The Impact of Covid-19 on Air Traffic: Spatiotemporal Forecasting and Benchmarking

Adelle Driker, Bo Yan, Yuan Hu

Advisor: Professor Rose Yu

Question Formulation

Many global industries have been affected by the COVID 19 pandemic, the airline industry being one of the most heavily hit. As a result, flight trends around the world have drastically deviated from their normal patterns. This presents an interesting scenario to explore, especially with the transience and abnormality of effects of the Covid data. The phenomenon also created uncertainty for both passengers and airline companies, especially due to the multiple waves of virus mutations prompting the following questions: How should airlines plan future flights? When should passengers schedule their travels? In other words, given a country's COVID situation, how should an airline/passengers plan ahead? Through EDA, some particular patterns were revealed - for example, the locality and distribution of flights around the world. Using time series flight and Covid data, we aim to forecast how many flights will be leaving a given country around the world on a given date. We predict that the more Covid cases a country has, the fewer flights will be departing that country. In order to test if our hypothesis is correct, we plan to use evaluation metrics such as ME, MSE, MAE, RMSE, and R^2 scores on our model results.

Team

Our team is composed of three key members - Bo Yan, Yuan Hu, and Adelle Driker. Based on each team member's strengths and experience, we have assigned our roles in the following way: Bo, as the Record Keeper as well as the Software/ML/DL Engineer, will manage the project GitHub repo and help provide additional understanding in the Deep Learning domain with her previous experience in the field. Yuan will be the Budget Manager and Data Engineer, and will be in charge of tracking resources used by the team, mapping out the ETL and code automation process, and providing insight for the visualization portion of the project. Finally, as the Project Coordinator/Manager and Data/Business Analyst, Adelle will be responsible for maintaining contact between the Group and the Advisor, tracking project progress, and aiding with coding/analytical tasks.

Throughout the duration of the Capstone, our roles and responsibilities have slightly evolved based on the state of the project. As a team, this Capstone project also helps us build our team's values, core, and spirit. It's all about people. We work together. We are one team, we respect and trust each other. This project helps us to learn the true reality of an end-to-end project. We will need to collaborate with team members who have various levels of experience on different parts of the project.

Data Pipeline

Current design of the data pipeline covers data flows from data source to a reliable data access/querying point, which includes:

- Sourcing Data: gzip files and CSVs will be downloaded from the website and saved to AWS S3 bucket <air-traffic-raw>, this process will be implemented with

a DAG running cURL script in Airflow. The S3 lambda function will be triggered to unzip and save it back to S3 bucket <air-traffic-csv> whenever there is a new zipped file uploaded.

- Data Preprocessing: data will be extracted from S3 through boto3 SDK for preprocessing, which includes but is not limited to data cleaning, feature engineering, timestamp parsing.
- Data Aggregation: based on the metrics taken, preprocessed data next will be aggregated according to the need of analysis and modeling.
- Formatting tables: in order to import to influxdb, data will be further transformed, based on metrics taken, tables(aka measurements in influxdb) will be generated, attributes will be transformed to fields, tags will be added as needed.

Once the processed data was loaded into influxdb, that data will be accessed via python API (influxdb-client) with influxQL or Flux language.

The entire pipeline will be implemented and automated by DAGs using Apache Airflow. This will enable us to set a time scheduler to run specific tasks or steps. The air traffic data is currently updating monthly, we could refresh the data by month, or we may leave a holdout dataset for data streaming under the task management of Airflow.

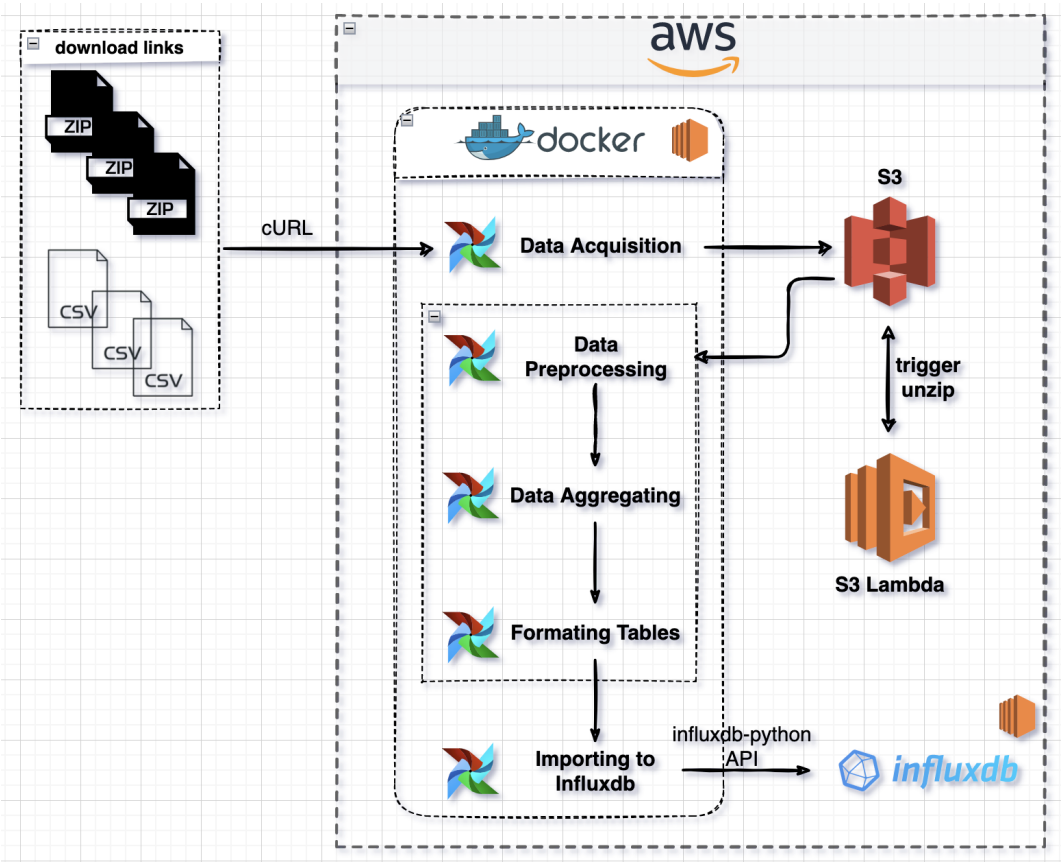


Figure 1. Diagram of Major Data Pipeline Components

Exploratory Data Analysis and Preprocessing

EDA

1. What's the trend in the number of global flights compared to Covid case totals?

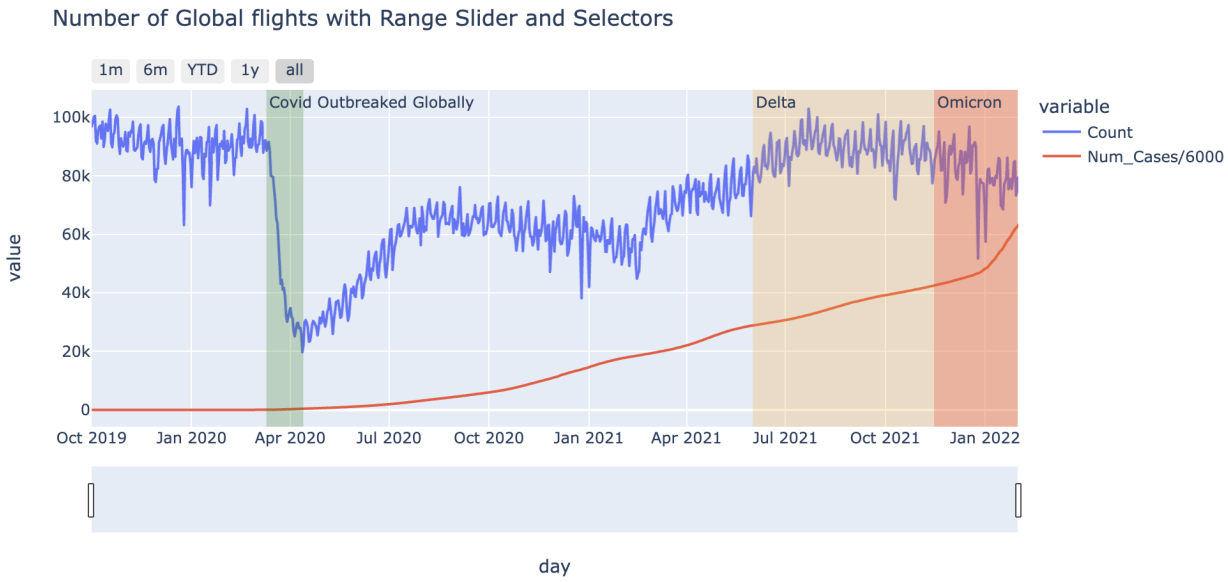


Figure 2. Number of Global Flights

- As confirmed Covid cases began to increase, a drastic drop become evident in March 2020
    a. General trend of flight numbers shows a continued steady recovery beginning June 2020
- Changes in cumulative Covid case totals are not uniform and indicate fluctuations
- Steep increase in Covid cases in Dec 2021 and Jan 2022 coupled with decrease in flights indicates Omicron wave

2. What is the trend number of departing aircrafts by the Top 6 Airlines in the United States?

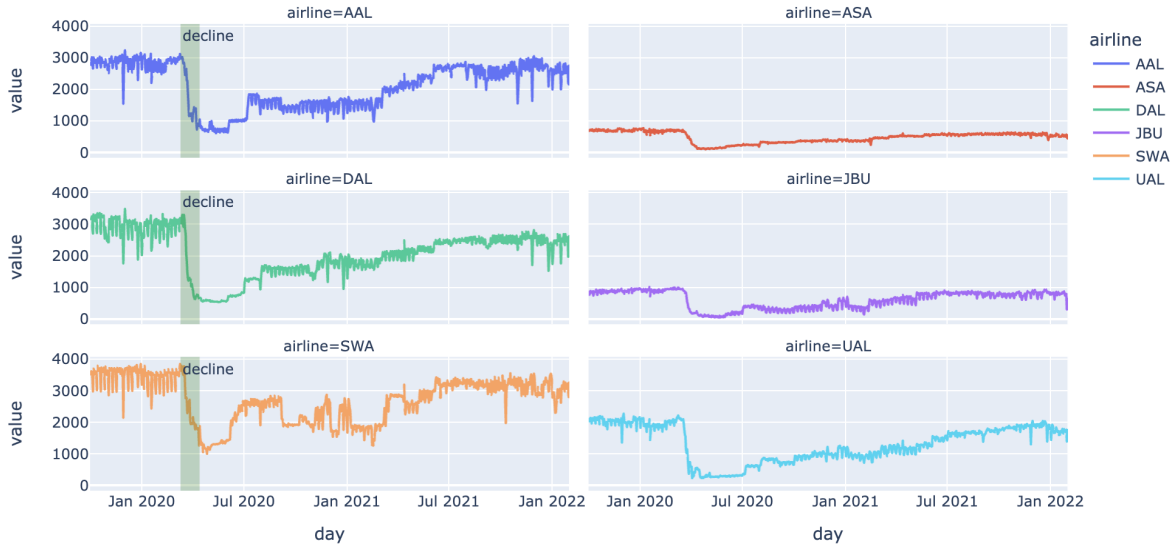


Figure 3. Number of Departing Aircrafts

- Drastic drop evident around March 2020, as the stay at home order and various travel restrictions were implemented
- A steady recovery beginning in June 2020 is present
    I. Southwest Airlines tends to fluctuate more in the number of operating flights between July 2020 to July 2021
- Regardless of how many flights the airlines serviced in 2019, the same drop and overall trend are apparent among all six airlines

3. What's the trend in the number of departing aircrafts from the Top 6 Airports in the United States during a pandemic?

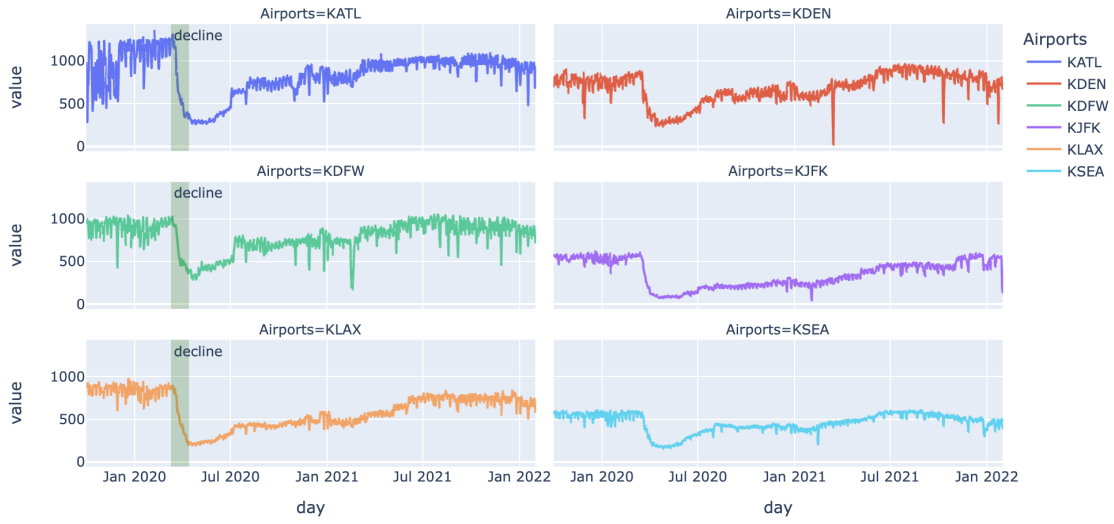


Figure 4. Number of Global Flights for the Top 6 Airports in the US

- A similar trend as in Task 2 is apparent in the Top Six Airports as well

- Airports in different regions in the US were affected in very similar ways due to federal guidelines
- A shallow decline reflecting the impact of the Delta and Omicron variants is present beginning August 2021
- Sharp dips due to data collection issues (e.g. problems with tracking sensors)

4. What is the change in the number of departing aircrafts daily from Dec, 2021?

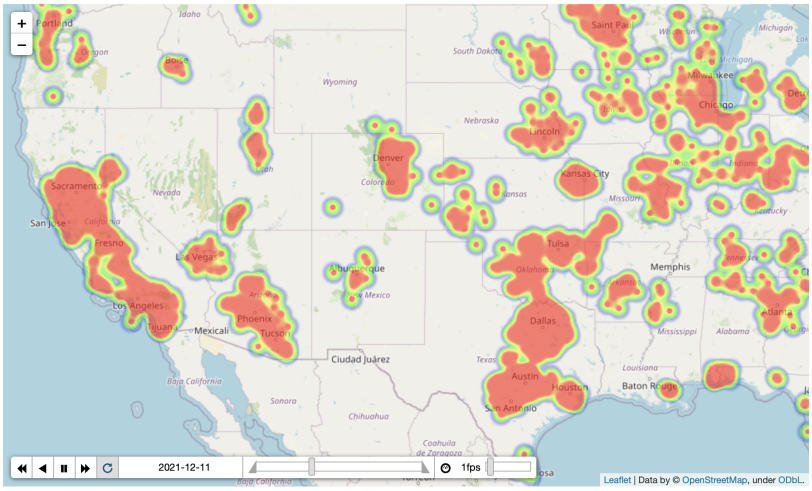


Figure 5. Heatmap for the Number of Departing Aircrafts

Graph above represents the number of departing flights from the majority of US airports. As of now, the data that is shown only covers Dec, 2021, and the time sliding interval has been set to days. Once all the data has been processed, the animation will cover the span of the entire pandemic with a monthly sliding window which will help monitor a whole picture of flight trends after the outbreak of COVID-19.

5. A normal day of global flight “Network” in the United States.

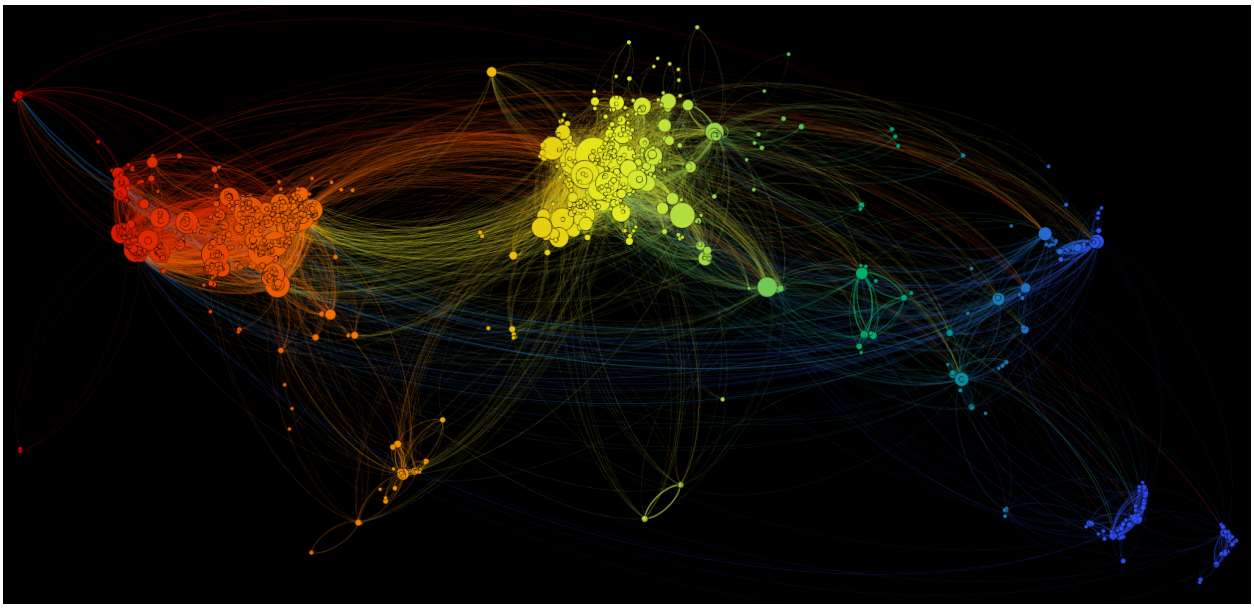


Figure 6. Global Flights Network

This directed node-link network exhibits worldwide flight trajectories on Dec 26, 2021. Nodes represent source and destination airports, edges represent a flight trajectory from source airport to destination. Node size indicates its degree of centrality, which evaluates the number of flights from the same source airport. A gradient color palette was chosen to help distinguish the geo-location of different continental and countries. We can see that most flights are concentrated within North America (specifically the US) and Europe, and in between the two continents.

6. Hypothesis testing: whether the outbreak of the Omicron variant wave affected the number of departing flights at Los Angeles airport.

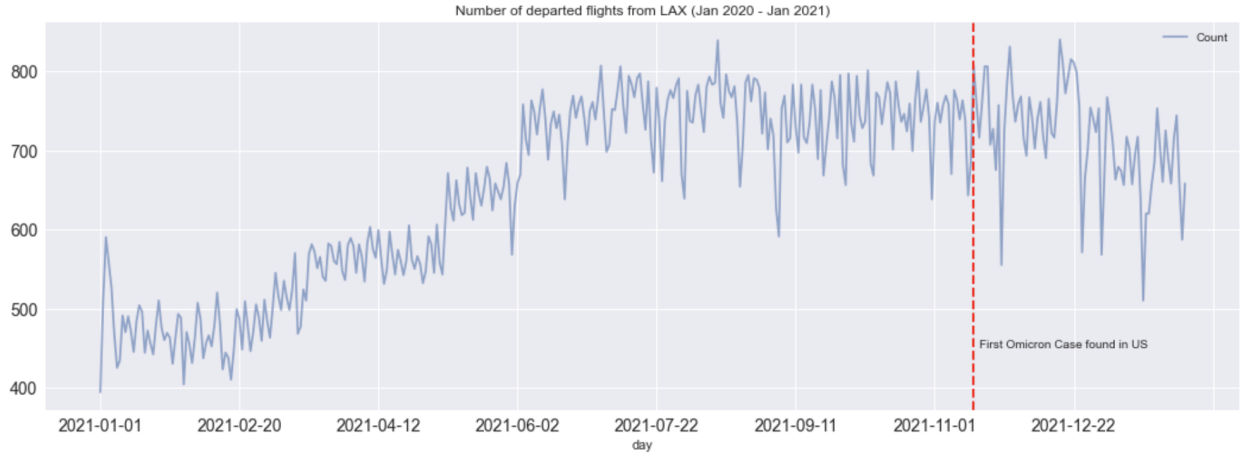


Figure 7. Number of Departed Flights from LAX

Time series decomposition and normality check:

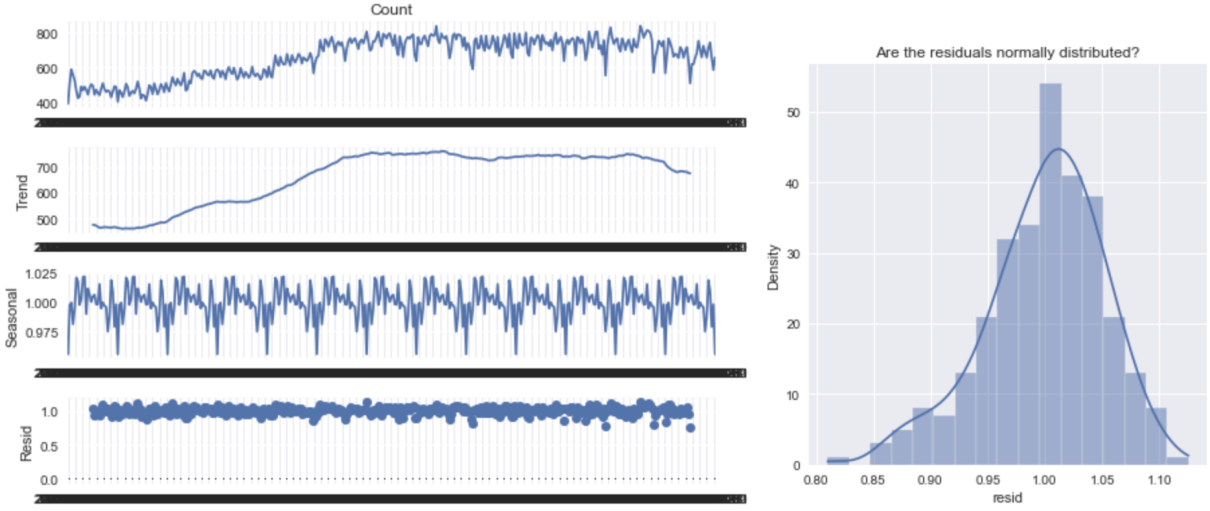


Figure 8. Time Series Decomposition and Normality Check

Once the distribution of residuals passed the normality test, a t-test on the number of departures was run based on the null hypothesis: the Omicron variant wave didn't affect the number of departing flights at Los Angeles. A fairly large p-value(0.97)was calculated, so that it failed to reject this null hypothesis. Therefore, the Omicron variant wave didn't affect the number of departing flights at Los Angeles.

Data Preprocessing

Some important factors we had to keep in mind were that the OpenSky dataset is provided as-is and both that and the Johns Hopkins data are updated on a monthly and daily basis, respectively, so incoming data must be preprocessed in the same fashion. Although both datasets contain some empty/dirty data as expected, we devised a plan to mitigate the amount of unusable data and extract the data that would be of highest importance to us. Before resorting to removing records with empty entries, we planned to impute the values of the missing data. For example, given the latitude, longitude, and altitude of origin and destination, it was possible to narrow down and fill in the missing airport codes. String values such as Country Names in the COVID and Airline Code to Country datasets were checked for spelling/abbreviations so that the mapping is smoother. Some columns ended up being removed, as they did not provide useful information, which also helped to save space in the database and add another layer of efficiency to finish preparing the data for storage in the database. Below are more in depth descriptions of how each dataset was preprocessed.

Open Sky Network Flight Data

The Data Preprocessing method for the flight data was broken into two main steps: merging the Flight and Covid data, and imputing as many missing values as possible. We have used a single file of flight data to create and test the data preprocessing method and expect this to apply to the other files with little to no additional adjustments. Using the Bansard Airline to Country Mapping data, we added two more columns - Airline Name and Country - to the Flight dataset with the help of the ICAO Designator.

Once completed, we created a list of distinct airport names and their GPS coordinates, which initially had very slightly different readings due to the nature of Open Sky Network's data collection process. To resolve this, we looked at one of two methods - order the entries in alphabetical order by airport name and take the first entry for each airport, or take the average of each coordinate reading for a single airport. The first idea, while easier to perform, may end up including dirty data and the second, while more computationally expensive, would allow for better accuracy when creating visualizations and models.

With the cleaned list of airports and their coordinates, we filled in more than half of the missing origin and destination airports. Since the records in the flight data are not dependent on each other when filling in the data, we plan to employ parallel processing to make the preprocessing more efficient.

Johns Hopkins Covid-19 Data

The Johns Hopkins Covid dataset contains a transposed format of data, with countries/regions in the first column and dates beginning on 01/22/2020 as columns holding cumulative counts of confirmed cases, recoveries, and deaths. In order to be stored in the InfluxDB database for easier querying, the time series data will need to be transposed once more so that the date columns will become rows with NULLs replaced with zeros. For EDA purposes however, the file

was loaded into a pandas dataframe with NaNs replaced with zeros. We expect any preprocessing of the Covid data to be smooth due to the consistency and regular maintenance by Johns Hopkins.

Findings to Date

We used three models to train our datasets at this stage.

- Autoregressive (AR) modeling is one of the techniques used for our time-series data analysis. AR models are a very powerful tool in time series analysis, allowing us to forecast the future based on historical data. AR models can be used to model anything that has some degree of autocorrelation which means that there is a correlation between observations at adjacent time steps.
- ARIMA modeling is another technique used for our time-series data analysis. It is actually a class of models that 'explains' a given time series based on its own past values - that is, its own lags and the lagged forecast errors, so that equation can be used to forecast future values.
- The last technique we used is LSTM, a recurrent neural network that is trained using Backpropagation Through Time and overcomes the vanishing gradient problem. We used it to create recurrent networks that in turn can be used to address difficult sequence problems. Instead of neurons, LSTM networks have memory blocks that are connected through layers. A block has components that make it smarter than a classical neuron and a memory for recent sequences. A block contains gates that manage the block's state and output. A block operates upon an input sequence and each gate within a block uses the sigmoid activation units to control whether they are triggered or not, making the change of state and addition of information flowing through the block conditional. Please refer to the details below for more information.

The following three figures are our prediction results against different models. Figure. 9 is the prediction results for the AR model. Figure. 10 is the prediction results for the ARIMA model. As we can see from these two figures, the trend of the predicted data is in line with the test data. Figure. 11 is the prediction results for the LSTM model. And the general trend has basically stayed the same, but we need to do further performance tuning.

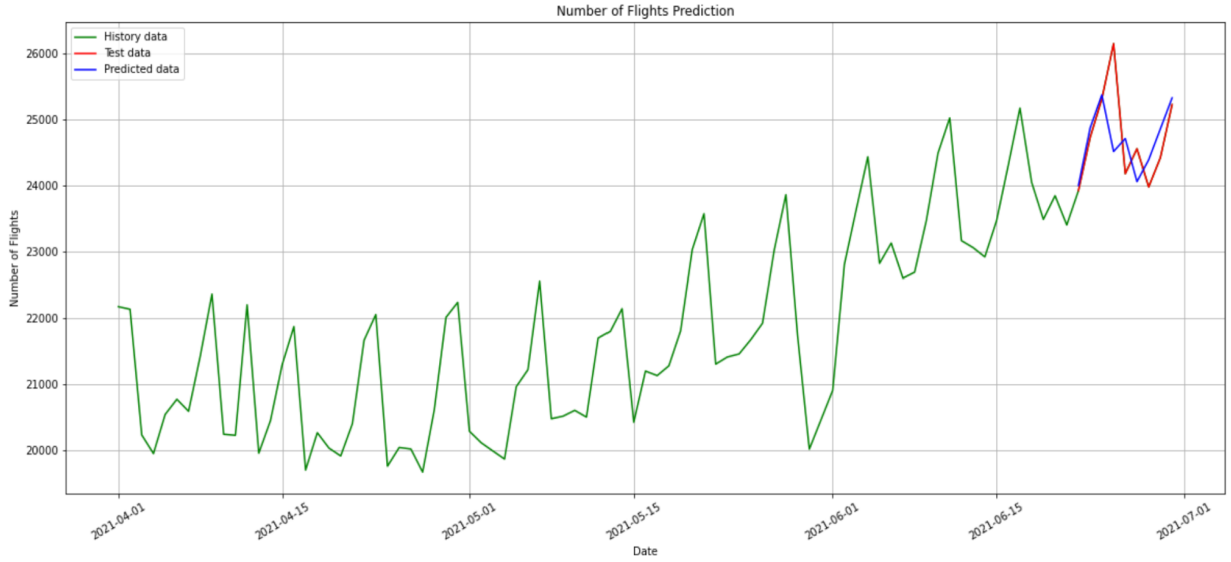


Figure. 9 Prediction Results for AR Model

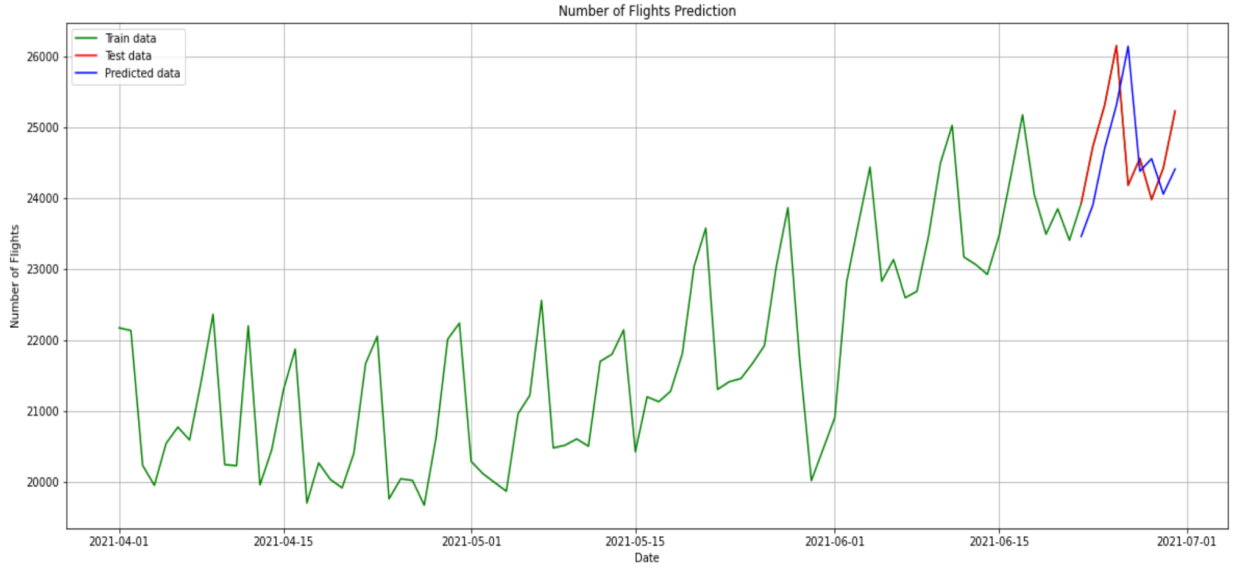


Figure. 10 Prediction Results for ARIMA Model

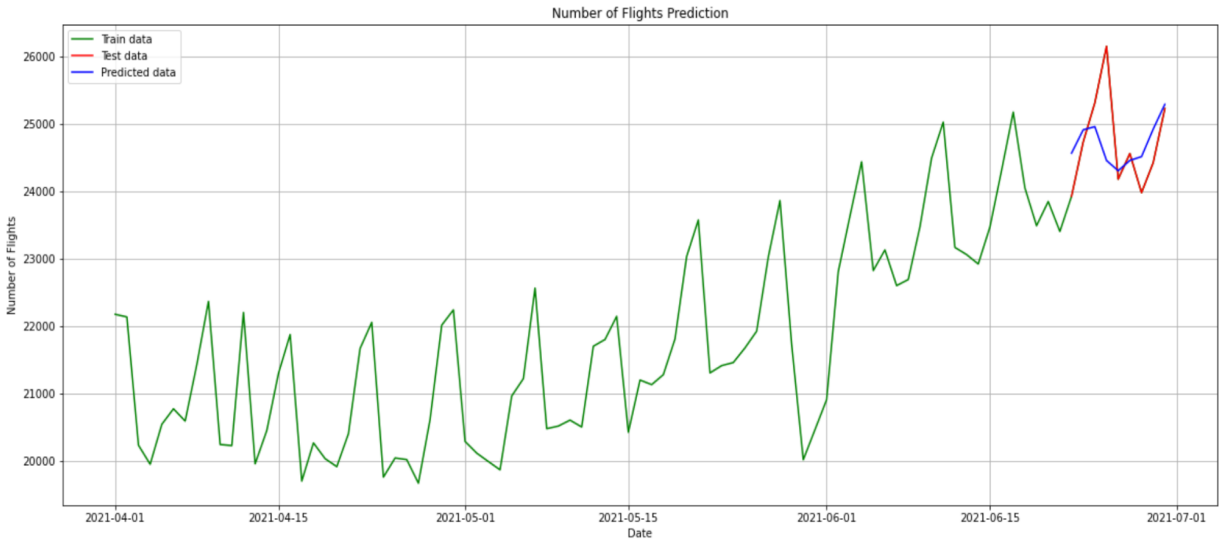


Figure. 11 Prediction Results for LSTM

Model Validation:

- For model validation, we split the data into training dataset and test dataset for all these three models(ratio 9:1). Since there are various ways of validating a model, we also plan to use other validation methods such as time series cross validation and bootstrapping in our next steps.

Techniques Used:

- We started with existing data.

We used the existing real data to learn from. In order to train the computer to understand what we want and what we don't want, we prepared, cleaned and labeled our data. We got rid of garbage entries, missing pieces of information, anything that's ambiguous or confusing. Filter our dataset down to only the information we're interested in right now. Please refer to the Combine_Flight_and_Covid.ipynb for the cleaned dataset we created.

- We analyzed data to identify patterns.

Based on our EDA results, we have chosen the right algorithms, applied them, configured them and tested them. To make the right choice, we experiment with a few algorithms and test until we find the one that gives us the results most aligned to what we're trying to achieve with our data. After that, we successfully applied a machine learning/deep learning algorithm to analyze our data and learn from it, with a trained model.

We decomposed the cleaned dataset and created autocorrelation and partial autocorrelation plots to help us identify the trends and correlations, which help us find the optimal parameters for different models. Also, we did feature scaling, built RNN, compiled RNN, and fit RNN to the training set and did the prediction for our deep learning algorithm.

- We made predictions.

The regression is supervised types of algorithms, we need to provide intentional data and direction for the computer to learn. We played around with each algorithm type and use case to better understand probability and practice splitting and training data in different ways.

Performance Graphs:

- We used ME, MSE, MAE, RMSE, and R^2 scores to report performance for different models. Please refer to table 1 for the evaluation results.

Models	ME	MSE	MAE	RMSE	R^2 score
AR	1635.20	401201.12	433.98	633.40	0.15
ARIMA	1964.06	779358.03	739.61	882.81	-0.65
LSTM	1695.72	445395.06	466.12	667.3	0.06

Table 1. Evaluation Results

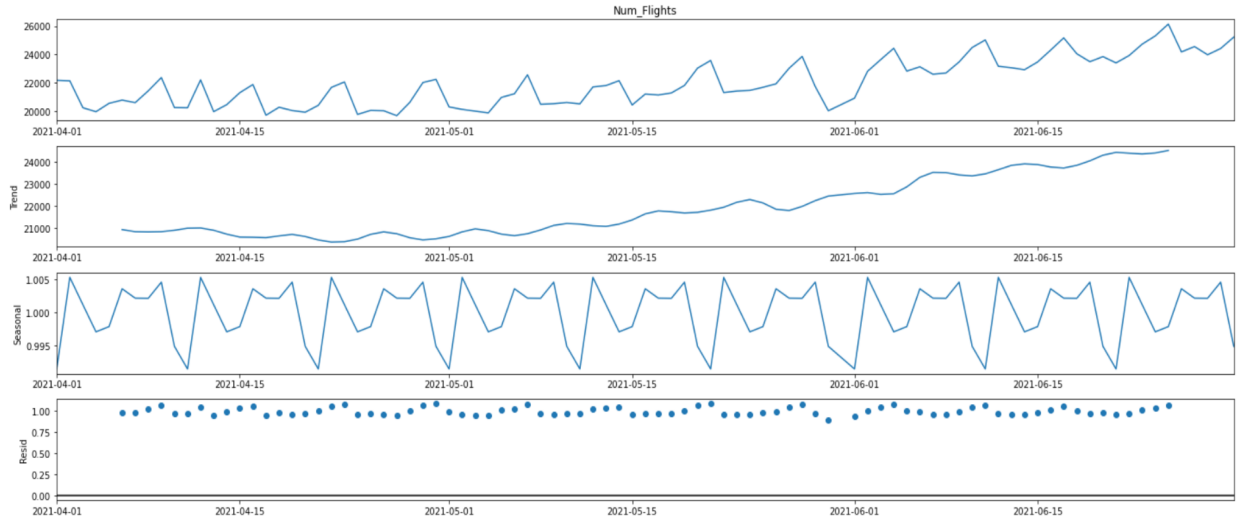


Figure. 12 Decomposed Dataset

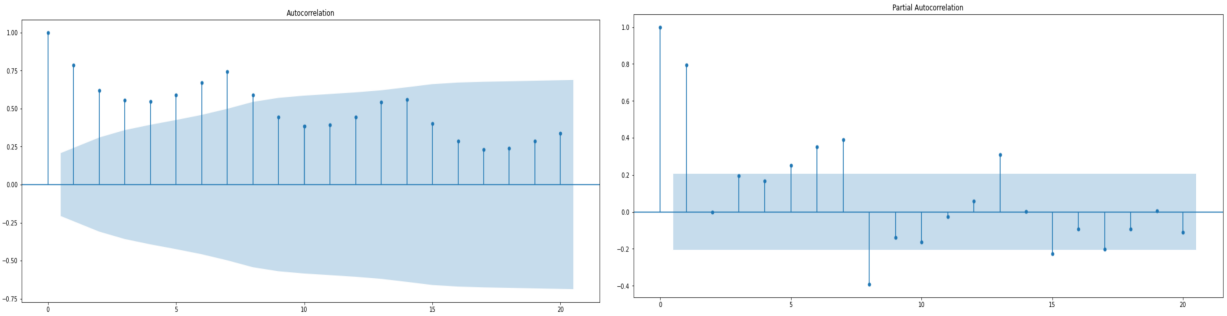


Figure. 13 Autocorrelation and Partial Autocorrelation

Insights Derived from Results:

- From Figure. 9 and Figure. 10, we can see that using AR model and ARIMA model as the baseline models can reflect the trend information and there is a trend component which grows the flight number month by month. Also there looks to be a seasonal component which has a cycle less than 2 weeks. The variance in the data keeps on increasing with time.

- From Figure. 11, we can see that using the LSTM model to train the data can reflect the trend information, however, we need to tune our performance to better improve the accuracy of the predicted data.
- From Figure. 12, we can see that the decomposed data shows that the trend and seasonality information extracted from the series does seem reasonable. The residuals are also interesting, showing periods of high variability in around 7 days of the series.
- From Figure. 13 we can see that we have a gradual decrease in the Autocorrelation plot and a sharp cut-off in the Partial Autocorrelation plot. These two plots help us find the optimal parameters.

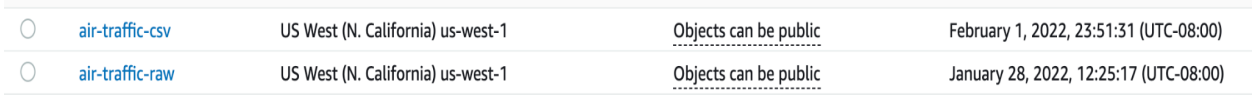
Significance of Results:

- Currently the prediction results for the AR and ARIMA models set the baseline for our project, which we can use these predictions to measure the baseline's performance and then become what we compare other machine learning and deep learning algorithms against.
- The prediction results for the LSTM model shows that our model still needs to do performance tuning.
- The reported performance for these models also reflects that we need to think about other factors such as seasonality, delay of reported covid cases that have effects on the schedule of flights, and so on.

Solution Architecture

For the processed and integrated data, we currently store it in InfluxDB, which is purpose-built for time series data. InfluxDB can store large volumes of time series data and quickly perform real-time analysis on that data.

For the raw data, we are using Amazon Simple Storage Service (Amazon S3), which is a scalable, high-speed, web-based cloud storage service to store processed and/or integrated data. As this service is designed for online backup and archiving of data and applications on Amazon Web Services (AWS), it can store and retrieve any amount of data from anywhere, which offers industry leading durability, availability, performance, security, and virtually unlimited scalability at very low costs.



○	air-traffic-csv	US West (N. California) us-west-1	Objects can be public	February 1, 2022, 23:51:31 (UTC-08:00)
○	air-traffic-raw	US West (N. California) us-west-1	Objects can be public	January 28, 2022, 12:25:17 (UTC-08:00)

- As shown above, two buckets on S3 were created to store data, 'air-traffic-raw' stores raw zipped files, 'air-traffic-csv' stores automated unzipped csv files.
- S3 lambda function to unzip files is not complete yet
- The Influxdb database has been set up on an AWS EC2 instance, to ensure data transfer speed 'gp3' was chosen and 'IOPS' was set to '3000'. This instance will serve as the main data entry point when the modeling stage starts.

- Majority of the processed data for EDA were kept locally

Initial design of data querying interface: data is processed and loaded into influxDB for access and querying. To query the data, the python API(influxdb-client-python) will be used to connect influxDB and query data via influxQL or Flux language.
We programmatically access the data. For accessing the data, we use InfluxDB to query and graph in dashboards. InfluxDB allows us to quickly see the data that we have stored via the Data Explorer UI. We can also use templates or Flux (InfluxData's functional data scripting language designed for querying and analyzing), which can rapidly build dashboards with real-time visualizations and alerting capabilities across measurements.

Additionally, current preliminary modeling was conducted from local notebooks, and further modeling steps will take additional architectural design in accordance with two considerations: one is that more data needs to be incorporated into the models and the other is that further deep learning models requires more computational resources which may have to rely on cloud computing resources such as AWS.

Future Work

- Conduct further hypothesis testing on the impact of the Delta variant, since the effects on the numbers of flights are not very apparent with Omicron.
- Create a set of network graphs to compare conditions on the same date but different years.
- Update the Heat Map to illustrate effects of Covid throughout the entire pandemic.
- For new features, currently we used the Timestamp and Number of Flights features, and we plan to add the Covid Number feature and Location related features to our project.
- For datasets, currently, we only use 3 months of data, our whole dataset contains 25 months of data and we plan to run our model against the whole dataset.
- Based on our decomposed dataset results, we can see that there is seasonality information in our data. So we plan to add seasonality information to our model.
- Based on our LSTM prediction results, we need to do further performance tuning to improve the performance.
- We also plan to try several different deep learning methods such as Seq2Seq and DCRNN.
- We will also create benchmarks based on our performance against different models.