UNIVERISTY OF MICHIGAN

SI 618 PROJECT 1

Recovery of the US Commerical Aviation Industry after COVID: An International View

Author:
Haoquan Zhou

October 27, 2022

Contents

List of Figures

1 Introduction and Motivation

Last few years are almost the toughest eva for the whole commercial aviation industry. We have seen bankruptcies, reconstructions or large-scale cutting of jobs of significant industrial figures like Alitalia, Avianca, Cathay Pacific, and Virgin Atlantic [1]. Meanwhile, two major aircraft manufacturers – Boeing and Airbus, were also struggling with enormous losses due to the extreme slump in the global aviation industry. Boeing, additionally, had to tolerate the loss caused by its grounded 737MAX model. By the end of February 2021, Airbus has reported a loss of 1 billion euro due to COVID pandemic [2]. This number becomes 11 times larger, which is 12 billion dollars for Boeing throughout 2020 [3]. Undoubtedly, the aviation industry entered a winter that tortured every companies' business. People must find a way to walk through that darkest period.

Luckily, after one year of fighting against COVID, the global transportation environment began to recover gradually. Thousands of airlines, especially the international airlines, came back to normal operations. Large orders were placed by countries like China to manufacturers including Airbus [4]. According to the International Air Transport Association (IATA)'s target, the aviation industry will cut down the loss to 9.7 billion dollars by the end of 2022 [5]. The industry is becoming more Optimistic about the future, and is trying to bring the development trend before the pandemic back.

Inspired by this circumstance, this project wants to explore the recovery of the US aviation industry, especially international passenger transportation, in a detailed manner. In this project, I will perform a data-driven analysis to answer questions including but not limited to

1. What is the impact of COVID on the US commercial aviation industry?
2. Has the US commercial aviation industry recovered completely from the COVID pandemic?
3. Is there any significant structural change made by the COVID pandemic to the US commercial aviation industry?
   (a) Is there a significant change in top international destinations considering the flights departed from the US?
   (b) Is there a significant change in US airports rank considering the flights departed from the US?
   (c) Is there a significant change in top airline carriers considering the flights departed from the US?
4. What are the current operating conditions of the US commercial aviation industry? Does the accident rate increase?
5. When will the US commercial aviation industry reach the scale before the COVID pandemic?
6. ...

During the data analysis process, big data analysis tools will be utilized to perform necessary manipulations. The results will be drawn completely based on the data analysis result.

2 Data Source

The major part of data used in this project will come from the official data provided by the U.S. Department of Transportation, under the U.S. International Air Passenger and Freight Statistics Report. Additional data sets including the Aircraft Accidents, Failures & Hijacks Data set, IATA airport Code Data set, and International Country Code Data set will also be used to assist the necessary analysis on the flight information.

1. **International_Report_Departure [6].** This is a csv file provided by the U.S. Department of Transportation. It keeps records of 988K non-stop commercial international flight departed from the U.S. airports to international points. The data set stretches a long timeline from January of 1990 to March 2022, which is designed to offer the public more access to aviation data. Among all 16 columns provided by the data set, 12 of them are useful to this project. Generally, these columns indicate the time, departure & destination airports, and the airline carrier of each flight, which can be used as categorical data for further inference.

2. **Aircraft Accidents, Failures & Hijacks Data set [7].** This is a csv file uploaded by user Deep Contractor. The data set contains 23159 rows recording aircraft incidents ranging from 1919 to 2022. In order to keep it consistent with the time scope of this project, we only consider incidents happened between 1990 and 2022. After filtering, 6987 records left, and they will be considered together with the departure records. Among all 23 columns, the incident date, aircraft model, departure & destination airports and incidents causes will contribute to this project.

3. **IATA Airports Code Data Set [8]**. IATA Airports Code Data Set is a JSON file created by GitHub user mwgg. The data set contains 28884 records providing basic information about most of the airports world wide together with its unique IATA (International Air Transport Association) 3-digit Code. Utilizing this data set will enable mapping each flight to a departure-destination city pair or country pair. Then we can further use a look up table to aggregate the international flights to continental level.

4. **Airline Codes[9] & US Airline Codes Data Set[10]** The Airline Codes data set is a csv file containing 1571 major airlines with their unique IATA code. This data set is provided by the US Bureau of Transportation Statics. Meanwhile, the US Airline Codes Data Set is crawled from Wikipedia using `pandas.read_html()` in order to compensate the incompleteness of Airline Codes data set. These data sets enable analyzing International Departure data set by carriers.

5. **Country and Continent Code List[11]** This list is a csv file created by GitHub user stevewithington. It contains a thorough look up table for country code and corresponding continent. This list helps to analyze the International Departure data set by continents.

3 Data Manipulation Methods

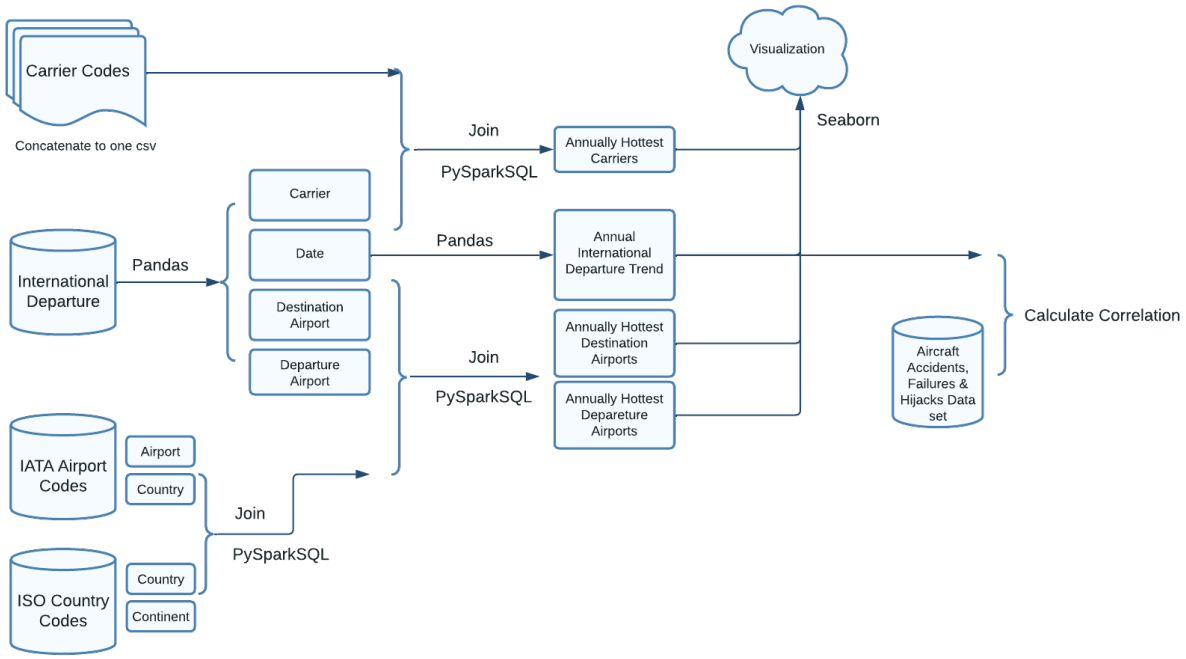

Figure 1: Data Manipulation Flow

The overall data manipulation flow is shown in Figure. 1. There are five data sets or lists adopted by this project, with the International Departure data set being the major one. Data analysis tools including Pandas, PySparkSQL and data visualization tools like Seaborn are used to process the data to support conclusions. Readers can also refer to source code `https://github.com/TaikiShuttle/SI618/blob/main/Project1/aviation_preprocess.ipynb` for details.

3.1 Processing of the International Departure data set

The data set is first loaded into a Jupyter Notebook file using

```
df = pd.read_csv("./archive/International_Report_Departures.csv")
```

command. After reading in the data, we focus on several important columns, on which we applied further manipulations:

1. `data_dte, Year, Month`: These column indicate the explicit departure date of each flight. It is in `pandas.TimeStamp` form. Thus, by applying

   ```
   df['data_dte'] = pd.to_datetime(df['data_dte'])
   df['data_dte'].describe()
   ```

   we can get a general overview of the starting and end time of this data set. Meanwhile, it is quite natural to aggregate all the records into monthly records, which will reduce the time category size and make it easier for future analysis to make sense. This aggregation process is done using

   ```
   dpt_per_month = df.groupby(["Year", "Month"])['Total'].agg(len)
   ```

2. `usg_apt`: This column represents the departure airport in its IATA code.

3. `fg_apt`: This column represents the destination airport in its IATA code.

The data set is quite clean and well-ordered, and there is no missing values as well. Therefore, there is no need to deal with missing or improper values. By now, we have completed the pre-process of this data set and it is ready to be involved in join and analysis.

3.2 Processing of the IATA Airport Codes

This data set is first loaded into the Jupyter Notebook file using

```
airport_list = pd.read_json('archive/airports.json')
```

However, we noticed that the table loaded in is wrongly configured with columns and rows exchanged. We then apply

```
airport_list = airport_list.T
```

to change the table into the right order. Also, we noticed that the `iata` column has some missing values. This may due to the size of the airport is too small to be confirmed by IATA. We will drop these columns when joining this data set with other data sets.

3.3 Processing of the Country and Continent List

We can simply load in this look up table

```
continent_list =
   pd.read_csv('archive/country-and-continent-codes-list-csv.csv')
```

and select the useful columns `Continent_Name, Country_Name` and `Two_Letter_Country_Code`.

3.4 Processing of the IATA Carrier Codes data set & US Carrier Codes data set

This data set is crawled from Wikipedia using `pd.read_html()`. The crawled results are distributed in five tables, which we concatenate to one csv file using

```
result = pd.DataFrame()
for table in tables[0:5]:
   result = pd.concat((result, table))

result.to_csv("US_airline_codes.csv")
```

Together with the file `airline_codes.csv`, we are ready to join them with the International Departure data set.

3.5 Processing of Aircraft Accidents, Failures & Hijacks data set

We first load in the data set by

```
incidents = pd.read_csv('Aircraft_Incident_Dataset.csv')
```

After inspecting the `'Incident_Date'` column, we find that there are some records have vague time. For example, some records only contains a year without sepcific month and day. These data are droped using

```
s = pd.to_datetime(incidents['Incident_Date'], errors = 'coerce')
s.dropna()
```

Also, since our focus periods is between 1990 and 2022, we also need to drop data that is before 1990. We do

```
incidents['Incident_year'] = incidents['Incident_Date'].apply(lambda
    x: x.year)
incidents = incidents[incidents['Incident_year'] >= 1990]
```

to filter out records we want. Next, we will need to group the incidents by year and month so that it matches with the International Departure data set, which further enables our correlation calculation.

```
incidents['Year'] = incidents['Incident_Date'].map(lambda x: x.year)
incidents['Month'] = incidents['Incident_Date'].map(lambda x: x.month)
incident_per_month = list(incidents.groupby(["Year",
    "Month"])['Incident_Date'].agg(len).values)
```

3.6 Join the Carrier Codes with International Departures

In order to get a glimpse on the potential changes of top airline carriers, we join the `airline_codes.csv` and `International_Report_Departures.csv` on `carrier code` using PySparkSQL. In this way, we filter out top 10 airline carriers each year for further analysis.

We also want to have a look at the international carriers that are not founded in the US. We can join the `airline_codes.csv` and `US_airline_codes.csv` on `carrier code` using PySparkSQL. We then drop the overlapping records of each data set, which are US airline carriers. Similar to former procedures, we further join the result with `International_Report_Departures.csv` and do the filtering one more time. This allows us to get the international annual top airline carriers. Note that we filter out 15 airline carriers here. This is due to the lack of data in the `US_airline_codes.csv`, which leads to a failure in dropping all the US airline carriers. Manual selection is applied to the result to drop improper records.

3.7 Join the IATA Airports Codes and ISO Country Codes

This step is relatively easy, we just use PySparkSQL to join two files on `Two_Digit_Country_Code`. It will give us an airport-country look up table.

3.8 Join the Airport-Country Lookup Table with International Departure data set

We continue to join the Country-Continent lookup table produced in former section with our main data set `International_Report_Departures.csv`. This time, we can join on both `usg_apt` and `fg_apt` to get the hottest departure places and hottest destination places.

4 Analysis and Visualization

4.1 Annual International Departure Trend

In this part, we will answer following questions raised in the introduction section:

1. What is the impact of COVID on the US commercial aviation industry?
2. Has the US commercial aviation industry recovered completely from the COVID pandemic?

To answer these questions, the annual international departure trend is calculated and visualized in 8.1. And the visualization result is shown in Figure. 2

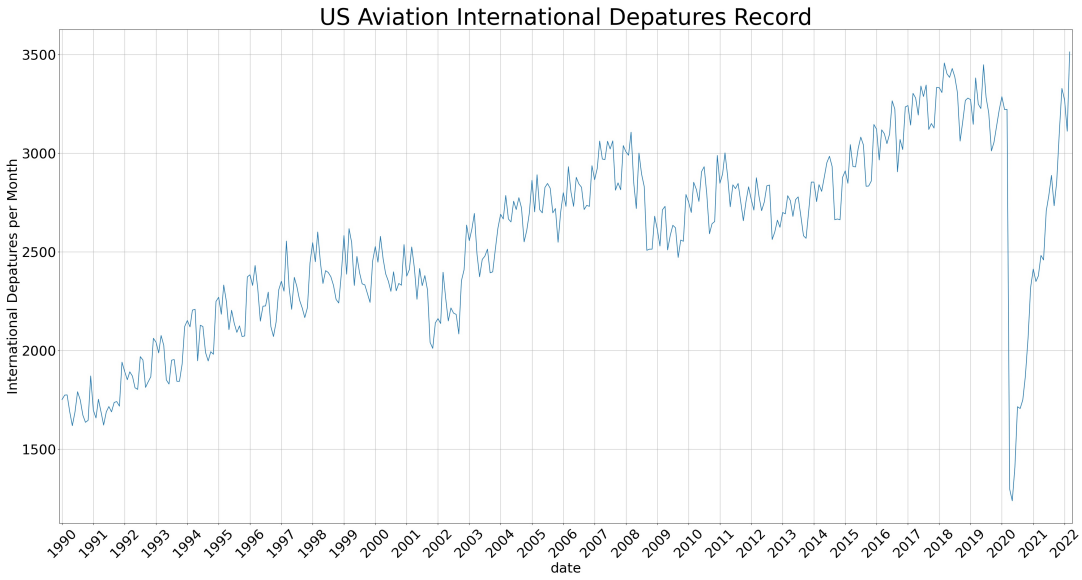

Figure 2: International Departure Trend

From the visualization we can spot without difficulty that the COVID has a extremely huge impact on the international aviation industry. The number of departures jumped to the lowest point since 1990 at the first half of 2020. This means the number of international flights were cut to about one-seventh of the original value. Such impact swallowed the industry like a tsunami, causing huge amount of loss.

Luckily, we also observe that after two years of recovery, the scale of international departures have already reached a high value. The number of departures per month in the beginning months of 2022 has already climbed to a normal level compared with those months

before COVID. This is a positives sign that the aviation industry is making its way to the normal track. But whether it will keep steady in following months needs further observation.

We can further divide the departures according to the destination continent. The result is shown in Figure.

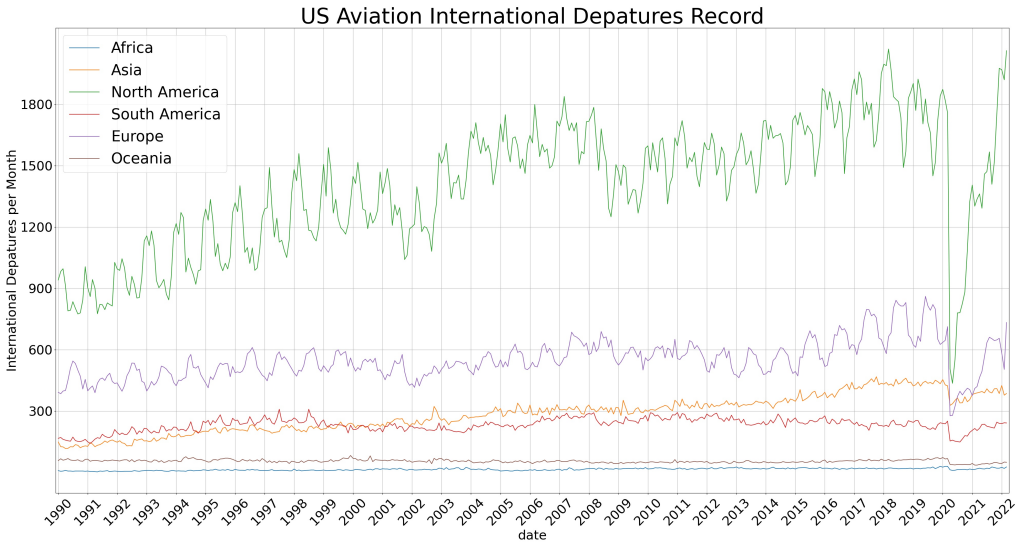

Figure 3: International Departure Trend by Continent

We can see that no matter which continent the departures are heading, they are all influenced by COVID significantly. The flights to North America, i.e. Canada dropped the most drastically. And flights to other continents all show decline in different scale. Also, some airlines towards Europe, Asia, Oceania have not recovered to the value before COVID. This indicates the huge and long-lasting impact exerted by the pandemic.

4.2 Changes Brought by COVID Pandemic

In this part, we will look at the data in a more detailed fashion, and try to discover some changes brought by the COVID. We will using the data manipulated before and derive several ranks according to corresponding number of departures. By inspecting the rank changes, we may draw some useful conclusions.

4.3 Hottest Destinations

By writing PySparkSQL queries in 8.2, we can get a list of top 10 hottest destinations annually. We specifically focus on the change of this list from 2019-2022, which is after the appearance of COVID. The rank change is shown in Figure. 4 using visualization method in 8.3.

From the figure, we can draw several conclusions:

1. Countries in North America and South America like Canada and Mexico are steady in the top destinations. COVID exerts little influence on them.

2. East Asia countries like Japan, China, and Korea are influenced severely by the COVID pandemic, with their ranks drop significantly after 2020. This may due to the strict policies published by these countries to limit the incoming flights in order to prevent the COVID from spreading into their country.

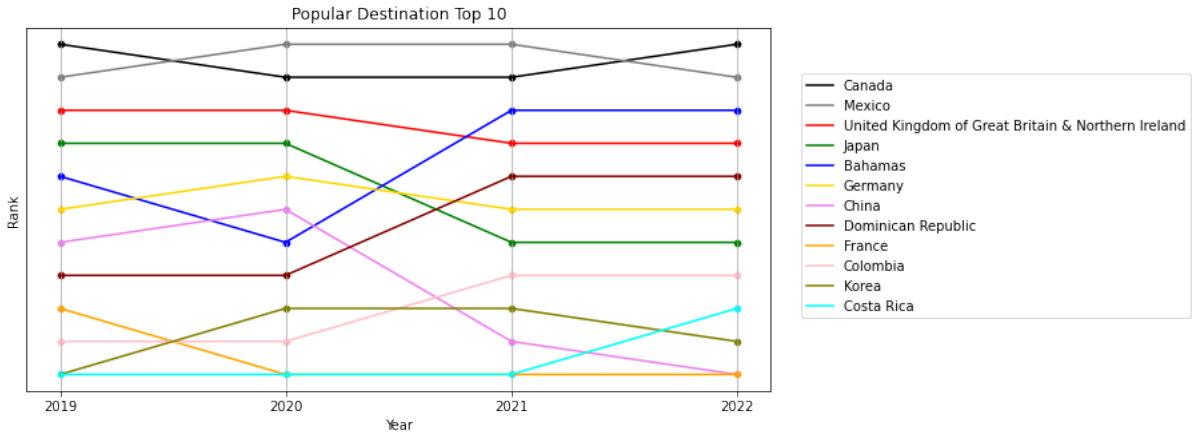

Figure 4: Hottest Destinations

4.4 Hottest Airports

By writing the PySparkSQL queries in 8.4, we can get a list of top ranked US airports with its rank evaluated by annual departure. Again, we are curious about the time period from 2019 to 2022. The rank change is shown in Figure. 5 using the visualization method 8.3.

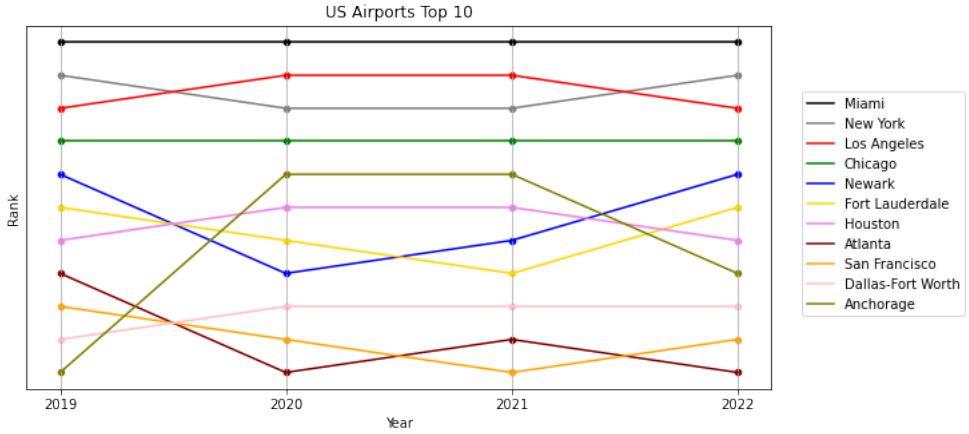

Figure 5: Hottest US Airports

From the figure, we can draw several conclusions:

1. The top 4 airports are stable. The COVID does not influence them much.

2. One thing we could discover is that, in 2020, Miami is far more stable than the other airports. We also notice that Anchorage used to be kicked out of the top 10 list, but it made to rank 5 in 2020. One important reason is that these two airports are the core inter-connection station for international cargo transportation. Almost all the Asia-North America cargo flights get connected in Anchorage, while almost all the North America-South America cargo flights get connected in Miami. During the COVID pandemic, these two airports play an important role in transporting the essential goods and materials, helping lots of people in needs. Their contributions deserve their ranks.

4.5 Hottest Carriers

In this part, we will rank the airline carriers by their international flights departed from the US. Since the US domestic carriers occupy a large scale in US international departures, we consider two cases. Namely, one case considers all carriers as a whole, while the other case only consider international carriers with their founding place outside of the US. We can write PySparkSQL queries in 8.5 to deal with both cases.

4.5.1 Hottest Carriers Considering US Domestic carriers

The visualization procedure is similar using 8.3. The result is shown in Figure. 6.

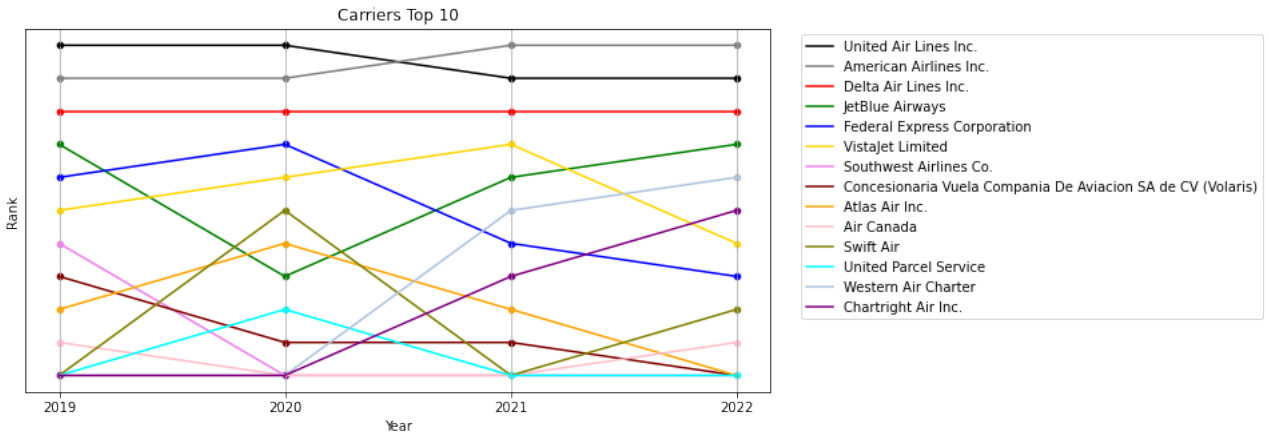

Figure 6: Hottest Carriers

From the visualization, we find several significant changes after COVID:

1. The biggest three companies, United, American Airlines, Delta are stable and are not influenced by COVID.

2. We noticed that in 2019 and years before, the top airline carriers are all big companies offering public transportation. Including three largest airlines offering passenger flights and FedEx, UPS, ALTAS offering international cargo transportation. However, since 2020, more private airline carriers began to took place, like VistaJet, Western Air Charter, and Chartright Air. This may due to the fact that wealthy people want to use private planes to travel in order to preventing themselves from catching COVID during the flight.

3. The international cargo airlines including FedEx, UPS, ALTAS begin to loss their rank after COVID. This may be caused by less international cargo orders. We may further infer that the pessimistic world economy market results in less imports and exports for countries.

4.5.2 Hottest International Carriers

The result of visualization is shown in Figure. 7
By observing the result, we can conclude that:

1. The COVID has exerted a significant influence on the international airline carriers. Lots of the top international carriers like WestJet, British Airways, and Norwegian Air have large reduce in the amount of flights.

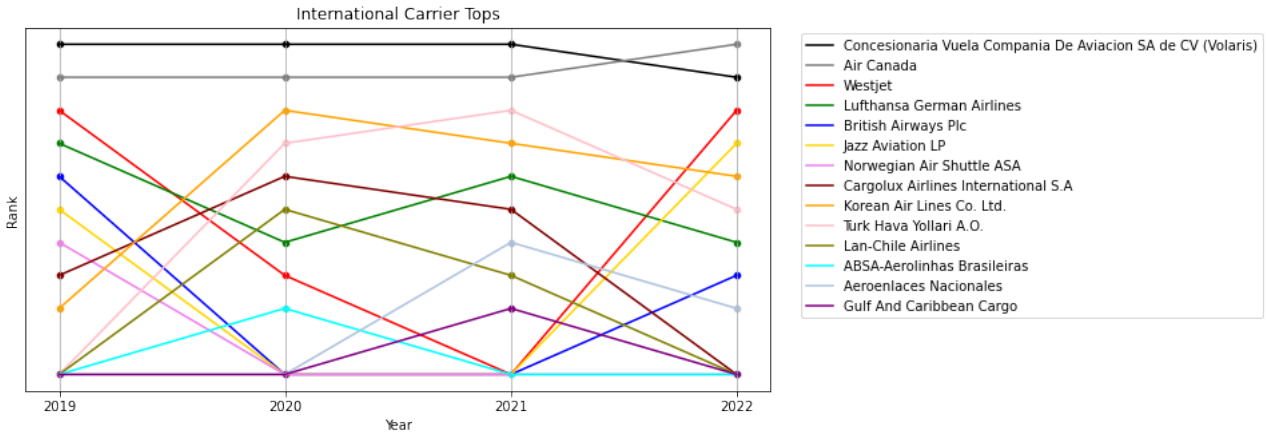

Figure 7: Hottest International Carriers

2. However, there are also some international carriers improve their ranks since 2020 like Turkish Airline and Korean Airlines. Meanwhile Lufthansa successfully maintain its top position. The reason for their high ranks is that they develop cargo transportation during these year including using passenger planes to carry cargo and buying new cargo planes. This method may reduce their loss during the pandemic period. And it seems to be proved efficient based on this figure.

4.6 Incident Frequency

It is natural for one to doubt whether the airplanes are still in good maintenance under the COVID pandemic condition. We want to answer the question raised earlier–"What are the current operating conditions of the US commercial aviation industry? Does the accident rate increase?"

To answer this question, we group the incident records data set pre-processed before into monthly records and plot the change trend in Figure. 8

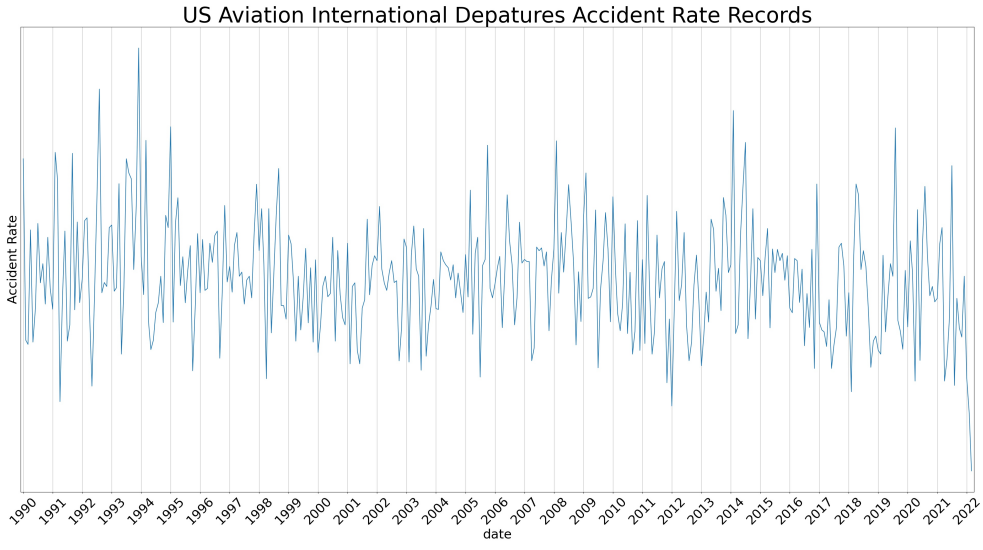

Figure 8: Accident Rate Trend

It seems that the accident rate has no significant change since COVID. We further cal-

culate the correlation coefficient with the number of departures since COVID. The result is 0.16, which means they are not quite related.

```
np.corrcoef(y[-27:-1], incident_per_month[-27:-1])
```

We are now able to conclude that the airplane incidents rate does not raise after COVID, which may indicate that the planes are well maintained though impacted by the pandemic.

4.7 ARIMA Prediction

In order to answer the question ”When will the US commercial aviation industry reach the scale before the COVID pandemic?”, we decide to use the Auto Regressive Integrated Moving Average (ARIMA) Model to predict the future departure data for next 12 months. ARIMA combines the techniques of autoregression and moving average to model a time series. Autoregression forecasts the variable of interest using a combination of its past values, while moving average uses past forecast errors to perform this task.

The basic mathematical formula for ARIMA is

$$\hat{y}_t = \mu + \phi_1 y_{t-1} + ... + \phi_p y_{t-p} + \theta_1 e_{t-1} + ... + \theta_q e_{t-q}$$

where μ is a constant. ϕ is the coefficients for AR. θ is the coefficients for MA. p, q are the lags values. By learning the coefficients from historical data, this equation will give the prediction for the value $\hat{y}_t$ for point t. We construct the codes in 8.6 and get a training & visualization result shown in Figure. 8.6.

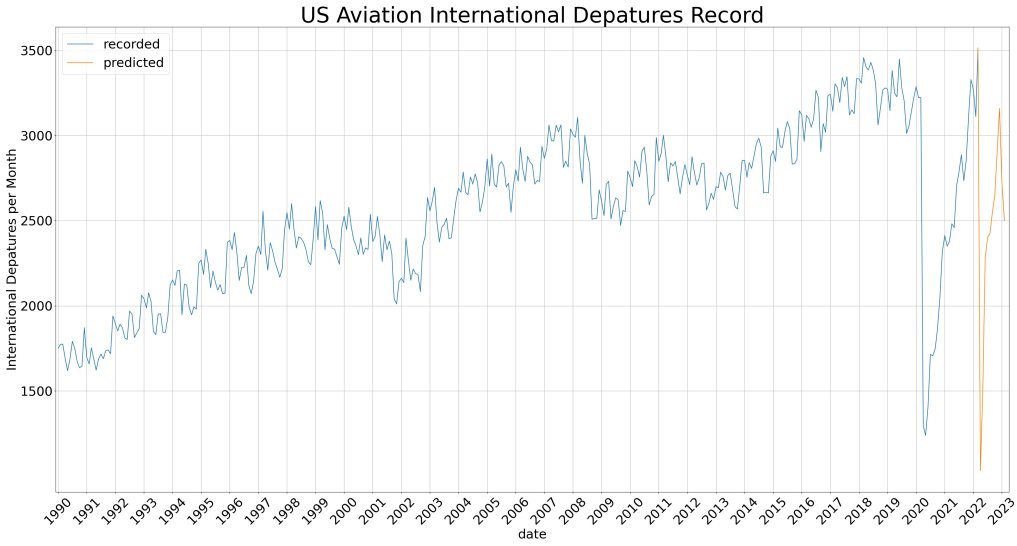

Figure 9: ARIMA Prediction Result

From the figure, we find that if we only take the past 18 months into consideration, the model tends to believe the departures will begin to swing for a few months and the stable at a value around 2750. This is completely reasonable, since the impact of the pandemic cannot be erased such a short time, not to mention that some of the countries are still fighting against the COVID. It seems that the aviation industry still needs time to completely recover.

5 Challenges and Limitations

5.1 Challenges

The major challenges I encountered in this project is finding a proper data set which records as many flights record as possible. Most of the data set published online only contains part of the flight records, or only contains flights in a short range before COVID, like 2016-2018. The complete data sets provided by certain companies require membership and are usually not open sources. To find a proper data set, I carefully checked every page on their data set description and clicked every single link to see whether there are related information. I finally found a data set which contains flights from 1990-2019, and in its description page, I found the website of Department of Transportation, where I luckily see the updated version of it.

It is also hard to find a complete airline codes list. All of the data sets online only contain part of the airline carriers. There are so many airline carriers in this world, and some of them are just established with a small scale targeted on certain group of passengers. It is hard for data collector to get information from those companies. To deal with this situation, I tried to search for different data sets and tried to join them together. Meanwhile, I evaluated the loss of dropping records and find that major carriers with large company scale are well recognized and recorded. So I realized that simply dropping the data with empty airline code will not cause a huge loss, which is exactly what I did later.

5.2 Limitations

Due to the time limitation, this project does not cover certain topics, including:

1. What is the impact of COVID on international freight transportation. Since the cargo flight has no risk on having passenger catching COVID on board, will it be influenced less compared with passenger flights?

2. What are the major causes of air plane incidents after COVID. Is it quite different from that before the COVID pandemic? Do the incidents related more on airplane part failure after pandemic?

3. What potential suggestions can be made to help airline companies go through this tough time?

6 Conclusion

There is a well-known saying in the aviation industry. It goes as ”Aviation industry is an industry developed through learning from failures.” People has been dreaming flying for more than one century, and this dream has never been stopped by air crashes or depressive markets. It is now about making good decisions and be prepared to go back to the blue sky!

7 References

[1] *List of airlines impacted by the COVID-19 pandemic*. Oct. 2022. URL: https://en.wikipedia.org/wiki/List_of_airlines_impacted_by_the_COVID-19_pandemic#:~:text=By%208%20October%202020%2C%2043,Europe%20were%20also%20risking%20bankruptcy.

[2] *Airbus reports loss of €1bn after Covid, and could shed 15,000 jobs*. Feb. 2021. URL: https://www.theguardian.com/business/2021/feb/18/airbus-loss-1bn-covid-jobs.

[3] David Schaper. *Pandemic piles on already reeling Boeing, leading to nearly $12 billion loss in 2020*. Jan. 2021. URL: https://www.npr.org/2021/01/27/961339159/pandemic-piles-on-already-reeling-boeing-leading-to-nearly-12-billion-loss-in-20.

[4] *Airbus awarded New Orders in China*. July 2022. URL: https://www.airbus.com/en/newsroom/press-releases/2022-07-airbus-awarded-new-orders-in-china.

[5] *Travel recovery rebuilding airline profitability - resilient industry cuts losses to $9.7 billion*. URL: https://www.iata.org/en/pressroom/2022-releases/2022-06-20-02/.

[6] Randall Keizer. *International_Report_Departures: Department of Transportation - Data Portal*. Oct. 2022. URL: https://data.transportation.gov/Aviation/International_Report_Departures/innc-gbgc.

[7] Deep Contractor. *Aircraft accidents, Failures & Hijacks Dataset*. Feb. 2022. URL: https://www.kaggle.com/datasets/deepcontractor/aircraft-accidents-failures-hijacks-dataset?resource=download.

[8] Mwgg. *MWGG/Airports: A JSON database of 28K+ airports with ICAO/IATA codes, names, cities, two-Letter country identifiers, elevation, Latitude amp; Longitude, and a timezone identifier*. URL: https://github.com/mwgg/Airports.

[9] *Airline codes*. Oct. 2019. URL: https://www.bts.gov/topics/airlines-and-airports/airline-codes.

[10] *List of airlines of the United States*. Oct. 2022. URL: https://en.wikipedia.org/wiki/List_of_airlines_of_the_United_States.

[11] stevewithington 262588213843476. *Country and continent codes list*. URL: https://gist.github.com/stevewithington/20a69c0b6d2ff846ea5d35e5fc47f26c.

8 Appendix

All the source code in this appendix can be found in https://github.com/TaikiShuttle/SI618/tree/main/Project1

8.1 Calculate International Departure Trend

```
depature_per_month = df.groupby(["Year", "Month"])['Total'].agg(len)
```

```
time_index = pd.date_range(start = '1990-01-01', end = '2022-03-01',
   freq = 'MS')
fig, ax = plt.subplots(1, 1)
fig.set_size_inches((40,20))
sns.lineplot(time_index,y)
plt.xlim((pd.to_datetime('1989-12-01'), pd.to_datetime('2022-04-01')))
plt.xticks(size = 30, rotation = 45)
ax.xaxis.set_major_locator(mdates.YearLocator())
date_form = mdates.DateFormatter("%Y")
ax.xaxis.set_major_formatter(date_form)
plt.yticks([1500,2000,2500,3000,3500],size = 30)
plt.xlabel('date',size = 30)
plt.ylabel('International Depatures per Month', size = 30)
plt.title("US Aviation International Depatures Record", size = 50)
plt.grid()
plt.show()
```

8.2 Get Top Destinations

```
from pyspark import SparkContext
from pyspark.sql import SQLContext

if __name__ == '__main__':
   sc = SparkContext(appName = "umsi618f22project")
   sqlc = SQLContext(sc)

   dpt = sqlc.read.csv('archive/International_Report_Departures.csv',
      header = True)
   airport_codes = sqlc.read.csv('archive/airports.csv', header = True)
   country_codes =
      sqlc.read.csv('archive/country-and-continent-codes-list-csv.csv',
      header = True)

   sqlc.registerDataFrameAsTable(dpt, 'dpt')
   sqlc.registerDataFrameAsTable(airport_codes, 'airport_codes')
   sqlc.registerDataFrameAsTable(country_codes, 'country_codes')

   hot_destination = sqlc.sql('''
      SELECT Year, Country_Name, NUM_dpt FROM
         (SELECT Year, Country_Name, NUM_dpt, ROW_NUMBER() OVER
            (PARTITION BY Year ORDER BY NUM_dpt DESC) AS Annual_Rank FROM
            (SELECT dpt.Year, T2.Country_Name, COUNT(*) AS NUM_dpt FROM
               dpt LEFT JOIN
               (SELECT airport_codes.iata, country_codes.Continent_Name,
                  country_codes.Country_Name FROM airport_codes
               LEFT JOIN country_codes ON airport_codes.country =
                  country_codes.Two_Letter_Country_Code
               WHERE iata IS NOT NULL) AS T2
            ON dpt.fg_apt = T2.iata
            WHERE Country_Name IS NOT NULL
```

```
            GROUP BY Year, Country_Name
            ORDER BY Year, NUM_dpt DESC))
      WHERE Annual_Rank <= 10
   ''')

   hot_destination.coalesce(1).write.csv("project_hot_destination", sep
      = ",")
```

8.3 Plot Top

```
import matplotlib.ticker as ticker

def plot_hot(hottest_list, item_name:str, title:str):
   dict_t = {}
   color_list = ['black', 'gray', 'red', 'green', 'blue', 'gold',
      'violet', 'maroon', 'orange', 'pink', 'olive', 'aqua',
      'lightsteelblue', 'purple', 'light green']
   for index, element in enumerate(hottest_list):
      for i, item in enumerate(element[item_name]):
         if item not in dict_t.keys():
            dict_t[item] = np.zeros(4)
            dict_t[item][index] = 10 - i

         else:
            dict_t[item][index] = 10 - i

   fig, ax = plt.subplots(1, 1)
   fig.set_size_inches((10,5))
   for color_index, item in enumerate(dict_t.keys()):
      ranks = dict_t[item]
      sns.lineplot(pd.date_range(start = '2019-01-01', end =
         '2022-01-01', freq = 'AS'), ranks, label = item.split(',')[0]
         if isinstance(item, str) else item, color =
         color_list[color_index])
      sns.scatterplot(pd.date_range(start = '2019-01-01', end =
         '2022-01-01', freq = 'AS'), ranks, color =
         color_list[color_index])
   # plt.xlim((pd.to_datetime('1989-12-01'),
      pd.to_datetime('2022-04-01')))
   plt.xticks(pd.date_range(start = '2019-01-01', end = '2022-01-01',
      freq = 'AS'))
   ax.xaxis.set_major_locator(mdates.YearLocator())
   date_form = mdates.DateFormatter("%Y")
   ax.xaxis.set_major_formatter(date_form)
   ax.yaxis.set_major_locator(ticker.NullLocator())
   plt.yticks()
   plt.xlabel('Year')
   plt.ylabel('Rank')
   plt.title(title)
   plt.legend(loc = (1.04,0.2))
```

```
    plt.grid()
    plt.show()
```

8.4 Get Top Airports

```
from pyspark import SparkContext
from pyspark.sql import SQLContext

if __name__ == '__main__':
    sc = SparkContext(appName = "umsi618f22project")
    sqlc = SQLContext(sc)

    dpt = sqlc.read.csv('archive/International_Report_Departures.csv',
        header = True)
    airport_codes = sqlc.read.csv('archive/airports.csv', header = True)
    country_codes =
        sqlc.read.csv('archive/country-and-continent-codes-list-csv.csv',
        header = True)

    sqlc.registerDataFrameAsTable(dpt, 'dpt')
    sqlc.registerDataFrameAsTable(airport_codes, 'airport_codes')
    sqlc.registerDataFrameAsTable(country_codes, 'country_codes')

    hot_airport = sqlc.sql('''
        SELECT iata, city, year, NUM_dpt FROM
            (SELECT iata, city, year, NUM_dpt, ROW_NUMBER() OVER (PARTITION
                BY year ORDER BY year, NUM_dpt DESC) AS Annual_Rank FROM
                (SELECT iata, city, year, COUNT(*) AS NUM_dpt FROM
                    (SELECT airport_codes.iata, airport_codes.city, dpt.Year
                        FROM
                    dpt LEFT JOIN airport_codes ON dpt.usg_apt =
                        airport_codes.iata
                    WHERE Year IS NOT NULL AND iata IS NOT NULL)
                GROUP BY iata, city, year
                ORDER BY year, NUM_dpt DESC))
        WHERE Annual_Rank<=10
    ''')

    hot_airport.coalesce(1).write.csv("project_hot_airport", sep = ",")
```

8.5 Get Top Carriers

```
from pyspark import SparkContext
from pyspark.sql import SQLContext

if __name__ == '__main__':
    sc = SparkContext(appName = "umsi618f22project")
```

```
sqlc = SQLContext(sc)

dpt = sqlc.read.csv('archive/International_Report_Departures.csv',
   header = True)
carrier_codes = sqlc.read.csv('archive/airline_codes.csv', header =
   True)
US_carrier_codes = sqlc.read.csv('archive/US_airline_codes.csv',
   header = True)

sqlc.registerDataFrameAsTable(dpt, 'dpt')
sqlc.registerDataFrameAsTable(carrier_codes, 'carrier_codes')
sqlc.registerDataFrameAsTable(US_carrier_codes, 'US_carrier_codes')

hot_carrier = sqlc.sql('''
   SELECT Year, Airline, carrier, NUM_dpt FROM
      (SELECT Year, Airline, carrier,NUM_dpt, ROW_NUMBER() OVER
         (PARTITION BY Year ORDER BY Year, NUM_dpt DESC) AS
         Annual_Rank FROM
         (SELECT Year, Airline, carrier, COUNT(*) as NUM_dpt FROM
            (SELECT dpt.Year, dpt.carrier ,carrier_codes.Description
               AS Airline FROM
            dpt LEFT JOIN carrier_codes ON dpt.carrier =
               carrier_codes.Code)
         GROUP BY Year, Airline, carrier))
   WHERE Annual_Rank <= 10
''')

hot_carrier.coalesce(1).write.csv("project_hot_carrier", sep = ",")

international_hot_carrier = sqlc.sql('''
   SELECT Year, Airline, carrier, NUM_dpt FROM
      (SELECT Year, Airline, carrier,NUM_dpt, ROW_NUMBER() OVER
         (PARTITION BY Year ORDER BY Year, NUM_dpt DESC) AS
         Annual_Rank FROM
         (SELECT Year, Airline, carrier,COUNT(*) as NUM_dpt FROM
            (SELECT dpt.Year, dpt.carrier ,T1.Description AS Airline
               FROM
            dpt LEFT JOIN
               (SELECT carrier_codes.Code, carrier_codes.Description,
                  US_carrier_codes.Airline FROM carrier_codes LEFT
                  JOIN US_carrier_codes ON
               carrier_codes.Code = US_carrier_codes.IATA
               WHERE Airline IS NULL) AS T1
            ON dpt.carrier = T1.Code)
         WHERE Airline IS NOT NULL
         GROUP BY Year, Airline, carrier))
   WHERE Annual_Rank <= 15
''')

international_hot_carrier.coalesce(1).write.
csv("project_international_hot_carrier", sep = ',')
```

8.6 ARIMA

```
import itertools
import statsmodels.api as sm
import statsmodels.graphics.tsaplots as tsplt
from statsmodels.tsa.arima.model import ARIMA

predicted_number = 12

data = y
length_data = len(data)

buffer = []
for i in range(predicted_number):
    train_data = data[length_data - 18 + i: length_data + i] # use the
        departure number of 12 months before target month to train a model
    model = ARIMA(train_data, order = (4,1,4))

    # learning
    result = model.fit()

    # predict the next value
    pred = result.predict(i, i, dynamic = True)
    data:np.ndarray
    data = np.append(data, pred)
    print(len(data))

    pred = np.array(pred)
    buffer.append(pred)
    print ("i = ",i,pred)

def avg(list_t):
    list_t:list
    sum = 0
    for element in list_t:
        sum += element
    return sum/len(list_t)

def MA(buffer):
    result = []
    for index, element in enumerate(buffer):
        if index < 3:
            result.append(avg(buffer[0:index + 1])) # if there are fewer
                than three elements in buffer, just calulate the average
        else:
            result.append(avg(buffer[index - 2:index+1]))
    return result

buffer = MA(buffer)

lis = np.array([y[-1]])
for element in buffer[1:]:
```

```
   lis = np.append(lis, element)

fig, ax = plt.subplots(1, 1)
fig.set_size_inches((40,20))
sns.lineplot(time_index,y, label = 'recorded')
time_index_2 = pd.date_range(start = '2022-03-01', end = '2023-02-01',
   freq = 'MS')
sns.lineplot(time_index_2, lis, label = "predicted")
plt.xlim((pd.to_datetime('1989-12-01'), pd.to_datetime('2023-04-01')))
plt.xticks(size = 30, rotation = 45)
ax.xaxis.set_major_locator(mdates.YearLocator())
date_form = mdates.DateFormatter("%Y")
ax.xaxis.set_major_formatter(date_form)
plt.yticks([1500,2000,2500,3000,3500],size = 30)
plt.xlabel('date',size = 30)
plt.ylabel('International Depatures per Month', size = 30)
plt.title("US Aviation International Depatures Record", size = 50)
plt.legend(fontsize = 30)
plt.grid()
plt.show()
```