using line plots to visualize the possible correlation between the financial dimension and operating dimensions by the tool of Tableau.

To better explore the meanings behind variables, we attempts many variables as try to find potential pattern between finical records and cancellation or delay of airlines over recent years. We attempted to analysis the relation of how revenue changes with the change of delay and how revenue changes with the change of cancellation through plotting the lines chart with selecting three major airline companies, American Airline, Delta and United Airline.

And we found one patten that the shape of line of Revenue is more similar with shape of line of Cancellation compared with Delays as shown in **Figures 17, and 18**

Explanatory Visualizations Monetary

Thus, after amend the graphs for better explanatory visualization, we updates the Figure 17 and Figure 18 to new **Figure 19 and Figure 20** to better delivery the message from data, which the Y axis has dual variables of Total Revenue and cancellation or Delays and X axis is Year variable to show how Y varibales change in recent 10 years

When seen the **figure 19**, the lines has not any special or features with associated variables. But in **Figure 20**, the growth lines of revenues and cancellation of three airlines are similar especially in Covid period 2019,2020 and 2021. The lines of revenue(light colored) and cancellation (deep colored)of all 3 arline subgroups are similar in shape of "letter V" in a nearly parallel distance, which means the revenue and cancellation has a positive correlation. The revenue increase with cancellation increase at same time.

After do the research, we find one possible explanation. According to Demand-Supplay theory in Economics, due to the increase of flight cancellation, the supply amount of flight in the market will decrease and the demand from passagers and travelers remain same or increase as the price of flight ticket will increase as well as the seat occupancy rate will go up of single flight with the result of increased the income of airline. Also the fewer flight and higher seat occupancy rate could reduce the cost of airlines. Thus, the revenue line has positive correlation with cancellation.

D. Discussion

Geographic

Given more time we would have liked to convert the basic drawing of the USA map into a choropleth where each state would be shaded on a continuous scale based on the total number of Covid-19 cases that state had. This would allow the user to see both the change in covid cases and flights in one visualization. We would also consider adding the case rate to the timeline graph then normalizing the data so that the user could see the overall trends of case rates Vs. total flights.

Temporal

Everyone could obtain the flight data from the website, however, not everyone knows what it says.
Data visualization can grab our interest and keep our eyes on the message by drawing color and pattern. Data visualization helps to tell stories by curating data into a form easier to understand, highlighting the trends and outlier.
We even could allow our users the freedom to fully explore analyzed data by interactive data visualization.It is not easy to see the trend in terms of the financial or temporal of geography, however , after visualizing the raw data, we all could tell a good story to our user.

Monetary

With the principle of data, message and audience, the visualization of how delay and cancellation affect revenue could provide an idea of seeing the relationship between financial activities with the effect of operating activities data during covid-19 period. The target audience could be the airline companies' stakeholders, investors, or airline industry people who are interested in the financial performance of the dataset. However, the factors influenced the revenue certainly not only the cancellations. The further research work would focus on design and test the hypothesis found from monetary visualization.

Graph

Figure 1

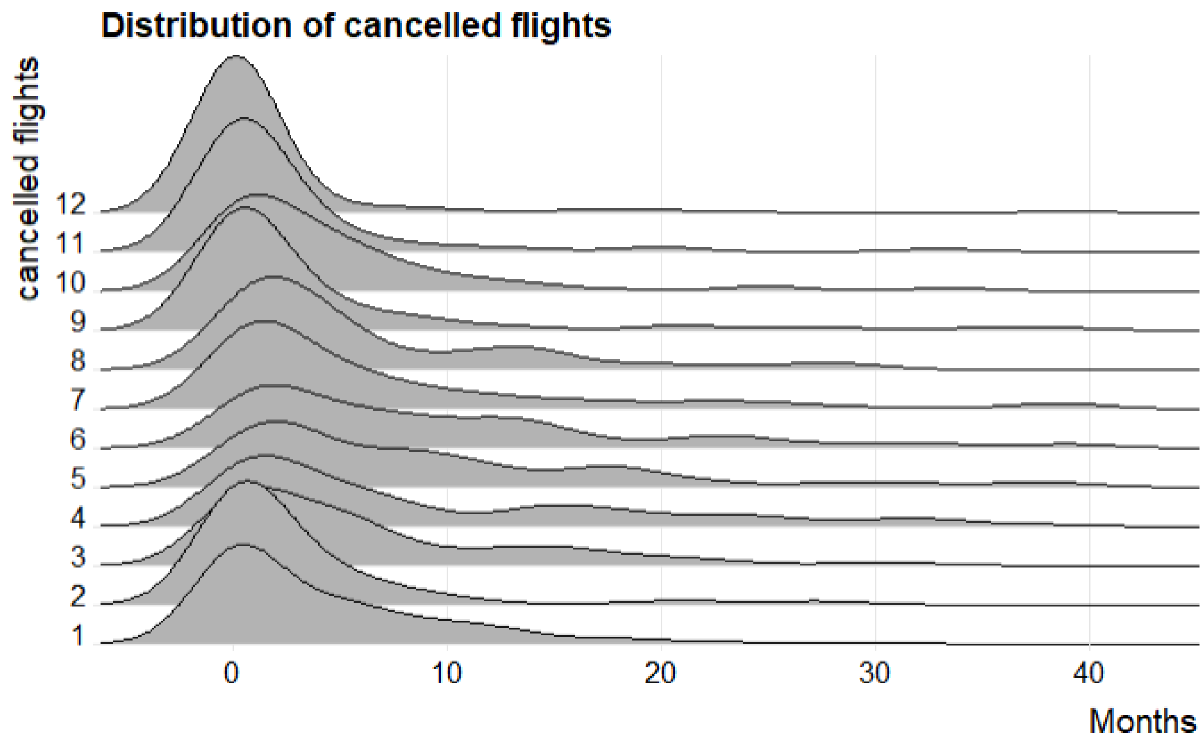


Figure 2

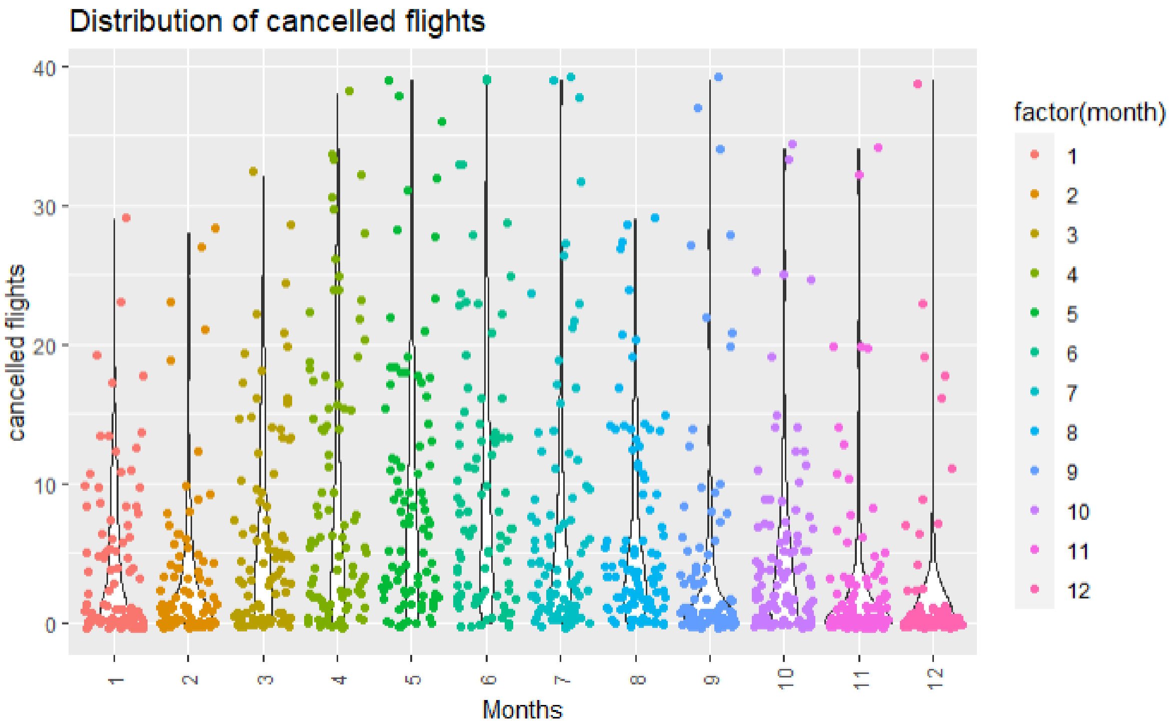


Figure 3

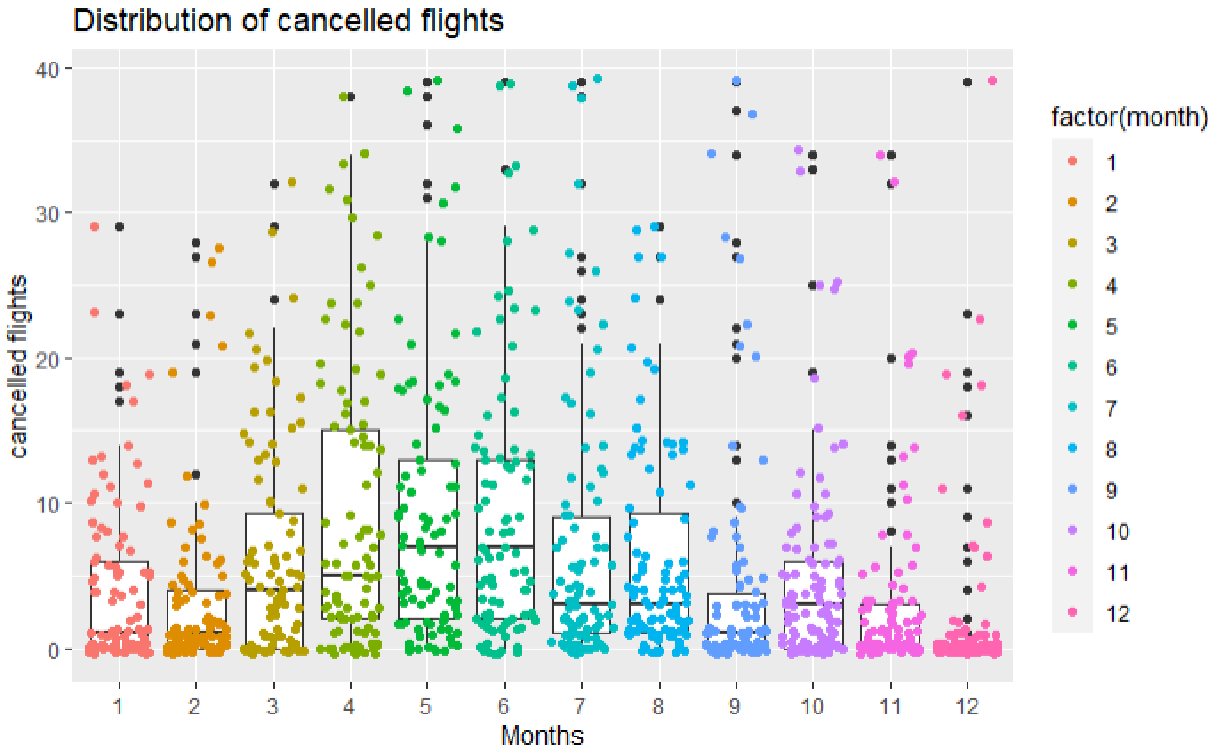


Figure 4

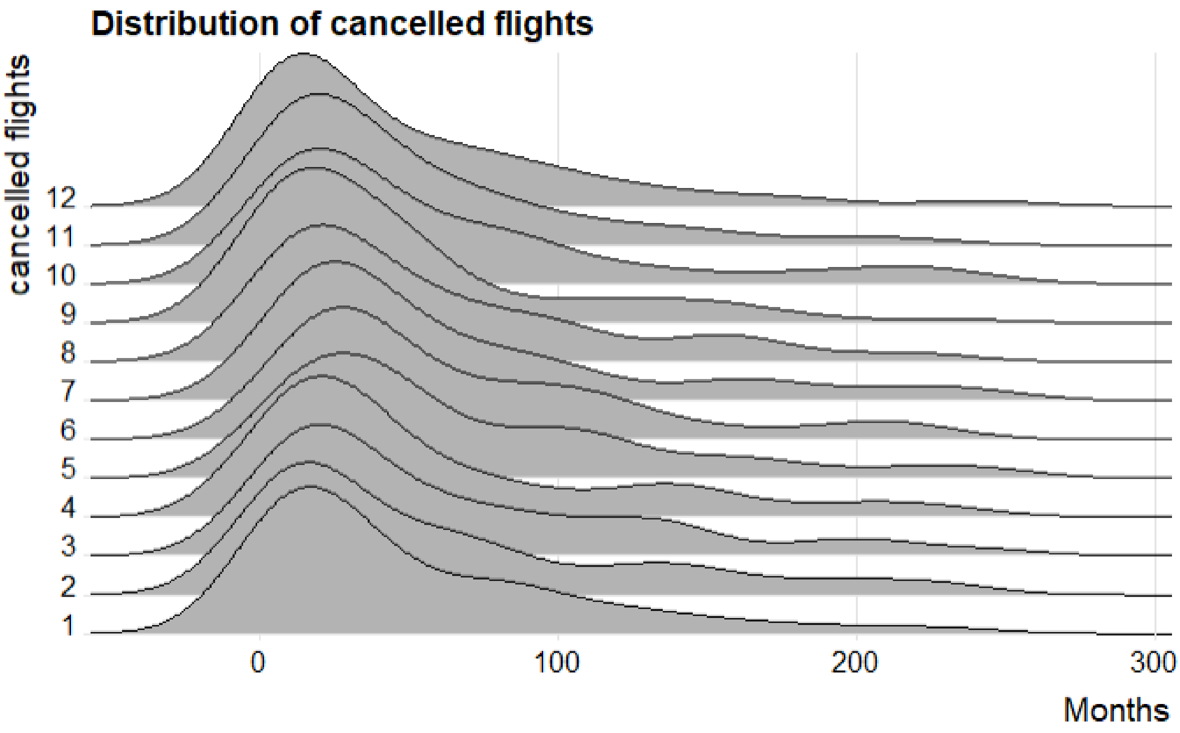


Figure 5

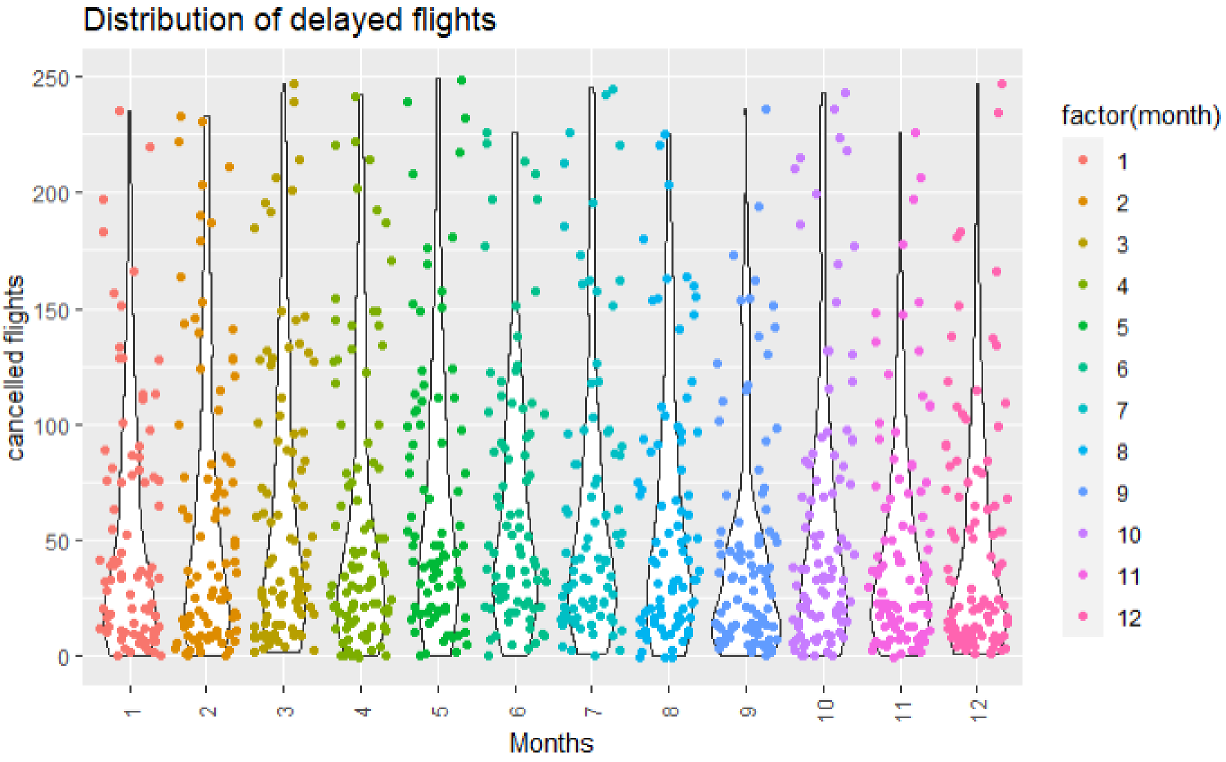


Figure 6

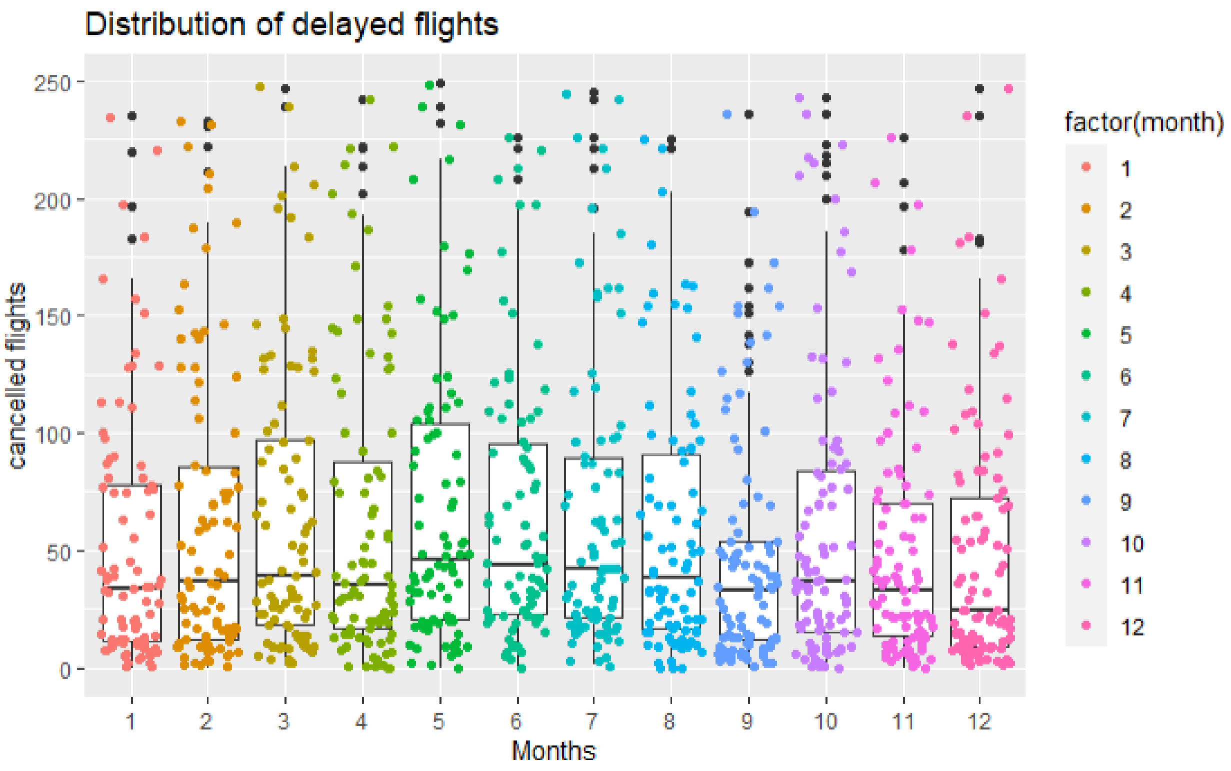


Figure 7

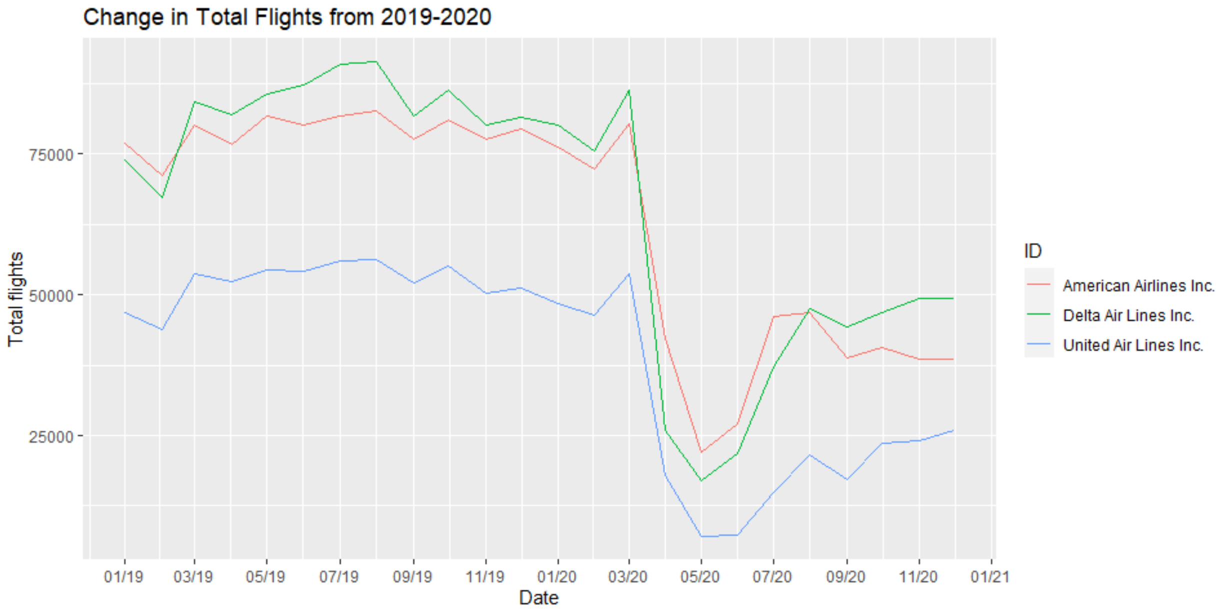


Figure 8

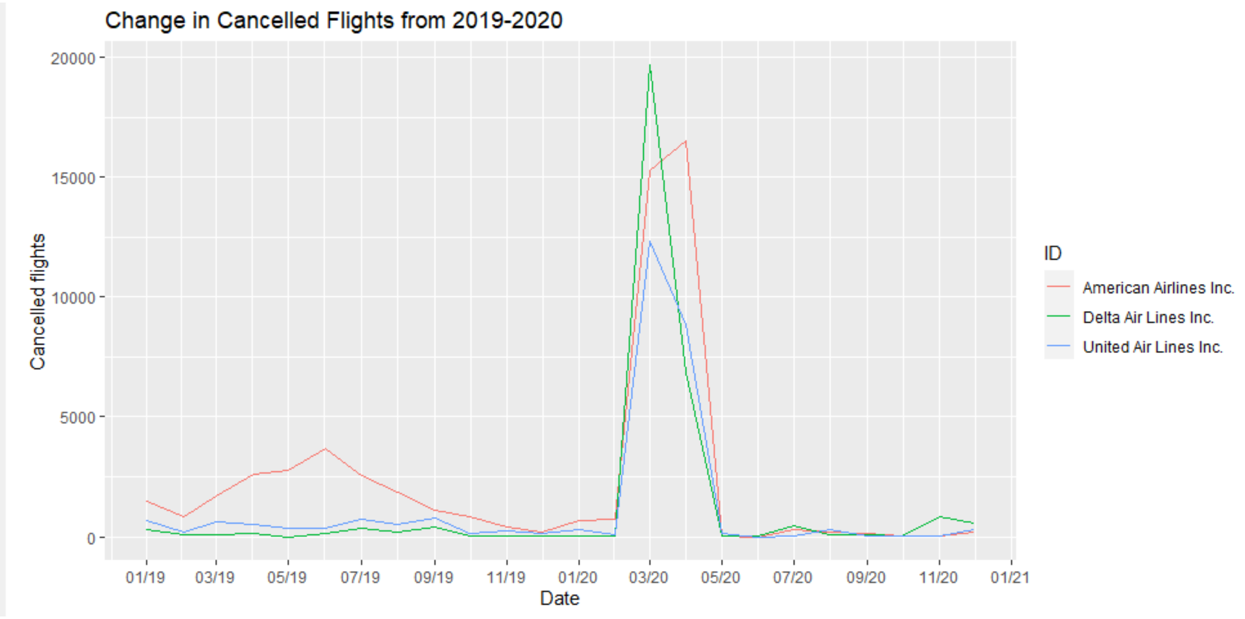

Change in Cancelled Flights from 2019-2020
Cancelled flights
20000
15000
10000
5000
0
01/19
03/19
05/19
07/19
09/19
11/19
01/20
03/20
05/20
07/20
09/20
11/20
01/21
Date
ID
American Airlines Inc.
Delta Air Lines Inc.
United Air Lines Inc.

Figure 9

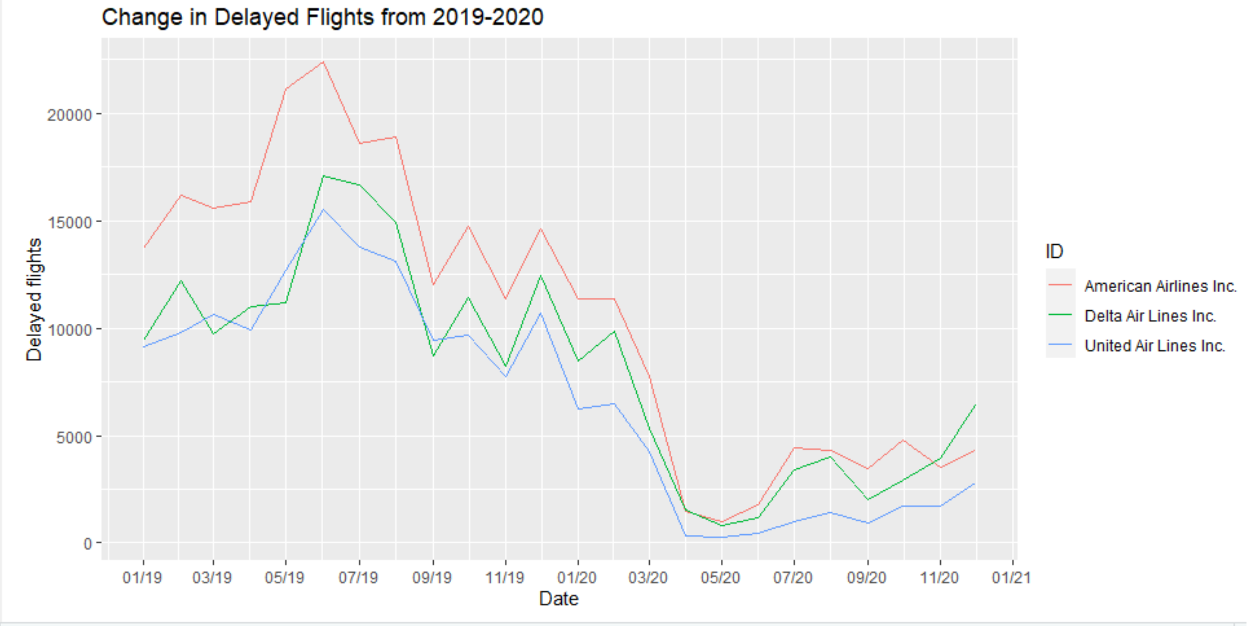

Change in Delayed Flights from 2019-2020
Delayed flights
20000
15000
10000
5000
0
01/19
03/19
05/19
07/19
09/19
11/19
01/20
03/20
05/20
07/20
09/20
11/20
01/21
Date
ID
American Airlines Inc.
Delta Air Lines Inc.
United Air Lines Inc.

Figure 10

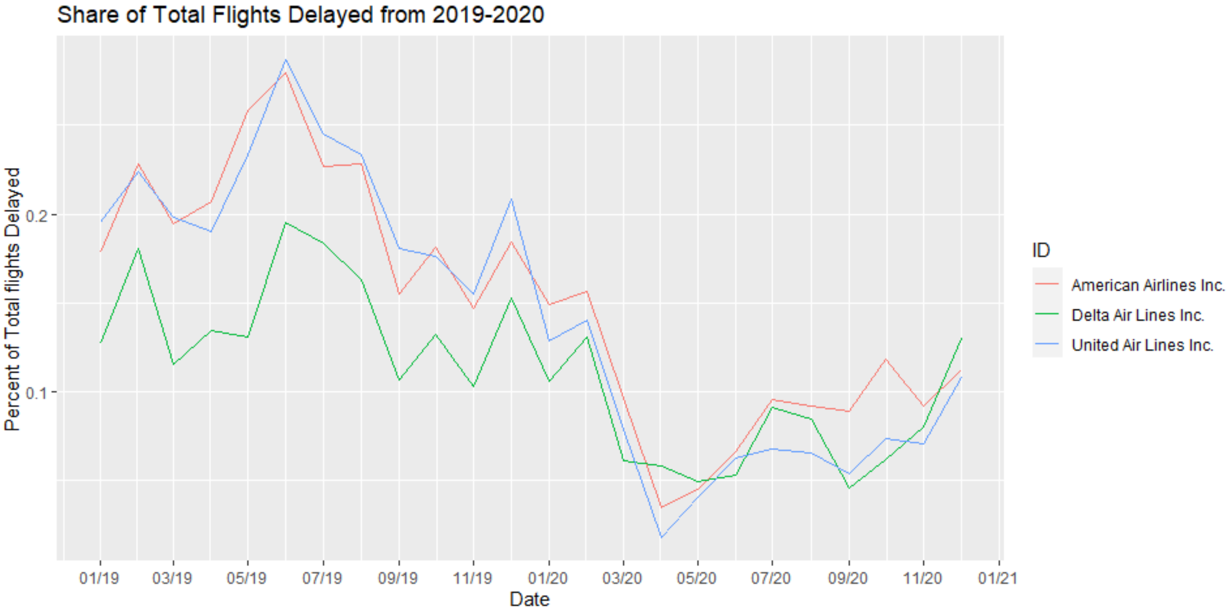


Figure 11

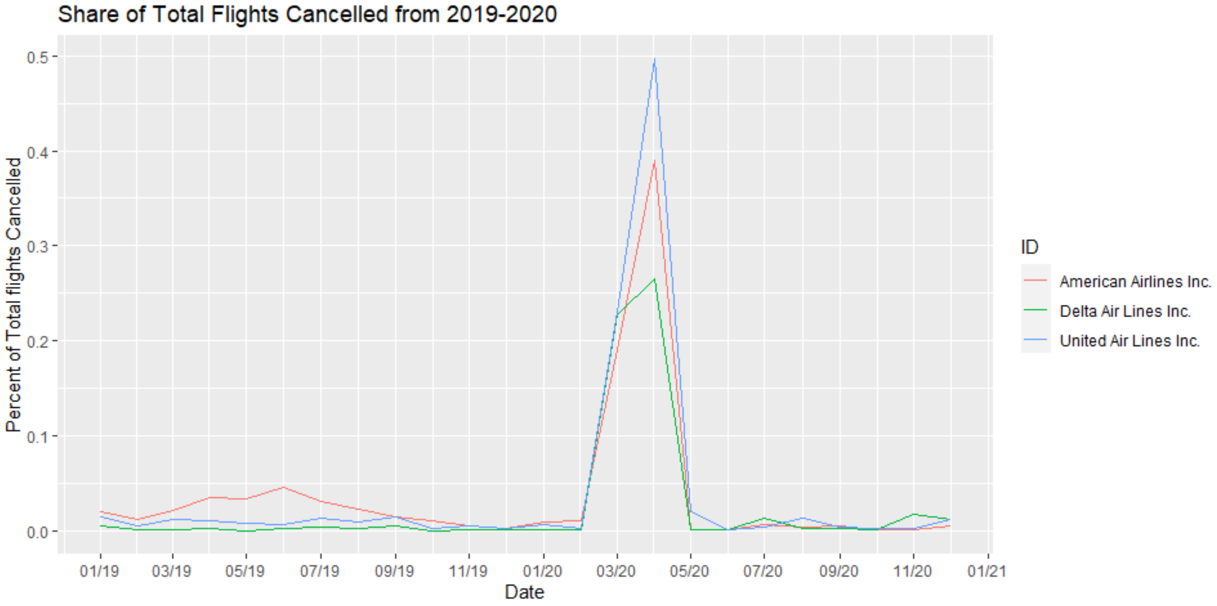


Figure 12

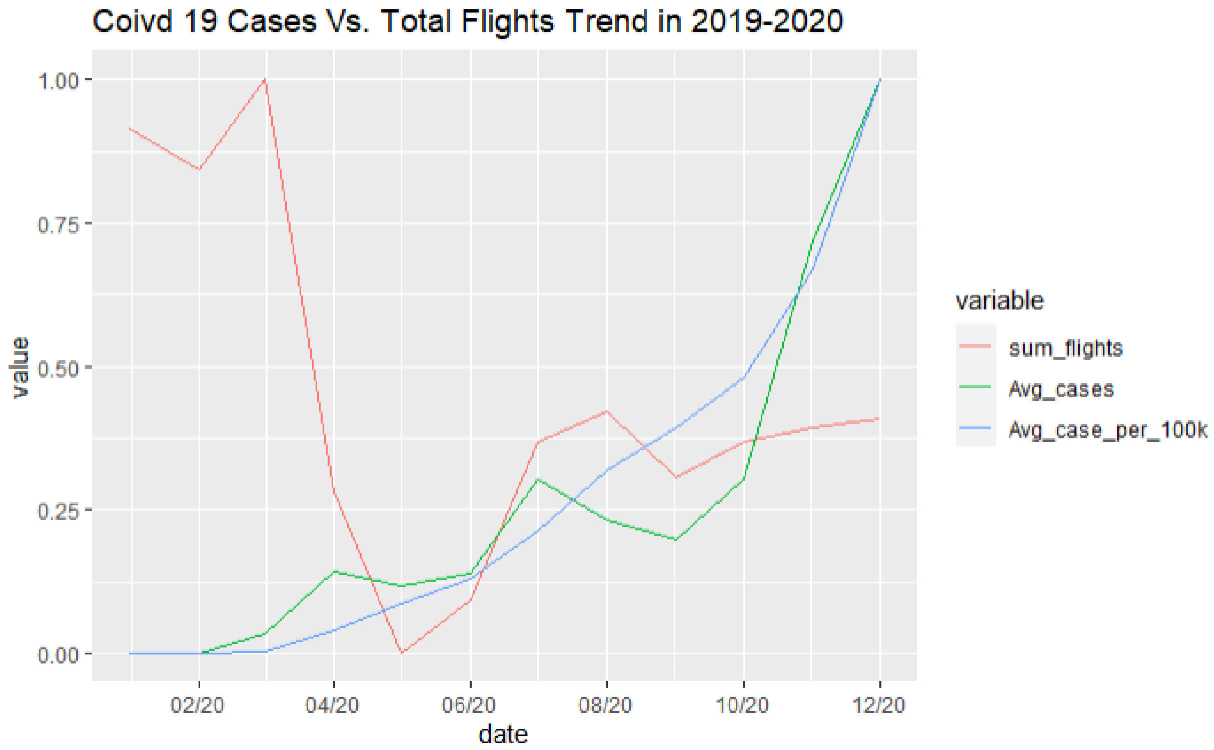

Coivd 19 Cases Vs. Total Flights Trend in 2019-2020
value
1.00
0.75
0.50
0.25
0.00
02/20
04/20
06/20
08/20
10/20
12/20
date
variable
sum_flights
Avg_cases
Avg_case_per_100k

Figure 13

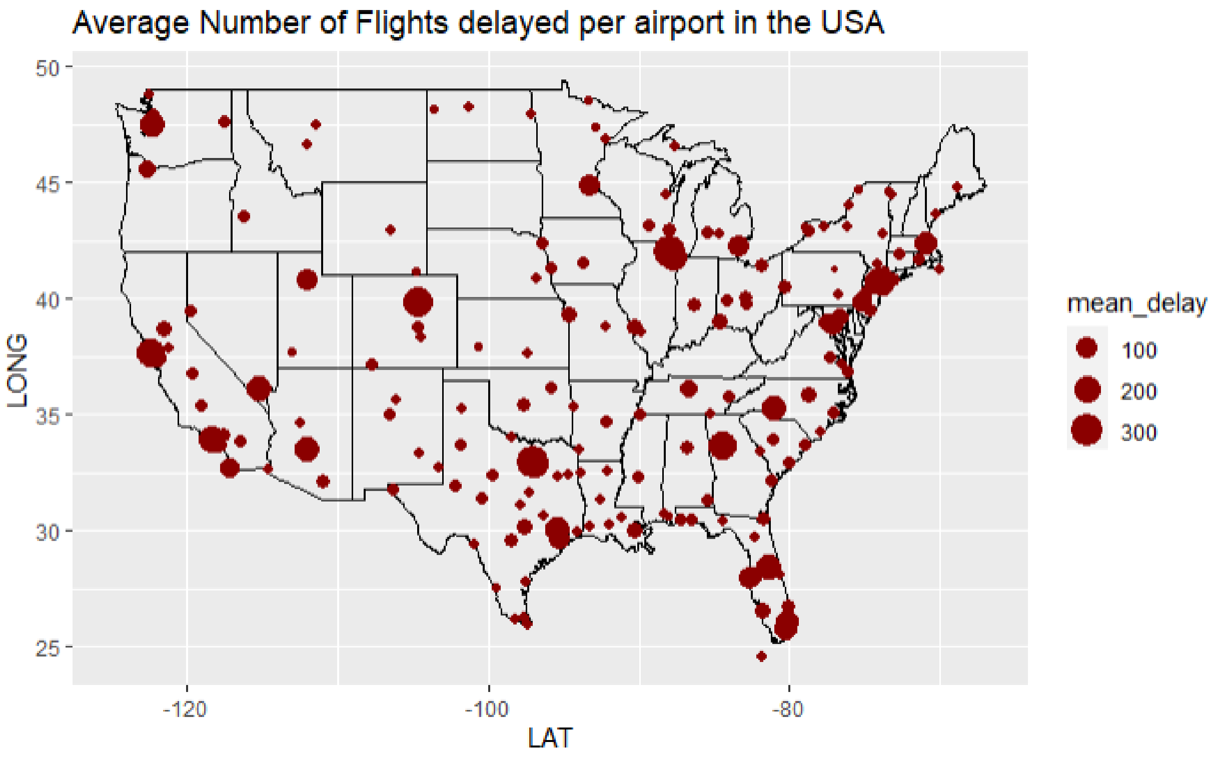

Average Number of Flights delayed per airport in the USA
LONG
50
45
40
35
30
25
-120
-100
-80
LAT
mean_delay
100
200
300

Figure 14

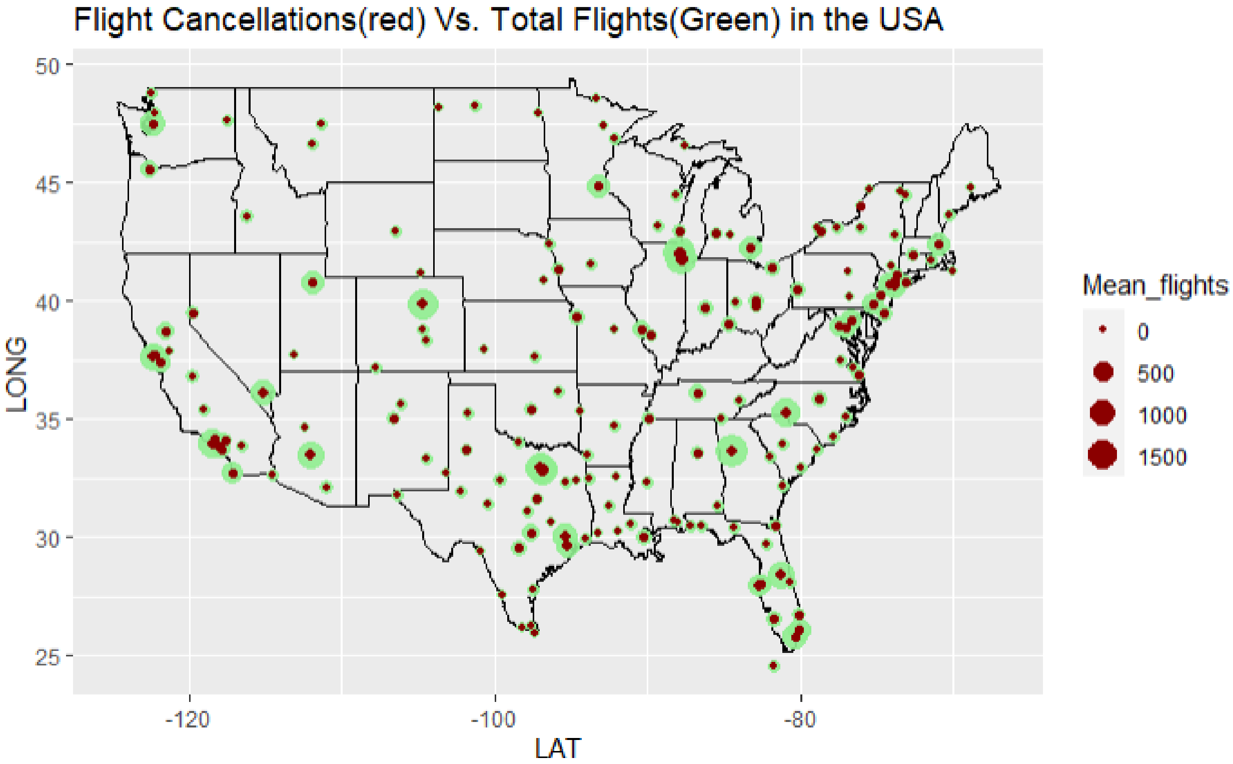
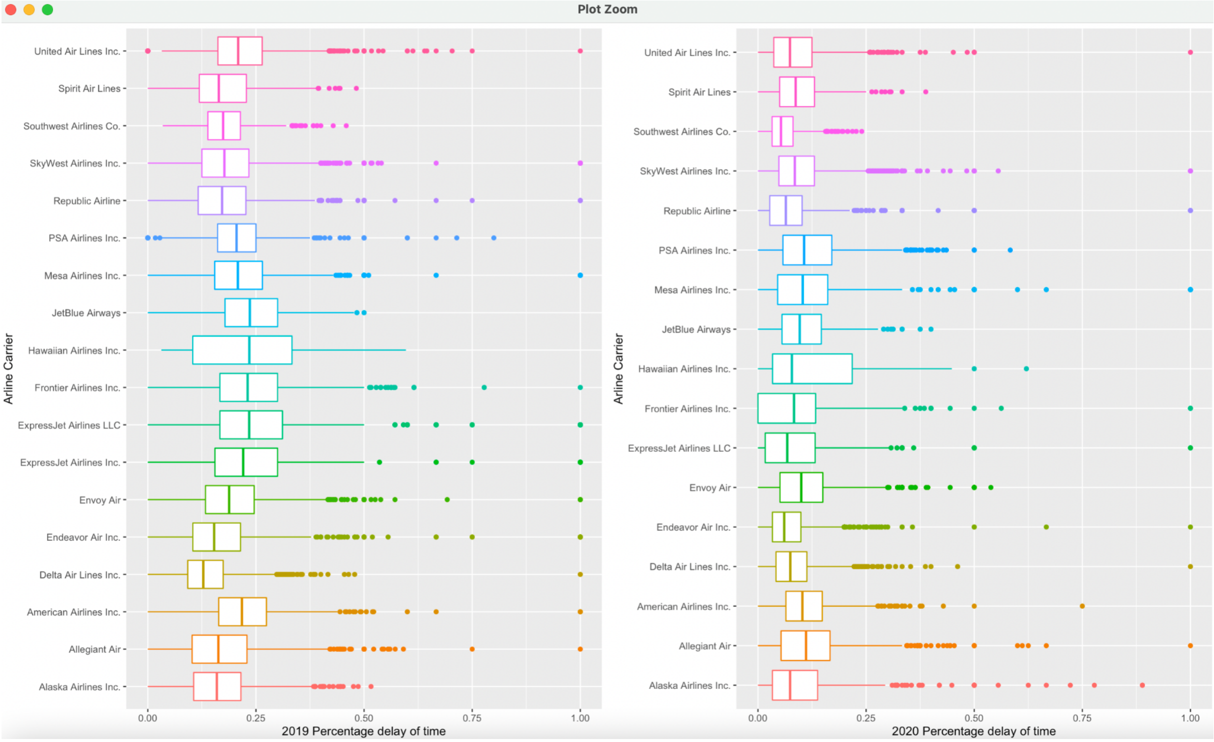


Figure 15


Plot Zoom
United Air Lines Inc.
Spirit Air Lines
Southwest Airlines Co.
SkyWest Airlines Inc.
Republic Airline
PSA Airlines Inc.
Mesa Airlines Inc.
JetBlue Airways
Hawaiian Airlines Inc.
Frontier Airlines Inc.
ExpressJet Airlines LLC
ExpressJet Airlines Inc.
Envoy Air
Endeavor Air Inc.
Delta Air Lines Inc.
American Airlines Inc.
Allegiant Air
Alaska Airlines Inc.
Arline Carrier
0.00
0.25
0.50
0.75
1.00
2019 Percentage delay of time
United Air Lines Inc.
Spirit Air Lines
Southwest Airlines Co.
SkyWest Airlines Inc.
Republic Airline
PSA Airlines Inc.
Mesa Airlines Inc.
JetBlue Airways
Hawaiian Airlines Inc.
Frontier Airlines Inc.
ExpressJet Airlines LLC
Envoy Air
Endeavor Air Inc.
Delta Air Lines Inc.
American Airlines Inc.
Allegiant Air
Alaska Airlines Inc.
Arline Carrier
0.00
0.25
0.50
0.75
1.00
2020 Percentage delay of time

Figures 16

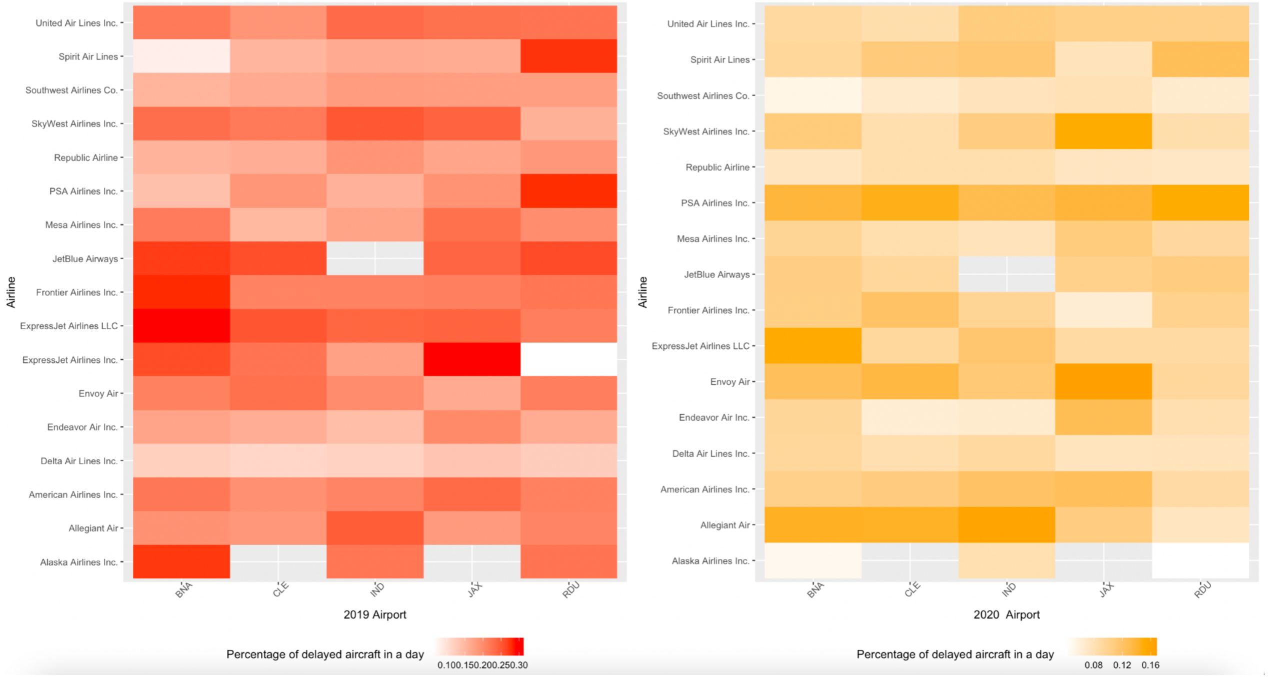

United Air Lines Inc.
Spirit Air Lines
Southwest Airlines Co.
SkyWest Airlines Inc.
Republic Airline
PSA Airlines Inc.
Mesa Airlines Inc.
JetBlue Airways
Frontier Airlines Inc.
ExpressJet Airlines LLC
ExpressJet Airlines Inc.
Envoy Air
Endeavor Air Inc.
Delta Air Lines Inc.
American Airlines Inc.
Allegiant Air
Alaska Airlines Inc.
Airline
BNA
CLE
IND
JAX
RDU
2019 Airport
Percentage of delayed aircraft in a day
0.10 0.15 0.20 0.25 0.30
United Air Lines Inc.
Spirit Air Lines
Southwest Airlines Co.
SkyWest Airlines Inc.
Republic Airline
PSA Airlines Inc.
Mesa Airlines Inc.
JetBlue Airways
Frontier Airlines Inc.
ExpressJet Airlines LLC
Envoy Air
Endeavor Air Inc.
Delta Air Lines Inc.
American Airlines Inc.
Allegiant Air
Alaska Airlines Inc.
Airline
BNA
CLE
IND
JAX
RDU
2020 Airport
Percentage of delayed aircraft in a day
0.08
0.12
0.16

Figure 17

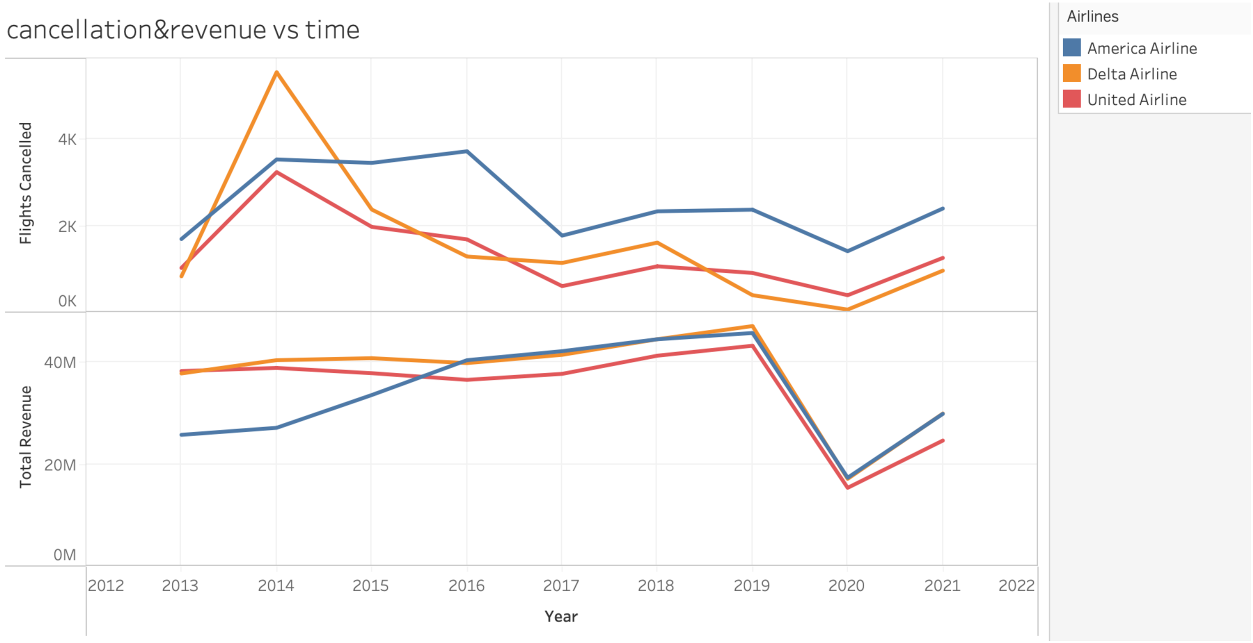
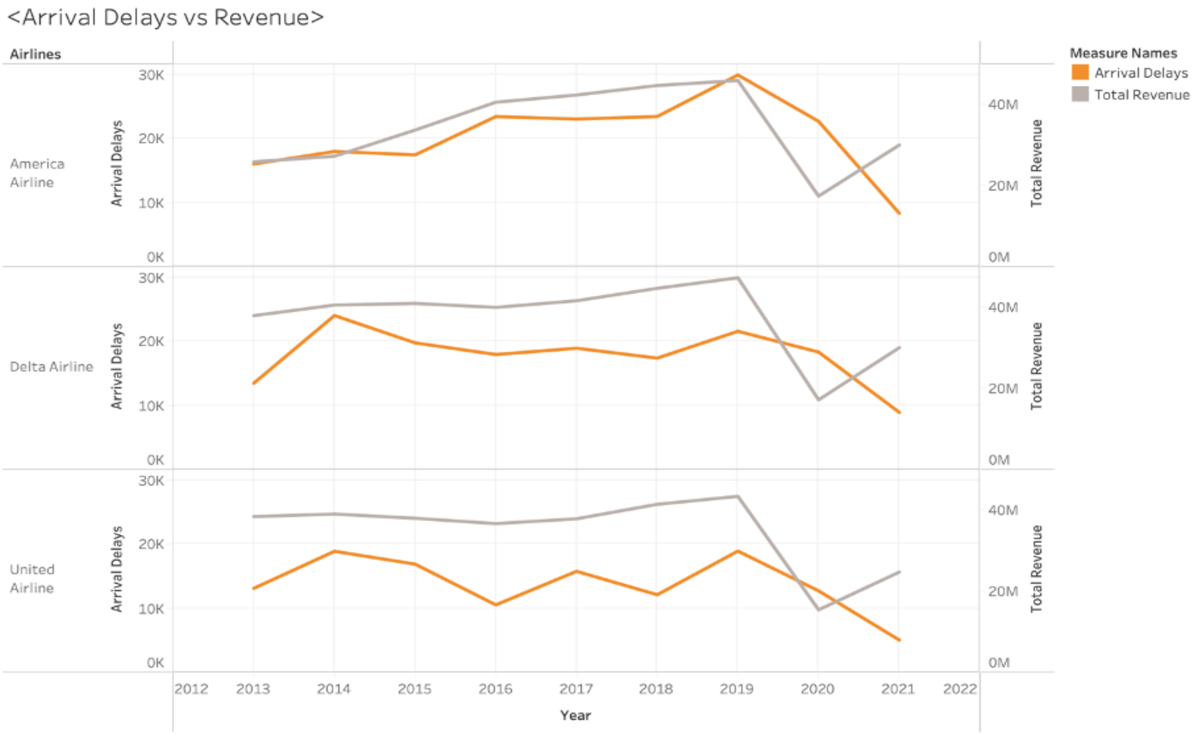
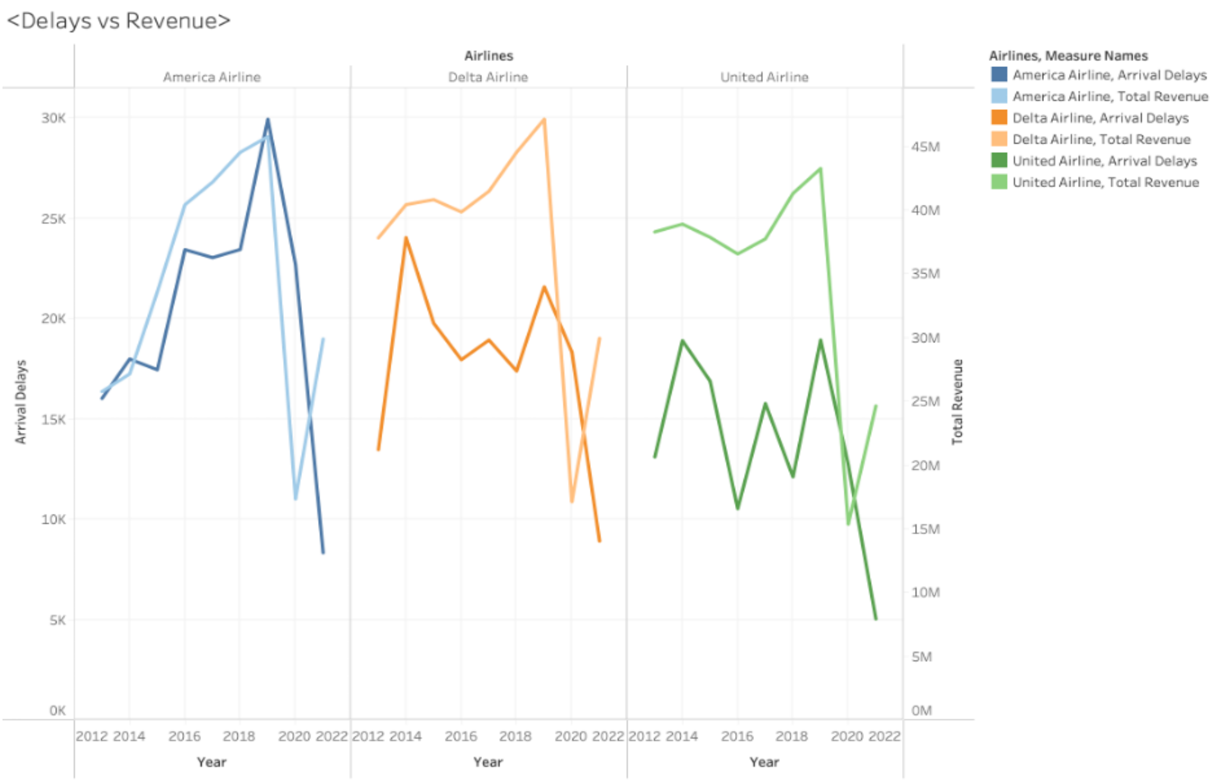


Figure18



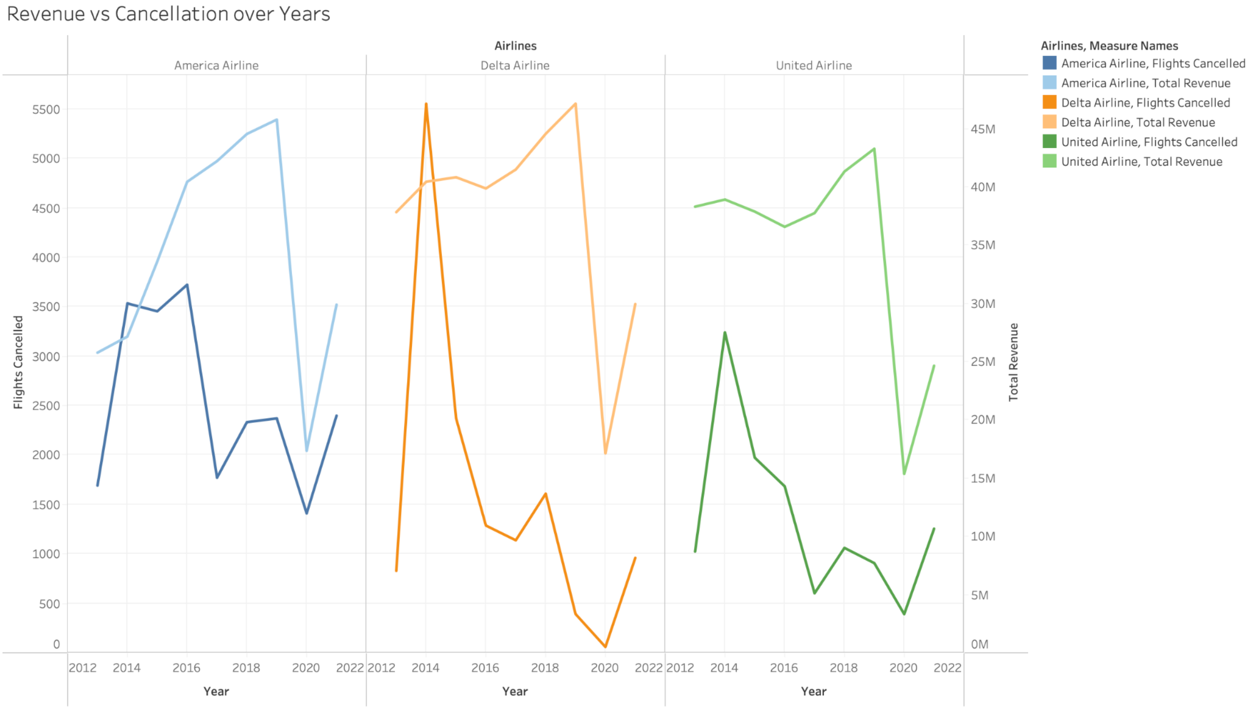
The trends of Arrival Delays and Total Revenue for Year broken down by Airlines. Color shows details about Arrival Delays and Total Revenue. The view is filtered on Airlines, which excludes Null.

Figure 19.



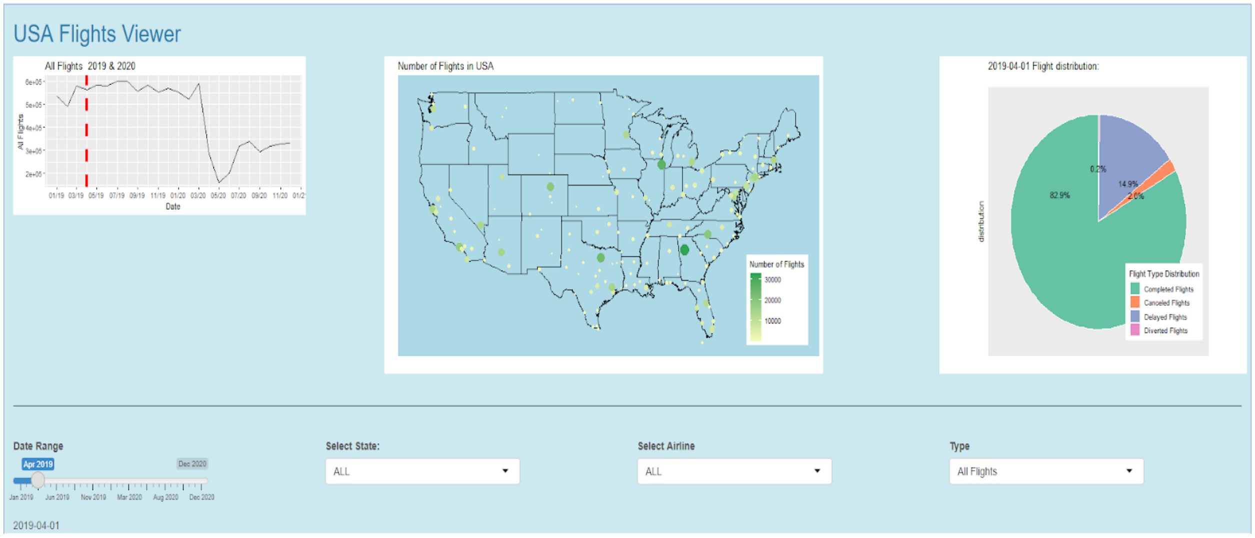
The trends of Arrival Delays and Total Revenue for Year broken down by Airlines. Color shows details about Airlines, Arrival Delays and Total Revenue.

Figure 20.



The trends of Flights Cancelled and Total Revenue for Year broken down by Airlines. Color shows details about Airlines, Flights Cancelled and Total Revenue.

Figure 21



USA Flights Viewer

Date Range: Apr 2019 – Dec 2020

Select State: ALL

Select Airline: ALL

Type: All Flights

2019-04-01

E. Appendices

Individual Report:

Stephen Montgomery Personal Write up

I was the group coordinator. Along with the help of my teammates I coordinated times for us all to meet weekly to discuss our findings and next steps in the group project. I also was the channel of communication between our group and the professor. I helped create the outline for our final report. I was in charge of all the exploratory graphs for the geoplot and shiny app. I wrote the code for figures 1-14 as well as the sections of the report that pertained to the geographic exploratory and explanatory visualizations. I also wrote and refined all the code for the shiny app itself. Through creating the shiny app I learned how important color is as a communication medium. Out of all the coding and formatting I had to do for the shiny app I spent the most time trying to pick the right colors that worked well with each other but did not distract or interfere with the various encodings. I learned what colors would work best for continuous vs categorical variables. I also learned how important and useful creating clear separations between grouped objects in a visualization could be. Finally I learned how to use and implement the tools provided by the website color brewer.

YunTzu Personal Write up

I was in charge of the heatmap and boxplot to explore the correlation between the feature of delay time , airline carrier and airport. I wrote the code for figures 15-16 and did the presentation of the sections that were relative to the temporal exploratory . I learned that we could easily see the distribution by box plot , it showed the quantile information so I could easily decode the graph and was not affected by outlier values to read the data.On the other hand, I learned how important the colors could help to decode the information on the heatmap. In accordance with the colors changed on legend, we could easily find out the relative extreme value on the heatmap. In the exploratory stage , we could visualize the data to help us find the trend so that we could take further action to do the exlanatory data analysis.

Yifei Personal Write up

I was mainly contributed on the part3 of monetary analysis of data to explore the correlation between the feature of total revenue and cancalleation or delay time of airlines in recent yearst, which finished with outcome on figures 17-20. And I made the the presentation slides, formatted and orgranzed the structure of final report. I learned a lot from every communicaton and discussion from my teammate of all meeting times. Also, I learned how to use tool to design appropriate graphs and delivery messages to audience with good visualizaton of data informaton through exploratory and explanatory analysis and also evaluate the work with both positive and negtive perspectives.

Github source:

Boxplot and Heatmap link :
https://github.com/yuyuntzu/DSC-465-PROJECT/blob/8399457acd8c299280793baf6791857c33b4cbad/Airline%20delay%20time.R

Code scripts:

Exploratory analysis Geographic Code

~~~
---
title: "Data Viz Project Explorotory Graphs"
output: html_notebook
---
```{r}
# Load the libraries
library(dbplyr)
library(ggplot2)
library(ggridges)
library(mosaic)
library(lubridate)
library(imputeTS)
library(tidyverse)
library(mapproj)
library(reshape2)
library(caret)

~~~

#Pre-Processing
```{r}

# Pull data contaning the Airport Code and respective lat/long values
setwd("C:/Users/montg/Desktop/DSC 465 Project data")
Airport_Zips<-read.table("Airport_Zip_Codes.txt",
        header = FALSE,    # Whether to display the header (TRUE) or not (FALSE)
        sep = ":",         # Separator of the columns of the file
        dec = ".",
        blank.lines.skip = FALSE,
        quote="",
        comment.char="")        # Character used to separate decimals of the numbers in the file

# Filter the data to get only USA Aiports that have data for lat/long & IATA codes
Airport_Zips <-Airport_Zips %>%
```

~~~
  filter(V5=='USA') %>%
  filter(V2!='N/A') %>%
  filter(V3!='N/A') %>%
  select(V1,V2,V3,V4,V5,V15,V16)%>%
  na.omit()


Flights_df <- read.csv("Delay_by_Airline.csv")

FUN <- function(arg_1) {
  Test=substr(str_extract(arg_1,"..:"),1,2)
  return(Test)
}

Flights_df$State_Abrv <- lapply(Flights_df$airport_name, FUN)

combined_df<-merge(Flights_df,Airport_Zips,by.x="airport",by.y="V2")
combined_df_char=combined_df
combined_df_char[]<- lapply(combined_df, as.character)
write.csv(combined_df_char,"C:/Users/montg/Desktop/DSC 465 Project data/combined_df.csv",
row.names = TRUE)
```

#histogram ridgeline for the distributions over time of cancellations
```{r}
AA_Flights_df<-Flights_df%>%
  filter(carrier=='AA') %>%
  filter(year=='2019') %>%
  select(arr_cancelled,month)%>%
  na.omit()


AA_Flights_df_zoomedin<-AA_Flights_df%>%
  filter(arr_cancelled<40)

ggplot(AA_Flights_df_zoomedin, aes(x = arr_cancelled, y = as.factor(month))) +
  geom_density_ridges(scale = 4) +
  scale_y_discrete(expand = c(0, 0)) +     # will generally have to set the `expand` option
  scale_x_continuous(expand = c(0, 0)) +   # for both axes to remove unneeded padding
  coord_cartesian(clip = "off") +          # to avoid clipping of the very top of the top ridgeline
  theme_ridges()+
  labs(x='Months',
          y='cancelled flights',
          title="Distribution of cancelled flights")
~~~

~~~
ggplot(AA_Flights_df_zoomedin,aes(x=factor(month), y=arr_cancelled))+
  geom_violin() +
  geom_jitter(aes(color=factor(month)))+
  theme(axis.text.x = element_text(angle = 90, hjust = .5, vjust = 0.5))+
  labs(x='Months',
           y='cancelled flights',
           title="Distribution of cancelled flights")

ggplot(AA_Flights_df_zoomedin,aes(x=factor(month), y=arr_cancelled))+
  geom_boxplot() +
  geom_jitter(aes(color=factor(month)))+
  labs(x='Months',
           y='cancelled flights',
           title="Distribution of cancelled flights")
```

#histogram ridgeline for the distributions over time of number of delays
```{r}
AA_Flights_df<-Flights_df%>%
  filter(carrier=='AA') %>%
  filter(year=='2019') %>%
  select(arr_del15,month)%>%
  na.omit()


AA_Flights_df_zoomedin<-AA_Flights_df%>%
  filter(arr_del15<250)

ggplot(AA_Flights_df_zoomedin, aes(x = arr_del15, y = as.factor(month))) +
  geom_density_ridges(scale = 4) +
  scale_y_discrete(expand = c(0, 0)) +     # will generally have to set the `expand` option
  scale_x_continuous(expand = c(0, 0)) +   # for both axes to remove unneeded padding
  coord_cartesian(clip = "off") +          # to avoid clipping of the very top of the top ridgeline
  theme_ridges()+
  labs(x='Months',
           y='cancelled flights',
           title="Distribution of cancelled flights")

ggplot(AA_Flights_df_zoomedin,aes(x=factor(month), y=arr_del15))+
  geom_violin() +
  geom_jitter(aes(color=factor(month)))+
  theme(axis.text.x = element_text(angle = 90, hjust = .5, vjust = 0.5))+
  labs(x='Months',
~~~

~~~
             y='cancelled flights',
             title="Distribution of delayed flights")

ggplot(AA_Flights_df_zoomedin,aes(x=factor(month), y=arr_del15))+
  geom_boxplot() +
  geom_jitter(aes(color=factor(month)))+
  labs(x='Months',
             y='cancelled flights',
             title="Distribution of delayed flights")
```

# AA UAL & DELTA STOCK RPICES
```{r, fig.width=10,fig.height=5}
AA <- read.csv("UAL Stock Prices.csv")
Start=AA[1,2]
AA <- AA %>%
  mutate(ID="American Airlines Inc.")%>%
  mutate(Percent_change=Open/Start)
UAL<- read.csv("AAL Stock Price.csv")
Start=UAL[1,2]
UAL <- UAL %>%
  mutate(ID="United Air Lines Inc.")%>%
  mutate(Percent_change=Open/Start)
DAL<- read.csv("DAL stock Price.csv")
Start=DAL[1,2]
DAL <- DAL %>%
  mutate(ID="Delta Air Lines Inc.")%>%
  mutate(Percent_change=Open/Start)

Stock_prices= rbind(AA,UAL,DAL)

Stock_prices <- Stock_prices %>%
        mutate(Date = as.Date(Date, "%m/%d/%Y"))

ggplot(data=Stock_prices,aes(x=Date,y=Open,group=ID,color=ID))+
  geom_point()+
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")

```

# Stock prices as percent of starting values
```{r, fig.width=10,fig.height=5}
ggplot(data=Stock_prices,aes(x=Date,y=Percent_change,group=ID,color=ID))+
  geom_point()+
~~~

```
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")
```

#stock price Vs. total amount of flights/cancellations/delays/%delays/%cancellations per airline per month
```{r}
Flights_monthly_agg<-Flights_df%>%
  na.omit()%>%
  group_by(year,month,carrier_name)%>%

summarise(sum_flights=sum(arr_flights),sum_cancel=sum(arr_cancelled),sum_delayed=sum(arr_del15),Mean_flights=mean(arr_flights),mean_cancel=mean(arr_cancelled),mean_delayed=mean(arr_del15))%>%
  filter(carrier_name %in% c("United Air Lines Inc.","Delta Air Lines Inc.","American Airlines Inc."))

Monthly_stock_price<-Stock_prices%>%
  na.omit()%>%
  mutate(month=format(Date, "%m"))%>%
  mutate(year=format(Date, "%Y"))%>%
  group_by(year,month,ID)%>%
  summarise(avg_price=mean(Open))%>%
  mutate(year=as.numeric(year))%>%
  mutate(month=as.numeric(month))

colnames(Flights_monthly_agg)[3] <- "ID"
Stocks_Flights<-merge(Flights_monthly_agg,Monthly_stock_price,by=c("ID","month","year"))
Stocks_Flights<-Stocks_Flights%>%
  mutate(percent_delay=sum_delayed/sum_flights)%>%
  mutate(percent_cancel=sum_cancel/sum_flights)
ggplot(data=Stocks_Flights,aes(x=sum_flights,y=avg_price,group=ID,color=ID))+
  geom_point()
ggplot(data=Stocks_Flights,aes(x=percent_delay,y=avg_price,group=ID,color=ID))+
  geom_point()
ggplot(data=Stocks_Flights,aes(x=percent_cancel,y=avg_price,group=ID,color=ID))+
  geom_point()
```

#Cancellations/dealys/Percent delays/Percent Cancellations over time
```{r, fig.width=10,fig.height=5}
```

~~~
Flights_monthly_agg=Flights_monthly_agg%>%
  mutate(percent_delay=sum_delayed/sum_flights)%>%
  mutate(percent_cancel=sum_cancel/sum_flights)%>%
  mutate(day = 1)%>%
  mutate(year=as.numeric(year))%>%
  mutate(month=as.numeric(month))%>%
  mutate(date = make_date(year, month, day))

ggplot(data=Flights_monthly_agg,aes(x=date,y=sum_flights,group=ID,color=ID))+
  geom_line()+
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")+
  labs(x='Date',
           y='Total flights',
           title="Change in Total Flights from 2019-2020")

ggplot(data=Flights_monthly_agg,aes(x=date,y=sum_cancel,group=ID,color=ID))+
  geom_line()+
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")+
  labs(x='Date',
           y='Cancelled flights',
           title="Change in Cancelled Flights from 2019-2020")

ggplot(data=Flights_monthly_agg,aes(x=date,y=sum_delayed,group=ID,color=ID))+
  geom_line()+
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")+
  labs(x='Date',
           y='Delayed flights',
           title="Change in Delayed Flights from 2019-2020")

ggplot(data=Flights_monthly_agg,aes(x=date,y=percent_delay,group=ID,color=ID))+
  geom_line()+
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")+
  labs(x='Date',
           y='Percent of Total flights Delayed',
           title="Share of Total Flights Delayed from 2019-2020")

ggplot(data=Flights_monthly_agg,aes(x=date,y=percent_cancel,group=ID,color=ID))+
  geom_line()+
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")+
  labs(x='Date',
           y='Percent of Total flights Cancelled',
           title="Share of Total Flights Cancelled from 2019-2020")
```
~~~

~~~
#geo plot of cancellations per location encoding the delay size or amount as size of the circle at the location delay
```{r}
agg_df<-combined_df%>%
  na.omit()%>%
  filter(V16>-125)%>%
  group_by(airport)%>%

summarise(Mean_flights=mean(arr_flights),mean_cancel=mean(arr_cancelled),mean_delay=mean(arr_del15),long=mean(V16),lat=mean(V15))
statesmap = map_data('state')
ggplot() +
  geom_polygon(data=statesmap,
               aes(x=long, y=lat, group=group),
               colour='black',
               fill=NA)+
  geom_point(data=agg_df,
             aes(long,lat,size=Mean_flights),
             alpha=.9,color='light green')+
  geom_point(data=agg_df,
             aes(long,lat,size=mean_cancel),
             color='dark red')+


 labs(x='LAT',
       y='LONG',
       title='Flight Cancellations(red) Vs. Total Flights(Green) in the USA'
       )
```


#geo plot of delays per location encoding the delay size or amount as size of the circle at the location delay
```{r}
ggplot() +
  geom_polygon(data=statesmap,
               aes(x=long, y=lat, group=group),
               colour='black',
               fill=NA)+
  geom_point(data=agg_df,
             aes(long,lat,size=mean_delay),
             color='dark red')+
 labs(x='LAT',
~~~

~~~
      y='LONG',
      title='Average Number of Flights delayed per airport in the USA'
      )
```

#Covid Cases VS Total Flights/Cancelled Flights
```{r}
Covid_df<-read.csv('covid.csv')
Covid_df<-Covid_df%>%
  mutate(Date=as.Date(Date,'%B %d %Y'))%>%
  mutate(month=as.numeric(format(Date,'%m')))%>%
  mutate(year=as.numeric(format(Date,'%Y')))%>%
  group_by(year,month)%>%

summarise(Avg_cases=mean(New.Cases),Avg_case_per_100k=mean(Total.Case.Rate.per.100k))

ALL_Flights_monthly_agg<-Flights_df%>%
  na.omit()%>%
  group_by(year,month)%>%

summarise(sum_flights=sum(arr_flights),sum_cancel=sum(arr_cancelled),sum_delayed=sum(arr_del15),Mean_flights=mean(arr_flights),mean_cancel=mean(arr_cancelled),mean_delayed=mean(arr_del15))

Covid_flights=merge(ALL_Flights_monthly_agg,Covid_df,by=c("month","year"))

Covid_flights<-Covid_flights%>%
  mutate(day = 1)%>%
  mutate(year=as.numeric(year))%>%
  mutate(month=as.numeric(month))%>%
  mutate(date = make_date(year, month, day))%>%
  ungroup()%>%
  mutate(sum_flights=rescale(sum_flights))%>%
  mutate(Avg_cases=rescale(Avg_cases))%>%
  mutate(Avg_case_per_100k=rescale(Avg_case_per_100k))

Covid_flights.melted=melt(Covid_flights,id.var='date',var.name='date')
Covid_flights.melted<-Covid_flights.melted%>%
  filter(variable=='sum_flights' | variable=='Avg_cases' | variable=='Avg_case_per_100k')%>%
  ungroup()
~~~

~~~
ggplot(data=Covid_flights.melted,aes(x=date,y=value,group=variable,color=variable))+
  geom_line()+
  scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")+
  labs(title="Coivd 19 Cases Vs. Total Flights Trend in 2019-2020")

```
~~~

Explanatory analysis Geographic Code

```
library(zoo)
library(shiny)
library(shinythemes)
library(dbplyr)
library(ggplot2)
library(lubridate)
library(tidyverse)
library(usmap)
library(stringr)
library(reshape2)
library(scales)

# Function for date slider by month
rm(list=ls())
monthStart <- function(x) {
  x <- as.POSIXlt(x)
  x$mday <- 1
  as.Date(x)
}

#UI Design
ui <- fluidPage(
              tags$style('.container-fluid {background-color: #cce9f0;}'),
              titlePanel("USA Flights Viewer"),
              theme=shinytheme("cerulean"),
              fluidRow(
                column(12,splitLayout(cellWidths = c("30%",
"45%","25%"),plotOutput("lineplot"),plotOutput("plot2"),plotOutput("pieplot")))
                ),
              br(),
              tags$hr(style="border-color: black;"),
              br(),
```

```
       fluidRow(
       column(3, sliderInput("slider", "Date Range",
              min = as.Date("2019-01-31"),
              max =as.Date("2020-12-31"),
              value=c(as.Date("2019-01-31")),
              timeFormat="%b %Y"),
              textOutput("SliderText")
          ),
       column(3,selectInput("States", "Select State:",
             append(c(unique(Summary_stats$State_Abrv)),"ALL"),
             selected='ALL')
          ),

       column(3,selectInput("AirLine", "Select Airline",
             append(c(unique(Summary_stats$carrier_name)),"ALL"),
             selected='ALL')
          ),
       column(3,selectInput("FlightType", "Type",
             c('All Flights','Delayed Flights','Cancelled flights','Diverted Flights'),
             selected='ALL'))
           ))

server <- shinyServer(function(input, output, session){
 Summary_stats<-read.csv("combined_df.csv")
 states_abrv=c(unique(Summary_stats$State_Abrv))
 append(states_abrv, "ALL")
 statesmap <-map_data("state")
 statesmap<-statesmap%>%
  mutate(region=str_to_title(region))%>%
  mutate(S_abbrv=state.abb[match(region,state.name)])

 airport_df<-Summary_stats%>%
  select(c("airport","State_Abrv"))%>%
  distinct()

 Summary_stats<-read.csv("combined_df.csv")
 states_abrv=c(unique(Summary_stats$State_Abrv))
 append(states_abrv, "ALL")
 statesmap <-map_data("state")
 statesmap<-statesmap%>%
  mutate(region=str_to_title(region))%>%
  mutate(S_abbrv=state.abb[match(region,state.name)])

 airport_df<-Summary_stats%>%
  select(c("airport","State_Abrv"))%>%
  distinct()
sliderMonth <- reactiveValues()
observe({
  full.date <- as.POSIXct(input$slider, tz="GMT")
  print(full.date)
  sliderMonth$Month <- as.character(monthStart(full.date))
})
output$SliderText <- renderText({sliderMonth$Month})

Summary_stats<-Summary_stats%>%
  mutate(on_time_flights=arr_flights-arr_del15-arr_cancelled-arr_diverted)

output$pieplot<-renderPlot({

  #filter summary_stats by month
  yeart=as.numeric(substr(sliderMonth$Month, 1, 4))
  montht=as.numeric(substr(sliderMonth$Month, 6, 7))
  Summary_stats=filter(Summary_stats,year==yeart & month==montht)
  date=as.Date(sliderMonth$Month)
  date=as.yearmon(date, format = "%b/%Y")
  title='Flight distribution: \n'
  title=paste(date,title,sep=' ')
  #filter summary_stats by air_carrier
  if ( input$AirLine == 'ALL') {
    Summary_stats=Summary_stats
  }
  else  {
    Summary_stats=filter(Summary_stats,carrier_name==input$AirLine)
    title=paste(title,input$AirLine,sep=' ')
  }

  #filter summary_stats by state
  if ( input$States == 'ALL') {
    Summary_stats=Summary_stats
  }
  else  {
    Summary_stats=filter(Summary_stats,State_Abrv == input$States)
    title=paste(title,input$States,sep=' ')
  }


  if (input$FlightType!='Delayed Flights'){
   agg_df<-Summary_stats%>%
    na.omit()%>%
    filter(V16>-125)%>%
    group_by(year,month)%>%

summarise(Mean_flights=sum(on_time_flights),mean_cancel=sum(arr_cancelled),mean_delay=
sum(arr_del15),
         mean_diverted=sum(arr_diverted))%>%
    #select(-c(year,month))%>%
    melt(id.var=c("year","month"),var.name='value')%>%
    mutate(value=value/sum(value))%>%
    mutate(labels = scales::percent(value))

   validate(need(nrow(agg_df) > 0, 'No data exists, please select a diffrent Category'))
   ggplot(agg_df, aes(x="", y=value, fill=variable)) +
    geom_bar(stat="identity", width=1, color="white") +
    coord_polar("y", start=0) +
    geom_text(aes(label = labels),
         position = position_stack(vjust = 0.8))+
    scale_fill_manual(values=c("#66C2A5", "#FC8D62", "#8DA0CB","#E78AC3"),
             name="Flight Type Distribution",
             labels=c("On Time Flights", "Canceled Flights", "Delayed Flights","Diverted
Flights"))+
    labs(x='distribution',
       y='',
       title=title)+
    theme(legend.position = c(0.8, 0.2),
       axis.text = element_blank(),
       axis.ticks = element_blank(),
       panel.grid  = element_blank())
  }
  else{
   title=paste("Delayed flight reasons in",title,sep=' ')
   agg_df<-Summary_stats%>%
    na.omit()%>%
    filter(V16>-125)%>%
    group_by(year,month)%>%
    summarise(carrier_ct=sum(carrier_ct),weather_ct=sum(weather_ct),nas_ct=sum(nas_ct),
         secuirty_ct=sum(security_ct),late_aircraft_ct=sum(late_aircraft_ct))%>%
    #select(-c(year,month))%>%
    melt(id.var=c("year","month"),var.name='value')%>%
    mutate(value=value/sum(value))%>%
    mutate(labels = scales::percent(value))
  ggplot(agg_df, aes(x="", y=value, fill=variable)) +
   geom_bar(stat="identity", width=1, color="white") +
   coord_polar("y", start=0)+
   geom_text(aes(label = labels),
        position = position_stack(vjust = 0.8))+
   scale_fill_manual(values=c("#8dd3c7", "#ffffb3", "#bebada","#fb8072","#80b1d3"),
            name="Flight Delay Reason",
            labels=c("Carrier", "Weather", "FAA System outage","Secuirty","Late Aircraft"))+
   labs(x='distribution',
     y='',
     title=title)+
   theme(legend.position = c(0.8, 0.2),
      axis.text = element_blank(),
      axis.ticks = element_blank(),
      panel.grid  = element_blank())
 }
})

#Draw Line graph
output$lineplot<-renderPlot({
 title=paste(input$FlightType,' 2019 & 2020')
 #filter summary_stats by air_carrier
 if ( input$AirLine == 'ALL') {
  Summary_stats=Summary_stats
 }
 else  {
  Summary_stats=filter(Summary_stats,carrier_name==input$AirLine)
  title=paste(title,input$AirLine,sep=' ')
 }

 #filter summary_stats by state
 if ( input$States == 'ALL') {
  Summary_stats=Summary_stats
 }
 else  {
  Summary_stats=filter(Summary_stats,State_Abrv == input$States)
  title=paste(title,input$States,sep=' ')
 }

 agg_df<-Summary_stats%>%
  na.omit()%>%
  filter(V16>-125)%>%
  group_by(year,month)%>%
summarise(Mean_flights=sum(arr_flights),mean_cancel=sum(arr_cancelled),mean_delay=sum(
arr_del15),
                diverted=sum(arr_diverted),long=mean(V16),lat=mean(V15))

    #filter agg by flight type
    if ( input$FlightType == 'All Flights') {
      ftype=agg_df$Mean_flights
    } else if ( input$FlightType == 'Delayed Flights') {
      ftype=agg_df$mean_delay
    } else if ( input$FlightType == 'Cancelled flights') {
      ftype=agg_df$mean_cancel
    } else {
      ftype=agg_df$diverted
    }



    agg_df$Date <- (with(agg_df, sprintf("%d-%02d", year, month)))
    agg_df$Date <- paste0(agg_df$Date,"-01")
    agg_df$Date <- as.Date(agg_df$Date,"%Y-%m-%d")

    ggplot(data=agg_df,aes(x=Date,y=ftype))+
      geom_line()+
      scale_x_date(date_breaks = "2 month", date_labels = "%m/%y")+
      geom_vline(xintercept =as.Date(sliderMonth$Month) , linetype="dashed",color = "red",
size=1.5)+
      labs(x='Date',
           y=input$FlightType,
           title=title)

  }, height = 200, width = 450)

  #draws USA Map
  output$plot2<-renderPlot({

    title=input$FlightType
    #filter summary_stats by month
    yeart=as.numeric(substr(sliderMonth$Month, 1, 4))
    montht=as.numeric(substr(sliderMonth$Month, 6, 7))
    Summary_stats=filter(Summary_stats,year==yeart & month==montht)
    date=as.Date(sliderMonth$Month)
    date=as.yearmon(date, format = "%b/%Y")
    title=paste(title,date,sep=' ')
    #filter summary_stats by air_carrier
    if ( input$AirLine == 'ALL') {
      Summary_stats=Summary_stats
    }
    else  {
      Summary_stats=filter(Summary_stats,carrier_name==input$AirLine)
      title=paste(title,'for',sep=' ')
      title=paste(title,input$AirLine,sep=' ')
    }


    #aggregate the summary_stats
    agg_df<-Summary_stats%>%
      na.omit()%>%
      filter(V16>-125)%>%
      group_by(airport)%>%

summarise(Mean_flights=sum(arr_flights),mean_cancel=sum(arr_cancelled),mean_delay=sum(
arr_del15),mean_diverted=sum(arr_diverted),
                long=mean(V16),lat=mean(V15))

    agg_df = merge(x=agg_df,y=airport_df,by="airport",all.x=TRUE)


    #filter agg by state
    if ( input$States == 'ALL') {
      statesmap=statesmap
    }
    else  {

      statesmap=filter(statesmap,S_abbrv %in% c(as.character(input$States)))
      agg_df=filter(agg_df,State_Abrv == input$States)
      title=paste(title,'in',sep=' ')
      title=paste(title,input$States,sep=' ')
    }

    #filter agg by flight type
    if ( input$FlightType == 'All Flights') {
      ftype=agg_df$Mean_flights
    } else if ( input$FlightType == 'Delayed Flights') {
      ftype=agg_df$mean_delay
    } else if ( input$FlightType == 'Cancelled flights') {
      ftype=agg_df$mean_cancel
    } else {
      ftype=agg_df$mean_diverted
    }
    if ( input$FlightType == 'All Flights') {
      color_type<- scale_color_gradient(low = '#f7fcb9', high = '#31a354')

    }
    else  {
      color_type<- scale_color_gradient(low = "#fb6a4a", high = "#a50f15")
    }

    ggplot()+
      geom_polygon(data=statesmap,
                   aes(x=long, y=lat, group=group),
                   colour='black',
                   fill=NA)+
      geom_point(data=agg_df,
                 aes(long,lat,size=ftype,color=ftype))+
      color_type+
       labs(x='',
           y='',
           title=title,
           color='Number of Flights')+
      theme(axis.ticks.x = element_blank(),
            axis.text.x = element_blank(),
            axis.ticks.y = element_blank(),
            axis.text.y = element_blank(),
            panel.grid  = element_blank(),
            panel.background = element_rect(fill = "lightblue"))+
      guides(size = FALSE)+
      theme(legend.position = c(0.90, 0.2))
    }, height = 400, width = 675)

  })

shinyApp(ui = ui, server = server)
```

Boxplot and Heatmap Code

Boxplot and Heatmap link :
https://github.com/yuyuntzu/DSC-465-PROJECT/blob/8399457acd8c299280793baf6791857c33b4cbad/Airline%20delay%20time.R

```
#Distribution of the delay flight and the the delay reason



    b1 <-Airline_Delay_Cause._2019 %>%
     mutate(del_pct = arr_del15/arr_flights) %>%
     ggplot(aes(x =carrier_name, y=arr_del15)) +
     geom_boxplot(aes(colour = carrier_name), na.rm = TRUE) + coord_flip() +
     theme(legend.position='none') + xlab("Arline Carrier") +ylab("Total number of delayed
flights in a day(2019)")



    b2 <-Airline_Delay_Cause_2020 %>%
     mutate(del_pct = arr_del15/arr_flights) %>%
     ggplot(aes(x =carrier_name, y=arr_del15)) +
     geom_boxplot(aes(colour = carrier_name), na.rm = TRUE) + coord_flip() +
     theme(legend.position='none') + xlab("Arline Carrier") +ylab("Total number of delayed
flights in a day(2020)")


    grid.arrange(b1,b2,ncol=2)



    #Correlation between the delay flight and the airport


    p1<-Airline_Delay_Cause._2019 %>% select(carrier_name, airport, arr_del15,
arr_flights) %>% drop_na() %>%
     filter(airport == as.character("RDU") |airport == as.character("JAX")|airport ==
as.character("BNA")|airport == as.character("CLE")|airport == as.character("IND"))%>%
     group_by(airport, carrier_name) %>% dplyr::summarize_all(funs(sum)) %>%
     mutate(del_pct = arr_del15/arr_flights) %>%
     ggplot(aes(x=factor(airport), y= factor(carrier_name), fill=del_pct),na.rm=TRUE) +
geom_tile() +
     theme(axis.text.x=element_text(angle=45)) +
     scale_fill_gradient(low = "white", high = "red")+labs(y = "Airline", x = "2019 Airport",fill
="Percentage of delayed aircraft in a day ")+theme(legend.position = "bottom")
```

```
        p2<-Airline_Delay_Cause_2020 %>% select(carrier_name, airport, arr_del15,
arr_flights) %>% drop_na() %>%
         filter(airport == as.character("RDU") |airport == as.character("JAX")|airport ==
as.character("BNA")|airport == as.character("CLE")|airport == as.character("IND"))%>%
         group_by(airport, carrier_name) %>% dplyr::summarize_all(funs(sum)) %>%
         mutate(del_pct = arr_del15/arr_flights) %>%
         ggplot(aes(x=factor(airport), y= factor(carrier_name), fill=del_pct),na.rm=TRUE) +
geom_tile() +
         theme(axis.text.x=element_text(angle=45)) +
         scale_fill_gradient(low = "white", high = "orange")+labs(y = "Airline", x = "2020
Airport",fill ="Percentage of delayed aircraft in a day  ") +theme(legend.position = "bottom")
        grid.arrange(p1,p2,ncol=2)
```