STAT 184 Final Project: Exploring Flight Status During the Week of Christmas

Felicia Vijayarangam, Varsha Giridharan

2024-12-18

5201 3:49 PM E54 On Time
910 11:19 AM B18 Delayed 1:20 PM
3196 12:30 PM A4 Cancelled
306 1:05 PM B17 On Time
3451 2:36 PM E53 On Time
418 3:00 PM B19 On Time
1871 4:25 PM A8 On Time
5166 2:54 PM D43B Delayed 3:24 PM
5269 2:57 PM E51 On Time
5147 1:59 PM E52 On Time
554 11:30 AM C24 Delayed 12:15 PM
5847 12:00 PM B20 NOW 12:13 PM
2254 12:30 PM C30 Cancelled
2454 1:30 PM C24 On Time
2142 2:25 PM C23 Cancelled
2554 2:30 PM C28 On Time
2149 3:00 PM C31 Cancelled
5742 3:25 PM B22 On Time
1254 3:30 PM C24 Cancelled
5138 3:39 PM E57 Cancelled
4406 2:53 PM D44B On Time
5337 1:45 PM D44B On Time
3393 3:32 PM E55 Cancelled
354 10:55 AM C31 Delayed 2:24 PM
620 1:56 PM C29 On Time
CHICAGO, ORD United 3546 1:00 PM B10C On Time
CHICAGO, ORD United 5859 2:50 PM B14 Cancelled
CHICAGO, ORD American 1107 2:55 PM D40 On Time
CINCINNATI American 5159 1:59 PM E47 Cancelled
CINCINNATI Delta 4905 2:00 PM B18 NOW 2:07 PM
CLEVELAND American 5197 3:27 PM E49 Cancelled
COLUMBUS American 1472 3:20 PM C29 Cancelled
COLUMBUS American 4519 4:20 PM E51 Cancelled
DALLAS, DAL Southwest 4887 1:50 PM A4 Cancelled
DALLAS, DFW D45 Delayed 1:55 PM
DALLAS, DFW On Time
DENVER Cancelled
DENVER On Time
DETROIT Delayed 1:50 PM
DETROIT Cancelled
FORT LAUDERDALE Delayed 3:17 PM
FORT MYERS On Time
FT WALTON BEACH Delayed 12:29 PM
GREENSBORO D35 On Time
HARTFORD E47 Delayed 12:49 PM
HARTFORD On Time
HOUSTON On Time
HOUSTON
KANSAS CITY American 4308
KANSAS CITY Southwest 1505
KNOXVILLE American 5276
LANSING American 5640
LOUISVILLE American 3460
MADISON Delta 5650
MANCHESTER American 5208
MEMPHIS American 4316
MEMPHIS American 5343
MIAMI American 2878
MIAMI American 1340
MILWAUKEE Southwest 2768
MINNEAPOLIS Delta 2093
MINNEAPOLIS American 5566
MINNEAPOLIS Delta 1343
MONTREAL Air Canada 8824
MYRTLE BEACH American 5652
NASHVILLE American 5126
NASHVILLE Southwest 2929
NEW ORLEANS American 5325
NEW YORK, JFK Delta 5765
NEW YORK, JFK Delta 5668
NEW YORK, JFK American 4668
NEW YORK, LGA American 2119
NEW YORK, LGA Delta 5660

Figure 1: Flight Departure Board

Introduction

The holiday season is one of the busiest times of the year because everyone is traveling to see family, going on vacations, and getting away from the everyday chaos of life. With this, one of the biggest hurdles is transportation. Millions of individuals are either driving or flying during this time, creating an increase in traffic and delays nationwide. In particular, how many of you have seen headlines on the news stating, “Hundreds of Flights are Delayed Due to Holiday Rush?” Whether it be for Memorial Day weekend, Thanksgiving, Christmas, or any other holiday, various factors play a part in delays such as a flight’s departure delay, arrival delay, weather delay, or carrier delay. There could also be overall flight diversions or cancellations that play a part in this.

In our report, we investigated flight delays during the Christmas Season. We have two specific datasets from the United States Department of Transportation compiled by the Bureau of Transformation Statistics. One dataset is from December 2020, and the other is from December 2023.

Both these datasets adhere to the FAIR principles. The datasets are findable and accessible to the general public since it is posted on an open-platform government website: Flight Information. The website provides a way to filter through what months, years, and columns a researcher wants to look into and descriptions for each, making the dataset reusable for various purposes. Then, after selecting the desired attributes, researchers can click the download button, saving to file as T_ONTIME_MARKETING.csv. This CSV format makes the file interoperable.

Both these datasets adhere to the CARE principles. By looking at the dataset, there are collective benefits since there are several research questions such as what flights/airlines have the most delays, etc that could be answered, helping travelers decide what airline to fly. Since this is a government dataset, the airlines have given consent to share flight information with individuals. Our data sets do not have sensitive data, allowing it to adhere to the responsible and ethical criteria.

Within the December 2020 dataset, there are 397,208 cases, each representing individual flights. In comparison, December 2023 had 606,218 cases of flights. Notice how in December 2023 the number of flights almost doubled. This is because, in 2020, there was COVID-19, so the number of individuals traveling greatly decreased. Throughout our analysis, we want to compare these years and look at 34 attributes (later described in detail) such as delay times.

These are our main research questions:

1. What is the average delay time for the week of Christmas 2020 for the top 10 most popular U.S. airlines?
2. What is the average delay time for the week of Christmas 2023 for the top 10 most popular U.S. airlines?
3. In 2023, what were the types of delays on average that the Top 10 airlines experienced the week of Christmas?
4. What are the Top 10 Most Popular destinations nationwide to travel to during Christmas at the Philadelphia Airport?

5. How did mean weather delays at the top 10 busiest airports during the Christmas travel week compare between 2020 and 2023?
6. How did the average delays (weather and carrier delays) differ between Pittsburgh (PIT) and Philadelphia (PHL) during the week of Christmas in 2023 compared to 2020?

Data Inventory

In order to start answering these research questions, we first need to load the December 2020 and December 2023 data set and the libraries needed for our data visualization. In this case, we will use libraries such dplyr, tidyr, kableExtra, stringr, ggplot2, scales, knitr, readr. Once we have these libraries, we can proceed to the data exploration step.

Data Exploration

Throughout our analysis, we want to compare both these years together and look at various attributes relationships. One of the challenges we faced was that looking at the entire month of dataset was hard to do because of how big the data files are. December 2020 contains 397802 flights and December 2023 contains 606218 flights. Each of these data sets have the following 34 attributes:

```
 [1] "YEAR"                "DAY_OF_MONTH"        "FL_DATE"
 [4] "OP_UNIQUE_CARRIER"   "OP_CARRIER"          "OP_CARRIER_FL_NUM"
 [7] "ORIGIN_AIRPORT_ID"   "ORIGIN"              "ORIGIN_CITY_NAME"
[10] "DEST_AIRPORT_ID"     "DEST"                "DEST_CITY_NAME"
[13] "DEP_DELAY_NEW"       "ARR_DELAY_NEW"       "CANCELLED"
[16] "DIVERTED"            "FLIGHTS"             "CARRIER_DELAY"
[19] "WEATHER_DELAY"       "NAS_DELAY"           "SECURITY_DELAY"
[22] "LATE_AIRCRAFT_DELAY" "DIV_AIRPORT_LANDINGS" "DIV_ARR_DELAY"
[25] "DIV1_AIRPORT"        "DIV1_AIRPORT_ID"     "DIV2_AIRPORT"
[28] "DIV2_AIRPORT_ID"     "DIV3_AIRPORT"        "DIV3_AIRPORT_ID"
[31] "DIV4_AIRPORT"        "DIV4_AIRPORT_ID"     "DIV5_AIRPORT"
[34] "DIV5_AIRPORT_ID"
```

Since this governmental dataset looks at all the flights traveling nationwide, we decided it was best to create a new dataset to analyze. Since we are concerned about the holiday rush, we decided to focus on the Christmas season and make a specific subset that looks at the week of Christmas.

In our subset, we want to use the filter() in order to filter the original datasets with the whole month of December to the specific week of Christmas. Based on the columns, we decided to filter based on the Day of the Month. Now, we will have two new datasets with only the flights for Christmas week.

```
#Filter December 2020:
christmas_week2020 <- flights2020 %>%
  filter(DAY_OF_MONTH >= 20 & DAY_OF_MONTH <= 27)

#Filter December 2023:
christmas_week2023 <- flights2023 %>%
  filter(DAY_OF_MONTH >= 20 & DAY_OF_MONTH <= 27)
```

We want to create summary() functions to understanding the data types and distributions of each of the attributes represented. In order to have the summary() properly formatted and have all 34 attributes fit into the pdf, we flipped the rows and the column values. With this, now the columns are the summary statistics and the rows are the data sets attributes. Therefore we used the tranpose function t(). To create a summary table, we used the kable(), kable_styling(), column_spec(), and row_spec() for styling. We also needed the package kableExtra().

In the first table, we see the flight summary table for the week of Christmas in 2020.

Table 1: Flight Summary Table of Christmas Week 2020

YEAR	Min. :2020	1st Qu.:2020	Median :2020	Mean :2020	3rd Qu.:2020	Max. :2020	NA
DAY_OF_MONTH	Min. :20.00	1st Qu.:21.00	Median :23.00	Mean :23.43	3rd Qu.:26.00	Max. :27.00	NA
FL_DATE	Length:11021	Class :character	Mode :character	NA	NA	NA	NA
OP_UNIQUE_CARRIER	Length:110216	Class :character	Mode :character	NA	NA	NA	NA
OP_CARRIER	Length:11021	Class :character	Mode :character	NA	NA	NA	NA
OP_CARRIER_FL_NUM	Min. : 1	1st Qu.:1108	Median :2366	Mean :2787	3rd Qu.:4575	Max. :8802	NA
ORIGIN_AIRPORT_ID	Min. :10135	1st Qu.:11292	Median :12884	Mean :12663	3rd Qu.:14098	Max. :16869	NA
ORIGIN	Length:110216	Class :character	Mode :character	NA	NA	NA	NA
ORIGIN_CITY_NAME	Length:11021	Class :character	Mode :character	NA	NA	NA	NA
DEST_AIRPORT_ID	Min. :10135	1st Qu.:11292	Median :12884	Mean :12662	3rd Qu.:14098	Max. :16869	NA
DEST	Length:11021	Class :character	Mode :character	NA	NA	NA	NA

DEST_CITY_NAME	Length:110216	Class :character	Mode :character	NA	NA	NA	NA
DEP_DELAY_NEW	Min. : 0.000	1st Qu.: 0.000	Median : 0.000	Mean : 9.032	3rd Qu.: 1.000	Max. :1779.000	NA's :1108
ARR_DELAY_NEW	Min. : 0.000	1st Qu.: 0.000	Median : 0.000	Mean : 9.311	3rd Qu.: 3.000	Max. :1776.000	NA's :1335
CANCELLED	Min. :0.00000	1st Qu.:0.00000	Median :0.00000	Mean :0.01047	3rd Qu.:0.00000	Max. :1.00000	NA
DIVERTED	Min. :0.000000	1st Qu.:0.000000	Median :0.000000	Mean :0.001642	3rd Qu.:0.000000	Max. :1.000000	NA
FLIGHTS	Min. :1	1st Qu.:1	Median :1	Mean :1	3rd Qu.:1	Max. :1	NA
CARRIER_DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 4.00	Mean : 27.16	3rd Qu.: 26.00	Max. :1776.00	NA's :95279
WEATHER_DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 5.53	3rd Qu.: 0.00	Max. :1593.00	NA's :95279
NAS_DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 1.00	Mean : 12.47	3rd Qu.: 18.00	Max. :1224.00	NA's :95279
SECURITY_DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 0.14	3rd Qu.: 0.00	Max. :127.00	NA's :95279
LATE_AIRCRAFT_DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 15.31	3rd Qu.: 14.00	Max. :1209.00	NA's :95279
DIV_AIRPORT_LANDIN	Min. :0.000000	1st Qu.:0.000000	Median :0.000000	Mean :0.003684	3rd Qu.:0.000000	Max. :9.000000	NA
DIV_ARR_DELAY	Min. : 44.0	1st Qu.: 125.0	Median : 174.0	Mean : 261.6	3rd Qu.: 253.0	Max. :1192.0	NA's :110063
DIV1_AIRPORT	Length:11021	Class :character	Mode :character	NA	NA	NA	NA
DIV1_AIRPORT_ID	Min. :10135	1st Qu.:11282	Median :12265	Mean :12642	3rd Qu.:14089	Max. :16101	NA's :110010
DIV2_AIRPORT	Length:11021	Class :character	Mode :character	NA	NA	NA	NA
DIV2_AIRPORT_ID	Min. :12889	1st Qu.:12889	Median :12889	Mean :12889	3rd Qu.:12889	Max. :12889	NA's :110215
DIV3_AIRPORT	Mode:logical	NA's:110216	NA	NA	NA	NA	NA
DIV3_AIRPORT_ID	Mode:logical	NA's:110216	NA	NA	NA	NA	NA
DIV4_AIRPORT	Mode:logical	NA's:110216	NA	NA	NA	NA	NA
DIV4_AIRPORT_ID	Mode:logical	NA's:110216	NA	NA	NA	NA	NA
DIV5_AIRPORT	Mode:logical	NA's:110216	NA	NA	NA	NA	NA
DIV5_AIRPORT_ID	Mode:logical	NA's:110216	NA	NA	NA	NA	NA

In the second table, we see the flight summary table for the week of Christmas in 2023.

Table 2: Flight Summary Table of Christmas Week 2020

YEAR	Min. :2023	1st Qu.:2023	Median :2023	Mean :2023	3rd Qu.:2023	Max. :2023	NA
DAY_OF_MONTH	Min. :20.00	1st Qu.:21.00	Median :23.00	Mean :23.45	3rd Qu.:26.00	Max. :27.00	NA
FL_DATE	Length:15721	Class :character	Mode :character	NA	NA	NA	NA
OP_UNIQUE_CARRIER	Length:157219	Class :character	Mode :character	NA	NA	NA	NA

OP__CARRIER	Length:15721	Class :character	Mode :character	NA	NA	NA	NA
OP__CARRIER__FL__NUM	Min. : 1	1st Qu.:1090	Median :2115	Mean :2423	3rd Qu.:3502	Max. :9658	NA
ORIGIN__AIRPORT__ID	Min. :10135	1st Qu.:11292	Median :12889	Mean :12674	3rd Qu.:14057	Max. :16869	NA
ORIGIN	Length:157219	Class :character	Mode :character	NA	NA	NA	NA
ORIGIN__CITY__NAME	Length:15721	Class :character	Mode :character	NA	NA	NA	NA
DEST__AIRPORT__ID	Min. :10135	1st Qu.:11292	Median :12889	Mean :12674	3rd Qu.:14057	Max. :16869	NA
DEST	Length:15721	Class :character	Mode :character	NA	NA	NA	NA
DEST__CITY__NAME	Length:157219	Class :character	Mode :character	NA	NA	NA	NA
DEP__DELAY__NEW	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 15.11	3rd Qu.: 10.00	Max. :3786.00	NA's :1157
ARR__DELAY__NEW	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 13.96	3rd Qu.: 8.00	Max. :3795.00	NA's :1660
CANCELLED	Min. :0.000000	1st Qu.:0.000000	Median :0.000000	Mean :0.007684	3rd Qu.:0.000000	Max. :1.000000	NA
DIVERTED	Min. :0.000000	1st Qu.:0.000000	Median :0.000000	Mean :0.002875	3rd Qu.:0.000000	Max. :1.000000	NA
FLIGHTS	Min. :1	1st Qu.:1	Median :1	Mean :1	3rd Qu.:1	Max. :1	NA
CARRIER__DELAY	Min. : 0.0	1st Qu.: 0.0	Median : 4.0	Mean : 22.6	3rd Qu.: 21.0	Max. :3786.0	NA's :127460
WEATHER__DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 5.36	3rd Qu.: 0.00	Max. :1496.00	NA's :127460
NAS__DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 11.06	3rd Qu.: 15.00	Max. :1260.00	NA's :127460
SECURITY__DELAY	Min. : 0.00	1st Qu.: 0.00	Median : 0.00	Mean : 0.19	3rd Qu.: 0.00	Max. :242.00	NA's :127460
LATE__AIRCRAFT__DELAY	Min. : 0.0	1st Qu.: 0.0	Median : 4.0	Mean : 28.4	3rd Qu.: 31.0	Max. :1606.0	NA's :127460
DIV__AIRPORT__LANDIN	Min. :0.000000	1st Qu.:0.000000	Median :0.000000	Mean :0.004929	3rd Qu.:0.000000	Max. :9.000000	NA
DIV__ARR__DELAY	Min. : 41.0	1st Qu.: 154.5	Median : 220.0	Mean : 366.5	3rd Qu.: 452.5	Max. :1388.0	NA's :156872
DIV1__AIRPORT	Length:15721	Class :character	Mode :character	NA	NA	NA	NA
DIV1__AIRPORT__ID	Min. :10140	1st Qu.:11292	Median :12953	Mean :12887	3rd Qu.:14492	Max. :15380	NA's :156732
DIV2__AIRPORT	Length:15721	Class :character	Mode :character	NA	NA	NA	NA
DIV2__AIRPORT__ID	Min. :10466	1st Qu.:11041	Median :11941	Mean :12284	3rd Qu.:13351	Max. :14869	NA's :157203
DIV3__AIRPORT	Mode:logical	NA's:157219	NA	NA	NA	NA	NA
DIV3__AIRPORT__ID	Mode:logical	NA's:157219	NA	NA	NA	NA	NA
DIV4__AIRPORT	Mode:logical	NA's:157219	NA	NA	NA	NA	NA
DIV4__AIRPORT__ID	Mode:logical	NA's:157219	NA	NA	NA	NA	NA
DIV5__AIRPORT	Mode:logical	NA's:157219	NA	NA	NA	NA	NA
DIV5__AIRPORT__ID	Mode:logical	NA's:157219	NA	NA	NA	NA	NA

Table Insights:

At a high level, this table helps us pinpoint what attributes are most meaningful for us to analyze and compare performance against. There are various categorical variables such as year, day of the month, OP unique carrier, OP carrier, OP carrier flight number, origin, city name, destination, and such. For numerical values, keep in mind what they represent. For example, ID is a numerical value, but it is categorical in this case. Each airport has a specific ID number, so the mean, median, and such do not mean anything. However, the numerical numbers such as departure delays, arrival delays, flights carrier delays, weather delays, NAS delays, security delays, late aircraft delays, and diverted airport delays do matter. In our research questions, we can use these numerical attributes as checkpoints to see how airlines performance is compared to each other.

With these thoughts in mind, let us begin answering our research questions.

Research Question 1: What is the Average Departure Delay time for the week of Christmas 2020 for the top 10 most popular U.S. airlines?

Before we answer this question. Let us understand what we are being asked to do. We want to find the average departure delay time for each day of Christmas week based on the top ten airlines. In the dataset that we have, the airlines are represented by the OP_UNIQUE_CARRIER column. However, our dataset does not provide a key to what letters and numbers represents a specific airline. Therefore, we found a new data source with two column, one of airline codes and the other with airline names.

With this information, we could use the left_join() to create a new dataset with a new column of airline names. The logic behind this was that we needed to left_join() by the condition that the new data sources code matched the orginal's **unique carrier code**.

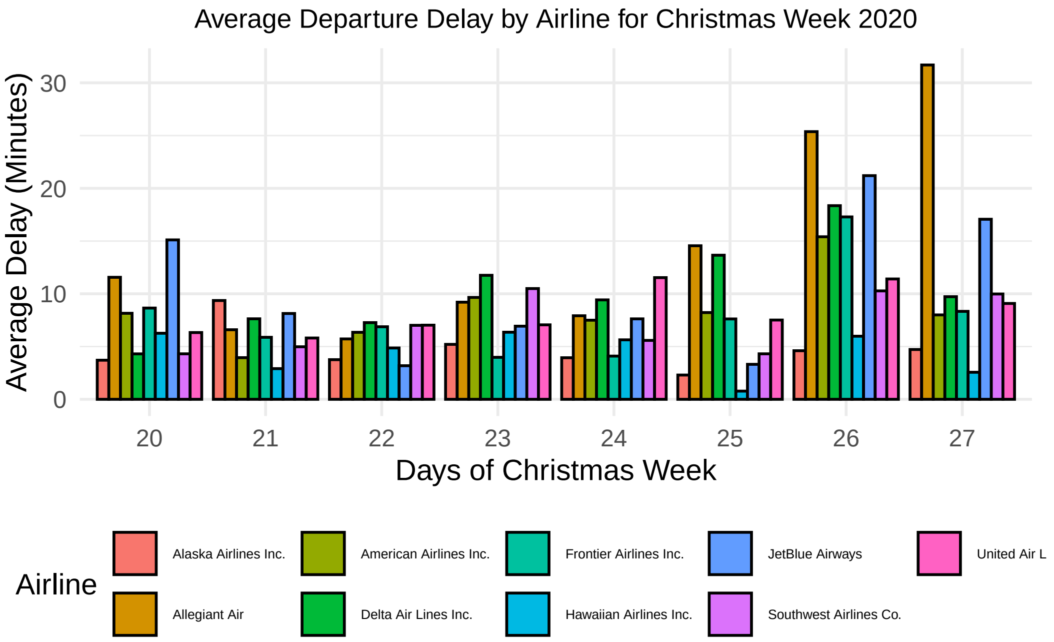
```r
# Merge the datasets based on the airline code
airlineInfo2020 <- christmas_week2020 %>%
  left_join(airlineCode, by = c("OP_UNIQUE_CARRIER" = "Code"))
```

Now that we have the specific airline names, we need to calculate the average airline delay based on the top 10 airlines. We need to group_by() the airline name and days of the week because we want to differentiate what departure delays the various airlines are experiencing the week of Christmas.

The average delay can be found by applying the mean() to departure delay. In order to ensure that our graph is clear to read, we decided to only plot a few airlines. In this case, we choose the top ten airlines according to the based on the https://airadvisor.com/en/top-us-airlines-rating website.

```r
# Calculate the average delay time by airline
avgDelay_airlineDay2020 <- airlineInfo2020 %>%
  group_by(Description, DAY_OF_MONTH) %>%
  summarize(AverageDelay = mean(DEP_DELAY_NEW, na.rm = TRUE), .groups = "drop")

specific_airlines <- c("Delta Air Lines Inc.", "Alaska Airlines Inc.", "Hawaiian Airlines Inc
                       "United Air Lines Inc.", "American Airlines Inc.", "JetBlue Airways",
                       "Southwest Airlines Co.", "Spirit Airlines", "Allegiant Air",
                       "Frontier Airlines Inc.")
```

Then, we created a bar graph using ggplot that filtered the graph based on the airline name and graphed 10 various colored bars for each day represents a different airline.

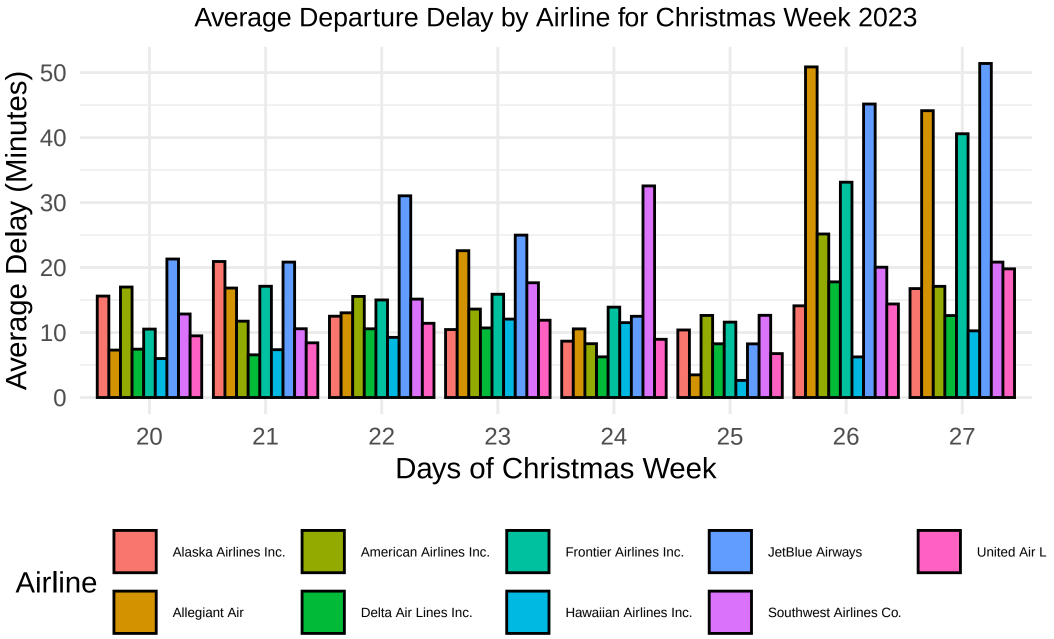


Graphical Insight: Looking at our x and y axis, we can see that Alaska Airlines Inc. had consistently low average departure delays that were less that 5 minutes. the Hawaiian Airlines Inc were also on the low end when it came to average departure delays since they were less than 10 minutes. The the other hand, Allegiant Air had the most delays by far, leading to an average more than 30 minutes on December 27. JetBlue Airways flight delays were also high compared to other flights, especially since they had an average departure delay of more than 20 minutes on December 27. Another aspect to consider is that the the average delays across airlines were the lowest on the 22. Also, the average delay rates spiked Christmas day and the days after.

Research Question 2: What is the Average Departure Delay time for the week of Christmas 2023 for the Top 10 most popular U.S. airlines?

To answer this research question, we follow a similar format as the previous question. First, we need to merge the 2023 Christmas week using the left_join() to get the airline name that corresponded to the existing columns airline code.

The next step is to use the group_by() and summarize() to find the average delay each day based on the airlines in 2023.



Graphical Insight:

In the graph above, the Hawaiian Airlines had the lowest average delays by far. Their average was consistently below 10 minutes throughout the entire week. Next, Alaska Airlines Inc. had consistently low average departure delays that are normally below 10-20 minutes. The airlines with high average delays seemed to occur after Christmas was Allegiant Air with more than 50 minutes delays on Christmas. JetBlue Airways flight was also up there with delays that were consitently higher than other airlines throughout the week and delays that were on average 40-50 minutes the days after Christmas. Delta's average delays were less throughout the week, and hit around 20 minutes the day after Christmas. Another aspect to consider is that on Christmas day the average delays across all airlines were below 20 minutes. However, average delay rates spiked immediately after Christmas.

Reflection on Research Question 1 and 2:

In 2020, the majority of airlines experienced average delay around 4-10 minutes leading up to Christmas day. One of the exceptions to this is the JetBlue Airways who had an average delay time around 15 minutes on December 20th. Even with these exceptions, the average delay time is significant less that the average delay time in 2023. The majority of airlines experienced an average delay time around between 8-15 minutes. JetBlue Airways repeatedly surpassed this, having a average departure delay around above 20 minutes for the weeks leading up to Christmas. From a broad lens, the rush after Christmas is high for both years. However, the top airlines who had the most avergae departure delays were consistently JetBlue Airways and Allegiant Air. In 2020, Allegiant Air had an average departure delay of 25 minutes on the 26th and 32 minutes on the 27th. In 2023, Allegiant Air had an average departure delay of 50 minutes on the 26th and 50 minutes on the 27th. Continuing onto JetBlue Airways, the average departure delay was around 21 minutes on the 26th and 17th minutes on the 27th. However, in 2023, they experienced an average departure delay time of 45 minutes on 26th and 51 minutes on the 27th.

We must also keep in mind that the number of people flying in 2020 was severely reduced due to COVID-19 restrictions and uncertainty. So, could conclude that in 2023, along with the overall U.S. welfare and economy, there was a higher average delay times for the majority of the top ten airlines compared to 2020.

Research Question 3: In 2023, what were the types of delays on average that the Top 10 airlines experienced the week of Christmas?

In our dataset, we are given various types of delays such as departure, arrival, carrier, weather, NAS, security, late aircraft, and diverted. We were curious to see what the average delay time in minutes would be for each airline. In order to do this, we want to create a dataset using the filter(), group(), and summarize(). The filter() filters by the top ten airlines that we previously discussed. The group_by() can be used to group our main dataset based on the airline names. Then, with the summarize(), we can use the mean() to get the average delay values.

```r
# Calculate average delays for Allegiant Air
avgDelayAlleg2023 <- airlineInfo2023 %>%
  filter(Description %in% specific_airlines2023) %>%
  group_by(Description) %>%
  summarize(
    Departure = mean(DEP_DELAY_NEW, na.rm = TRUE),
    Arrival = mean(ARR_DELAY_NEW, na.rm = TRUE),
    Carrier = mean(CARRIER_DELAY, na.rm = TRUE),
    Weather = mean(WEATHER_DELAY, na.rm = TRUE),
    NAS = mean(NAS_DELAY, na.rm = TRUE),
    Security = mean(SECURITY_DELAY, na.rm = TRUE),
    LateAircraft = mean(LATE_AIRCRAFT_DELAY, na.rm = TRUE),
    Diverted = mean(DIV_ARR_DELAY, na.rm = TRUE),
    .groups = "drop"
  )
```

Once we create this subset, we can use the kable() and kable_styling() to adjust the table format, font size, and etc. Once we do this, we will get the following table:

Table 3: Average Delay (in minutes) Based on Types of Delays for the Top 10 Airlines Christmas Weeks 2023

Description	Departure	Arrival	Carrier	Weather	NAS	Security	LateAircraft	Diverted
Alaska Airlines Inc.	13.741430	12.726664	17.92800	1.4906667	11.1884444	0.4168889	21.00800	208.5789
Allegiant Air	23.910235	23.532888	25.06926	14.3997114	16.9206349	0.0952381	41.54113	499.6000
American Airlines Inc.	15.492561	15.486012	25.23402	1.4374534	9.2835116	0.2767968	33.51206	249.5455
Delta Air Lines Inc.	10.152833	8.699815	40.05097	4.3948100	10.1510658	0.0532901	19.69184	504.0000
Frontier Airlines Inc.	19.518681	19.003910	21.05888	1.1959391	15.6690355	0.0000000	37.03249	423.0000
Hawaiian Airlines Inc.	8.173349	8.933333	29.43985	0.6428571	0.7518797	0.0000000	12.55263	200.0000
JetBlue Airways	27.382237	25.120263	33.46903	1.6159910	11.1891892	0.0833333	32.29054	284.2857
Southwest Airlines Co.	17.649189	15.119734	15.06974	0.7081888	7.8356465	0.1767291	29.28799	251.4576
United Air Lines Inc.	11.446092	10.855868	19.06123	0.7793998	15.0750203	0.0000000	27.10178	411.8500

Table Insights:

Looking at the Average Delay Table, we see that for the average departure delay, Hawaiian Airlines has the lowest around 8.17 minutes. On the other hand, JetBlue Airways has the highest around 27.38 minutes. For arrival delays, Delta has a average delay around 8.70 minutes. This is slightly less than Hawaiian Airline Inc., which was an average arrival delay around 8.93 minutes. JetBlue airlines is consistently higher average arrival delay with 25.12 minutes.

Now, looking at other factors like Carrier delay, on average Delta has the highest with 40.05 minutes, while Southwest Airline Co. has the lowest carrier delay of 15.07 minutes. The average weather delays for Allegiant Air was the highest with 14.40 minutes, and Hawaiian Airlines had the lowest average delay with 0.64 minutes. Looking at the National Aviation System delay, Hawaiian Airlines has the lowest with 0.75 average minutes, and Allegiant Air has the highest with 16.92 minutes. Looking at a security delay, Frontier, Hawaiian, and United Air Lines had no security delays, while Alaska Airlines had the highest 0.42 minutes. Hawaiian airlines had the lowest average late aircraft with 12.55 minutes, while Alliegiant Air had the highest with average 41.54 minutes delays.

Finally, looking at the diverted flight delays, Hawaiian Airlines had the lowest average delay around 200 minutes, while Delta had the highest around 504 average minutes delayed.

Based on this information, we can conclude that the performance of the Hawaiian Airlines were by far the best in the week of Christmas 2023.

Research Question 4: What are the Top 10 Most Popular destinations nationwide to travel to during Christmas at the Philadelphia Airport?

In order to look at the Top 10 most popular air travel destinations, we want to create a map of the U.S. and plot the Philadelphia Airport as the origin and have arrows going to various other airports.

In order to do this, we need to use the ggmap package. We also need to have the latitude and longitude coordinates of the various airports. Since our dataset did not have this information, we got another dataset from the U.S. Airport website. In this website, the dataset has the longitude and latitudes of the the airports, and has a specific column that breaks down what each airports IATA is. Since we have IATA code columns in our own dataset (ORIGIN and DEST), we can use the select() to get the three columns latitude_deg, iata_code, and longitude_deg. Then, we use the left_join() by the IATA codes for both the orgin and destination. This also meant that we renamed these column based on what coordinates they match to.

```r
library(ggmap)

# Load airport data and select IATA codes, latitude, and longitude
airports <- read.csv("C:/Users/felic/Downloads/us-airports.csv")

airports_coords <- airports %>%
  select(latitude_deg, iata_code,longitude_deg)

# Merge the selected columns with the ORIGIN column based on the IATA codes
flight_with_coords <- airlineInfo2023 %>%
  left_join(airports_coords, by = c("ORIGIN" = "iata_code")) %>%
  #Now we can store the orgin's latitude and longitude values
  rename(
    originLat = latitude_deg,
    originLong = longitude_deg,
  )
# Merge the selected columns with the DEST column based on the IATA codes
flight_with_coords <- flight_with_coords %>%
  left_join(airports_coords, by = c("DEST" = "iata_code")) %>%
  rename(
    destLat = latitude_deg,
    destLong = longitude_deg
  )
```

The next step is to take this newly merged dataset use the filter() to have the origin equal to

Philadelphia and use the groupby() to group by the destination. Also, to in order insure that the latitude and longitude values are numberical, we can read them using the as.numberic().

Now, in order to find the most popular destinations, we are going to use the count() to count how many destinations flights from Phildaephia were traveling to. Then, we can use the arrange() and desc(n) so the counts will be from most destination to least.

Then, using the head(10) and filter(), we stored the top 10 destinations.

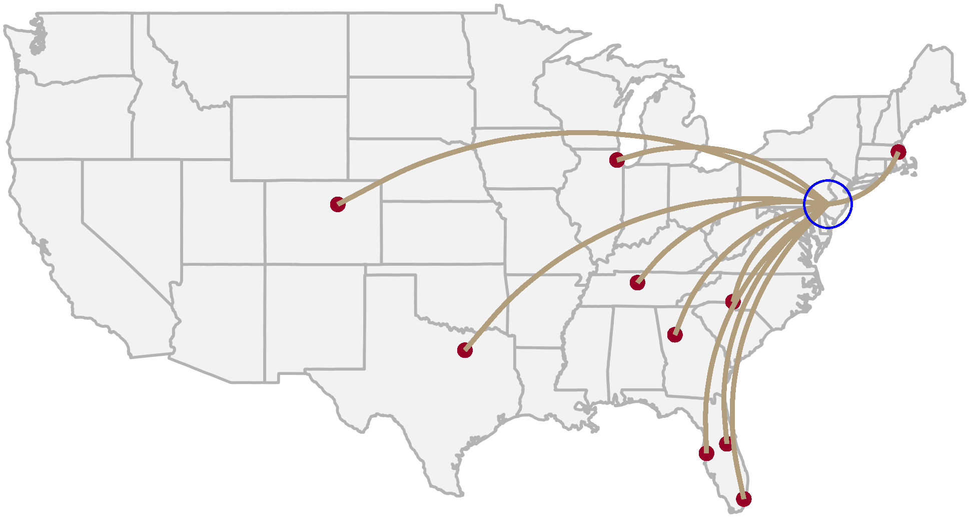
One of the most challenging parts of this analysis was the data visualization. In order to do this, we used another package called ggrepel. Next, since we needed to plot a U.S. map, we had to use this specific code that created the template for the plot:

```r
# Define the U.S. map
library(ggrepel)
us_map <- borders("state", colour = "gray70", fill = "gray95")
```

With this template, we used the ggplot(), geom_point(), geom_text_repel(), geom_curve(), theme(), theme_void(), and other functions to plot specific points of the U.S. map that correlates to the latitude and longitude destinations. Once we did this, we got the following plot:

Top 10 Destinations from Philadelphia (PHL) Based on Number of Flights

Flight Routes with Arrows to Popular Destinations



Source: Airline Data

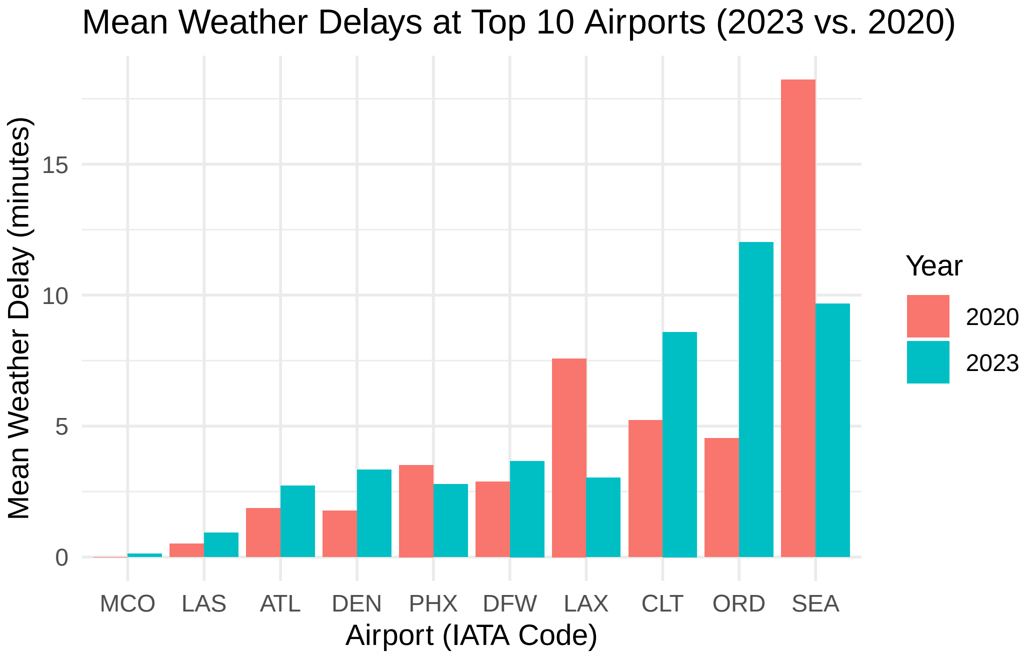
Looking at the code, we can see that the top ten number of flights go to Texas, Georgia, Florida, Illinois, Tennessee, Massachusetts, Colorado, and North Carolina. Then, we can further see that since three of the airports in Florida have high number of flights, showing that Florida is one of the popular destination spots.

Research Question 5: How did mean weather delays at the top 10 busiest airports during the Christmas travel week compare between 2020 and 2023?

For the following graph, I first identified the top 10 busiest airports during the Christmas week of 2023 by grouping the data by airport (ORIGIN), calculating the number of flights, and determining the average delays. I then filtered the dataset to include only these airports and merged the 2023 data with the corresponding year information. To compare weather delays between 2020 and 2023, I used bind_rows() to combine the two datasets and filtered for the top 10 airports from 2023. I grouped the data by airport and year, calculated the mean weather delay, and visualized the results in a bar chart where the x-axis shows the mean weather delays, the y-axis represents airports, and the bars are split by year for comparison.

```
# 1. Find most popular airports in the week of Christmas for 2023

# Create a dataframe that counts the number of flights and calculates average delays by ORIGI
flights_by_airport_2023 <- christmas_week2023 %>%
  group_by(ORIGIN) %>%
  summarise(
    Number_of_Flights = n(),
    Avg_Weather_Delay = median(WEATHER_DELAY, na.rm = TRUE),
    Avg_Carrier_Delay = median(CARRIER_DELAY, na.rm = TRUE)
  ) %>%
  arrange(desc(Number_of_Flights))

# Filter for the top 10 busiest airports
top10_flights_by_airport_2023 <- flights_by_airport_2023 %>%
  slice_max(order_by = Number_of_Flights, n = 10) %>%
  left_join(airlineInfo2023 %>% select(ORIGIN),
            by = "ORIGIN") %>%
  distinct()

# 2. Find most popular airlines in the week of Christmas for 2023

# Create a dataframe that counts the number of flights and calculates average delays by Airli
flights_by_airline_2023 <- christmas_week2023 %>%
  group_by(OP_UNIQUE_CARRIER) %>%
  summarise(
    Number_of_Flights = n(),
```

```r
    Avg_Weather_Delay = median(WEATHER_DELAY, na.rm = TRUE),
    Avg_Carrier_Delay = median(CARRIER_DELAY, na.rm = TRUE)
  ) %>%
  arrange(desc(Number_of_Flights))

# Filter for the top 10 airlines by number of flights
top10_flights_by_airline_2023 <- flights_by_airline_2023 %>%
  slice_max(order_by = Number_of_Flights, n = 10) %>%
  left_join(airlineInfo2023 %>% select(OP_UNIQUE_CARRIER),
            by = "OP_UNIQUE_CARRIER") %>%
  distinct()

# 3. Compare mean weather delays at top 10 airports for 2023 vs. 2020

bind_rows(
  christmas_week2023 %>% mutate(Year = 2023),
  christmas_week2020 %>% mutate(Year = 2020)
) %>%
  filter(ORIGIN %in% top10_flights_by_airport_2023$ORIGIN) %>%
  group_by(ORIGIN, Year) %>%
  summarise(Mean_Weather_Delay = mean(WEATHER_DELAY, na.rm = TRUE), .groups = "drop") %>%
  ggplot(aes(x = Mean_Weather_Delay, y = reorder(ORIGIN, Mean_Weather_Delay), fill = as.facto
  geom_bar(stat = "identity", position = "dodge") +
  coord_flip() +
  labs(
    title = "Mean Weather Delays at Top 10 Airports (2023 vs. 2020)",
    x = "Mean Weather Delay (minutes)",
    y = "Airport (IATA Code)",
    fill = "Year"
  ) +
  theme_minimal()
```

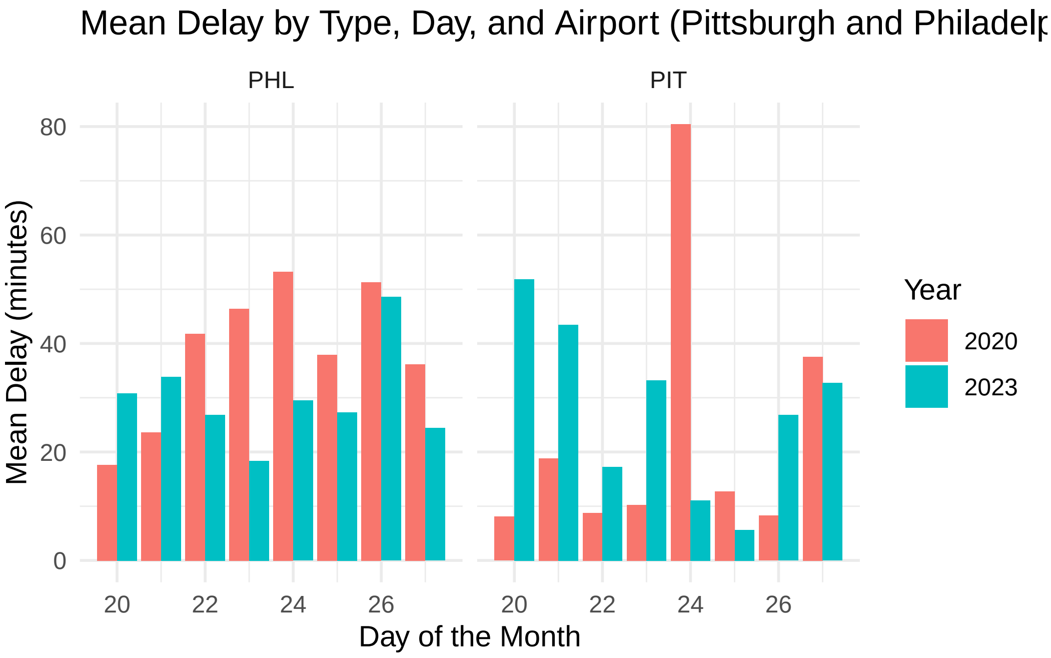


This bar chart compares the mean weather-related delays at the top 10 busiest airports during the Christmas week (20th-27th December) for the years 2020 and 2023. Notable differences are observed at airports like Los Angeles (LAX), San Francisco (SFO), and Seattle (SEA), where weather delays in 2020 were significantly higher. All of these airports are located on the West Coast, suggesting that weather-related delays in 2020 may have been influenced by common regional weather factors such as winter storms or heavy rain. This trend highlights airports where weather conditions during the holiday season posed considerable operational challenges, with 2020 showing a notably more disruptive impact on flight schedules at these locations compared to 2023.

Research Question 6: How did the average delays (weather and carrier delays) differ between Pittsburgh (PIT) and Philadelphia (PHL) during the week of Christmas in 2023 compared to 2020?

For this graph, I followed a similar approach to Research Question 5, but focused on comparing delays between Pittsburgh (PIT) and Philadelphia (PHL) during Christmas week in 2020 and 2023. After combining the two datasets using bind_rows(), I filtered the data for these two airports and the dates between December 20th and 27th. I then used pivot_longer() to reshape the delay data into a longer format, splitting it by delay type (carrier and weather). The data was grouped by day, airport, delay type, and year, and the mean delay was calculated. The results were visualized in a bar chart, with separate bars for each day and airport, allowing for a direct comparison of delay types and year-on-year differences between Pittsburgh and Philadelphia.

```
bind_rows(
  christmas_week2023 %>% mutate(Year = 2023),
  christmas_week2020 %>% mutate(Year = 2020)
) %>%
  filter(ORIGIN %in% c("PIT", "PHL"), DAY_OF_MONTH >= 20 & DAY_OF_MONTH <= 27) %>%
  pivot_longer(cols = c(CARRIER_DELAY, WEATHER_DELAY), names_to = "Delay_Type", values_to = "
  group_by(DAY_OF_MONTH, ORIGIN, Delay_Type, Year) %>%
  summarise(Mean_Delay = mean(Delay_Value, na.rm = TRUE), .groups = "drop") %>%
  ggplot(aes(x = DAY_OF_MONTH, y = Mean_Delay, fill = as.factor(Year))) +
  geom_bar(stat = "identity", position = "dodge") +
  facet_wrap(~ ORIGIN, ncol = 2) +
  theme_minimal() +
  labs(
    title = "Mean Delay by Type, Day, and Airport (Pittsburgh and Philadelphia, 2023 vs. 202
    x = "Day of the Month",
    y = "Mean Delay (minutes)",
    fill = "Year"
  )
```



This bar chart compares the mean weather-related delays at the top 10 busiest airports during the Christmas week (20th-27th December) for the years 2020 and 2023. Notable differences are observed at airports like Los Angeles (LAX), San Francisco (SFO), and Seattle (SEA), where weather delays in 2020 were significantly higher. All of these airports are located on the West Coast, suggesting that weather-related delays in 2020 may have been influenced by common regional weather factors such as winter storms or heavy rain. This trend highlights airports where weather conditions during the holiday season posed considerable operational challenges, with 2020 showing a notably more disruptive impact on flight schedules at these locations compared to 2023.

Conclusions

Overall in our analysis, we were able to answer the following questions:

In the first few research questions we focused on looking and comparing airline performance with each other and based on various years. We also looked at the various locations that people are traveling.

1. How did mean weather delays at the top 10 busiest airports during the Christmas travel week compare between 2020 and 2023? Significant weather delays in 2020 were observed, especially at West Coast airports like LAX, SFO, and SEA, where storms may have had a disruptive impact on flight schedules. In 2023, these weather delays were notably lower—an indication of improvement in weather conditions or even operational preparedness.
2. How did the average delays (weather and carrier delays) differ between Pittsburgh (PIT) and Philadelphia (PHL) during the week of Christmas in 2023 compared to 2020? The analysis for PIT and PHL highlighted similar patterns in weather and carrier delays across both years, with 2020 showing more erratic delay patterns, especially around December 24th, likely due to weather conditions over the region. In 2023, delays were generally more consistent across the days of the week.

References

"10 Best Airlines in US - Top Carrier Rankings 2023 by Air…" *AirAdvisor*, 2023, airadvisor.com/en/top-us-airlines-rating.

"Airports in United States of America - Humanitarian Data Exchange." *Data.humdata.org*, data.humdata.org/dataset/ourairports-usa.

"Download Page." *Bts.gov*, 2017, www.transtats.bts.gov/DL_SelectFields.aspx?gnoyr_VQ=FGK&QO_fu146_ gvzr.

Cnn.com, 2024, media.cnn.com/api/v1/images/stellar/prod/230221184248-christmas-flight-cancellation-221223.jpg?c=16x9&q=w_800. Accessed 19 Dec. 2024.

Code Appendix

```
#Load December Datasets and Additional Packages
library(ggplot2)
library(dplyr)
library(tidyr)
library(knitr)
library(readr)

original2020 <- "C:/Users/felic/Downloads/DL_SelectFields (4)/DecDataset2020.csv"
flights2020 <- read.csv(original2020)

original2023 <- "C:/Users/felic/Downloads/DL_SelectFields (3)/DecDataset2023.csv"
flights2023 <- read.csv(original2023)

colnames(flights2023)
#Filter December 2020:
christmas_week2020 <- flights2020 %>%
  filter(DAY_OF_MONTH >= 20 & DAY_OF_MONTH <= 27)

#Filter December 2023:
christmas_week2023 <- flights2023 %>%
  filter(DAY_OF_MONTH >= 20 & DAY_OF_MONTH <= 27)

# Load required libraries
library(kableExtra)

# Flight Summary of Christmas week in 2020
summary2020 <- summary(christmas_week2020)
summary2020_transposed <- t(summary2020)

# PDF output Table
kable(summary2020_transposed, caption = "Flight Summary Table of Christmas Week 2020") %>%
  #Style the format such as table heading font and position
  kable_styling(
    latex_options = c("striped", "scale_down"),
    font_size = 8,
    position = "center"
  ) %>%
  #Looks at the columns and rows and adjust the lengths accordingly
  column_spec(1, bold = TRUE, width = "4cm") %>%
  column_spec(2:ncol(summary2020_transposed), width = "1.5cm") %>%
  row_spec(0, bold = TRUE, background = "#D3D3D3")
# Flight Summary Table for Christmas week in 2023
summary2023 <- summary(christmas_week2023)
summary2023_transposed <- t(summary2023)

# PDF output Table
```

```
kable(summary2023_transposed, caption = "Flight Summary Table of Christmas Week 2020") %>%
  kable_styling(
    latex_options = c("striped", "scale_down"),  # Use LaTeX-specific options
    font_size = 8,                                # Adjust font size
    position = "center"                           # Align to center by default
  ) %>%
  column_spec(1, bold = TRUE, width = "4cm") %>%  # Adjust the first column width
  column_spec(2:ncol(summary2023_transposed), width = "1.5cm") %>% # Adjust other columns
  row_spec(0, bold = TRUE, background = "#D3D3D3") # Style header row
# Data Source 2
code <- "C:/Users/felic/Downloads/L_UNIQUE_CARRIERS.csv"
airlineCode <- read.csv(code)
# Merge the datasets based on the airline code
airlineInfo2020 <- christmas_week2020 %>%
  left_join(airlineCode, by = c("OP_UNIQUE_CARRIER" = "Code"))
# Calculate the average delay time by airline
avgDelay_airlineDay2020 <- airlineInfo2020 %>%
  group_by(Description, DAY_OF_MONTH) %>%
  summarize(AverageDelay = mean(DEP_DELAY_NEW, na.rm = TRUE), .groups = "drop")

specific_airlines <- c("Delta Air Lines Inc.", "Alaska Airlines Inc.", "Hawaiian Airlines Inc
                       "United Air Lines Inc.", "American Airlines Inc.", "JetBlue Airways",
                       "Southwest Airlines Co.", "Spirit Airlines", "Allegiant Air",
                       "Frontier Airlines Inc.")
#Plotting Average
ggplot(avgDelay_airlineDay2020 %>% filter(Description %in% specific_airlines),
       aes(x = factor(DAY_OF_MONTH), y = AverageDelay, fill = Description)) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  labs(title = "Average Departure Delay by Airline for Christmas Week 2020",
       x = "Days of Christmas Week",
       y = "Average Delay (Minutes)",
       fill = "Airline") +
  theme_minimal() +
  theme(
     plot.title = element_text(hjust = 0.5, size = 10),
     legend.position = "bottom",
     legend.text = element_text(size = 5))
# Merge the datasets based on the airline code
airlineInfo2023 <- christmas_week2023 %>%
  left_join(airlineCode, by = c("OP_UNIQUE_CARRIER" = "Code"))
# Calculate the average delay time by airline
avgDelay_airlineDay2023 <- airlineInfo2023 %>%
  group_by(Description, DAY_OF_MONTH) %>%
  summarize(AverageDelay2023 = mean(DEP_DELAY_NEW, na.rm = TRUE), .groups = "drop")

specific_airlines2023 <- c("Delta Air Lines Inc.", "Alaska Airlines Inc.", "Hawaiian Airlines
                       "United Air Lines Inc.", "American Airlines Inc.", "JetBlue Airways",
                       "Southwest Airlines Co.", "Spirit Airlines", "Allegiant Air",
                       "Frontier Airlines Inc.")
#Plotting Average
ggplot(avgDelay_airlineDay2023 %>% filter(Description %in% specific_airlines2023),
       aes(x = factor(DAY_OF_MONTH), y = AverageDelay2023, fill = Description)) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  labs(title = "Average Departure Delay by Airline for Christmas Week 2023",
       x = "Days of Christmas Week",
       y = "Average Delay (Minutes)",
       fill = "Airline") +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size =10 ),
    legend.position = "bottom",
    legend.text = element_text(size = 5)
  )

# Calculate average delays for Allegiant Air
avgDelayAlleg2023 <- airlineInfo2023 %>%
  filter(Description %in% specific_airlines2023) %>%
  group_by(Description) %>%
  summarize(
    Departure = mean(DEP_DELAY_NEW, na.rm = TRUE),
    Arrival = mean(ARR_DELAY_NEW, na.rm = TRUE),
    Carrier = mean(CARRIER_DELAY, na.rm = TRUE),
    Weather = mean(WEATHER_DELAY, na.rm = TRUE),
    NAS = mean(NAS_DELAY, na.rm = TRUE),
    Security = mean(SECURITY_DELAY, na.rm = TRUE),
    LateAircraft = mean(LATE_AIRCRAFT_DELAY, na.rm = TRUE),
    Diverted = mean(DIV_ARR_DELAY, na.rm = TRUE),
    .groups = "drop"
  )

# PDF output Table
kable(avgDelayAlleg2023, caption = "Average Delay (in minutes) Based on Types of Delays for t
  kable_styling(
    latex_options = c("striped", "scale_down"),  # Use LaTeX-specific options
    font_size = 8,                                  # Adjust font size
    position = "center"                             # Align to center by default
  )
library(ggmap)

# Load airport data and select IATA codes, latitude, and longitude
airports <- read.csv("C:/Users/felic/Downloads/us-airports.csv")

airports_coords <- airports %>%
  select(latitude_deg, iata_code,longitude_deg)

# Merge the selected columns with the ORIGIN column based on the IATA codes
flight_with_coords <- airlineInfo2023 %>%
  left_join(airports_coords, by = c("ORIGIN" = "iata_code")) %>%
  #Now we can store the orgin's latitude and longitude values
  rename(
    originLat = latitude_deg,
    originLong = longitude_deg,
  )
# Merge the selected columns with the DEST column based on the IATA codes
flight_with_coords <- flight_with_coords %>%
  left_join(airports_coords, by = c("DEST" = "iata_code")) %>%
  rename(
    destLat = latitude_deg,
    destLong = longitude_deg
  )

phl_departures <- flight_with_coords %>%
  filter(ORIGIN == "PHL") %>%
  group_by(DEST)  %>%
  mutate(
    destLong = as.numeric(destLong),
    destLat = as.numeric(destLat),
    originLong = as.numeric(originLong),
    originLat = as.numeric(originLat)
  )
# Count the number of flights to each destination (DEST)
top_destinations <- phl_departures %>%
  count(DEST) %>%
  arrange(desc(n)) %>%
  head(10)
```

```r
# Filter data to keep only top 10 destinations
phl_top_10 <- phl_departures %>%
  filter(DEST %in% top_destinations$DEST)
# Define the U.S. map
library(ggrepel)
us_map <- borders("state", colour = "gray70", fill = "gray95")

# Plot the map with Philadelphia as the origin and arrows to the top 5 destinations
ggplot() +
  us_map +
  geom_point(
    data = phl_top_10,
    aes(x = destLong, y = destLat, label = DEST),
    col = "#970027", size = 2,
  ) +  # Add destination points
  geom_text_repel(
    data = phl_top_10,
    aes(x = destLong, y = destLat, label = DEST),
    col = "black", size = 10, segment.color = "gray"
  ) +  # Add labels for destinations
  geom_curve(
    data = phl_top_10,
    aes(x = originLong, y = originLat, xend = destLong, yend = destLat),
    col = "#b29e7d", size = 0.8, curvature = 0.3
  ) +  # Add arrows (curved lines) from Philadelphia to the top 5 destinations
  geom_point(
    data = phl_departures %>% filter(ORIGIN == "PHL"),
    aes(x = originLong, y = originLat),
    color = "blue", size = 8, shape = 21
  ) +  # Mark Philadelphia (PHL) with a blue dot
  labs(
    title = "Top 10 Destinations from Philadelphia (PHL) Based on Number of Flights",
    subtitle = "Flight Routes with Arrows to Popular Destinations",
    caption = "Source: Airline Data"
  ) +
  theme_void() +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 10),
    plot.subtitle = element_text(hjust = 0.5, size = 10),
    plot.caption = element_text(hjust = 1, size = 8)
  )
```

```r
# 1. Find most popular airports in the week of Christmas for 2023

# Create a dataframe that counts the number of flights and calculates average delays by ORIGI
flights_by_airport_2023 <- christmas_week2023 %>%
  group_by(ORIGIN) %>%
  summarise(
    Number_of_Flights = n(),
    Avg_Weather_Delay = median(WEATHER_DELAY, na.rm = TRUE),
    Avg_Carrier_Delay = median(CARRIER_DELAY, na.rm = TRUE)
  ) %>%
  arrange(desc(Number_of_Flights))

# Filter for the top 10 busiest airports
top10_flights_by_airport_2023 <- flights_by_airport_2023 %>%
  slice_max(order_by = Number_of_Flights, n = 10) %>%
  left_join(airlineInfo2023 %>% select(ORIGIN),
            by = "ORIGIN") %>%
  distinct()

# 2. Find most popular airlines in the week of Christmas for 2023

# Create a dataframe that counts the number of flights and calculates average delays by Airli
flights_by_airline_2023 <- christmas_week2023 %>%
  group_by(OP_UNIQUE_CARRIER) %>%
  summarise(
    Number_of_Flights = n(),
    Avg_Weather_Delay = median(WEATHER_DELAY, na.rm = TRUE),
    Avg_Carrier_Delay = median(CARRIER_DELAY, na.rm = TRUE)
  ) %>%
  arrange(desc(Number_of_Flights))

# Filter for the top 10 airlines by number of flights
top10_flights_by_airline_2023 <- flights_by_airline_2023 %>%
  slice_max(order_by = Number_of_Flights, n = 10) %>%
  left_join(airlineInfo2023 %>% select(OP_UNIQUE_CARRIER),
            by = "OP_UNIQUE_CARRIER") %>%
  distinct()

# 3. Compare mean weather delays at top 10 airports for 2023 vs. 2020

bind_rows(
  christmas_week2023 %>% mutate(Year = 2023),
  christmas_week2020 %>% mutate(Year = 2020)
) %>%
  filter(ORIGIN %in% top10_flights_by_airport_2023$ORIGIN) %>%
  group_by(ORIGIN, Year) %>%
  summarise(Mean_Weather_Delay = mean(WEATHER_DELAY, na.rm = TRUE), .groups = "drop") %>%
  ggplot(aes(x = Mean_Weather_Delay, y = reorder(ORIGIN, Mean_Weather_Delay), fill = as.facto
  geom_bar(stat = "identity", position = "dodge") +
  coord_flip() +
  labs(
    title = "Mean Weather Delays at Top 10 Airports (2023 vs. 2020)",
    x = "Mean Weather Delay (minutes)",
    y = "Airport (IATA Code)",
    fill = "Year"
  ) +
  theme_minimal()

bind_rows(
  christmas_week2023 %>% mutate(Year = 2023),
  christmas_week2020 %>% mutate(Year = 2020)
) %>%
  filter(ORIGIN %in% c("PIT", "PHL"), DAY_OF_MONTH >= 20 & DAY_OF_MONTH <= 27) %>%
  pivot_longer(cols = c(CARRIER_DELAY, WEATHER_DELAY), names_to = "Delay_Type", values_to = "
  group_by(DAY_OF_MONTH, ORIGIN, Delay_Type, Year) %>%
  summarise(Mean_Delay = mean(Delay_Value, na.rm = TRUE), .groups = "drop") %>%
  ggplot(aes(x = DAY_OF_MONTH, y = Mean_Delay, fill = as.factor(Year))) +
  geom_bar(stat = "identity", position = "dodge") +
  facet_wrap(~ ORIGIN, ncol = 2) +
  theme_minimal() +
  labs(
    title = "Mean Delay by Type, Day, and Airport (Pittsburgh and Philadelphia, 2023 vs. 2020
    x = "Day of the Month",
    y = "Mean Delay (minutes)",
    fill = "Year"
  )
```