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

Adelle Driker, Bo Yan, Yuan Hu

Advisor: Professor Rose Yu

Hypothesis Definition

With the Covid pandemic having had a strong effect on the air travel industry, 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. Throughout the EDA section, 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.

Analytic Approach for MVP

All Possible Inputs, Targets, and Models

Given that we are working with time series data, we will need to incorporate the timestamp (yyyy-mm-dd) as one of the main input features for each data point. Another key identifying feature is the Country of the airline of each flight in addition to the number of Covid cases. The target variable will be the number of flights originating from a Country on a given day. We will be experimenting with a few models to get a sense of which one would perform best on this kind of dataset.

Our two traditional time series models, which will be used as a baseline comparison, include the Autoregressive (AR) Model and ARIMA Model, and our Deep Learning model, which incorporates neural networks, is the LSTM (Long Short Term Memory) Model. Some potential models we will be additionally looking into as suggested by our Advisor are the Seq2Seq (both with GRU and LSTM) and DCRNN (Diffusion Convolutional Neural Network) models. In order to test these models, we will be using a scaled down version of the combined dataset, namely three (consecutive) months out of the 25 full months we have available.

Modeling

Models: Training and Scoring, Types of Learners, Learner Parameterization

We used the 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.

Baseline Models:

- Autoregressive (AR) Model
    - Training: AR(11), CMLE
    - Scoring: r^2 score
    - Type of learners: Supervised Learning
    - Learner Parameterization:
        - Train-test split ratio: 9:1
        - Loss function: MSE, MAE
        - The coefficients (or weights): -1241.8136
        - P value: 0.524 (p > |z|)
- ARIMA Model
    - Training: ARIMA(1, 1, 0), CSS-MLE
    - Scoring: r^2 score
    - Type of learners: Supervised Learning
    - Learner Parameterization:
        - Train-test split ratio: 9:1
        - Loss function: MSE, MAE
        - The coefficients (or weights): 25.1092
        - P value: 0.802 (p > |z|)

Deep Learning Model:

- LSTM
    - Training: sequential, LSTM
    - Scoring: RMSE score
    - Type of learners: Supervised Learning
    - Learner Parameterization:
        - Train-test split ratio: 9:1
        - Optimization algorithm: Adam optimizer
        - Drop-out rate: 0.2, 0.25
        - Bach size: 25
        - Units: 100
        - Loss function: MSE

Results and Evaluation: Model Validation, Techniques Used, Performance Graphs

The following three figures are our prediction results against different models. Fig. 1 is the prediction results for the AR model. Fig. 2 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. Fig. 3 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.

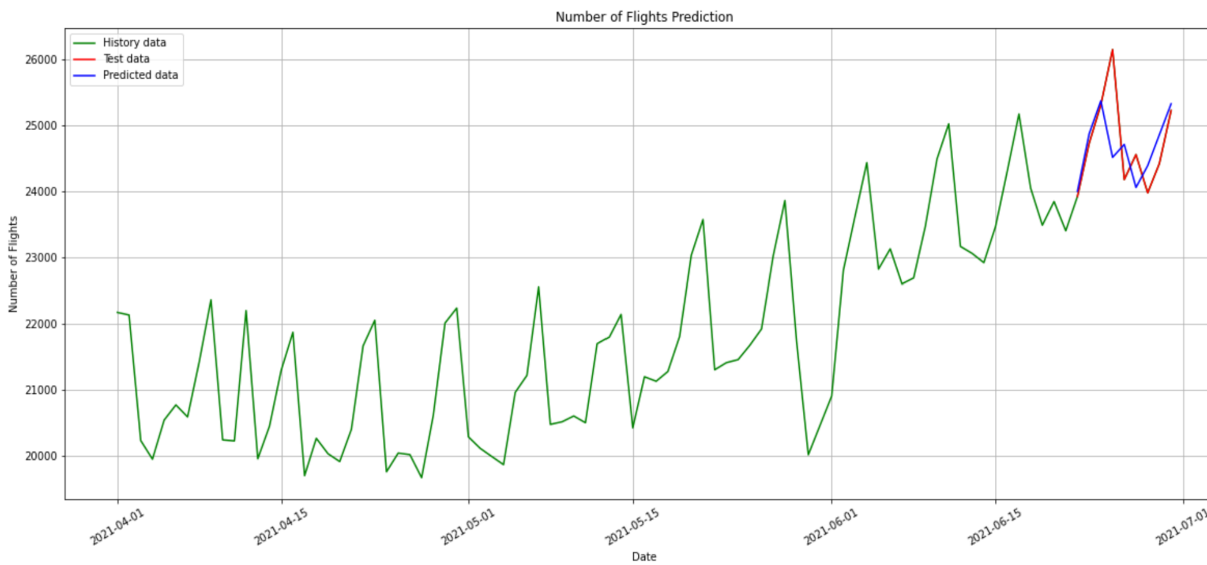

Fig. 1 Prediction Results for AR Model

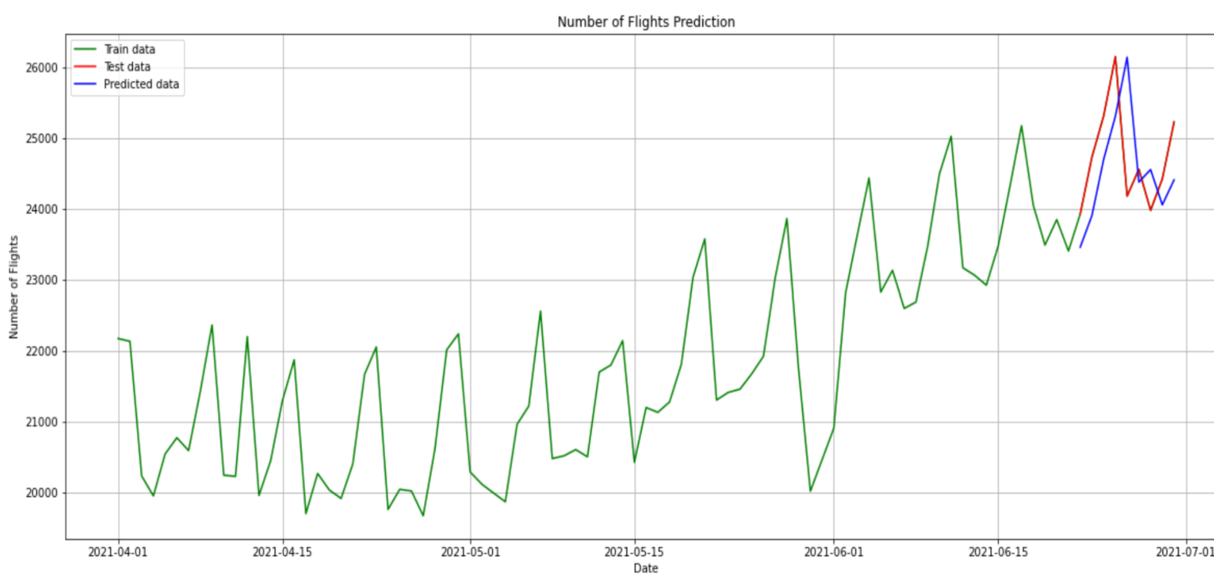

Fig. 2 Prediction Results for ARIMA Model

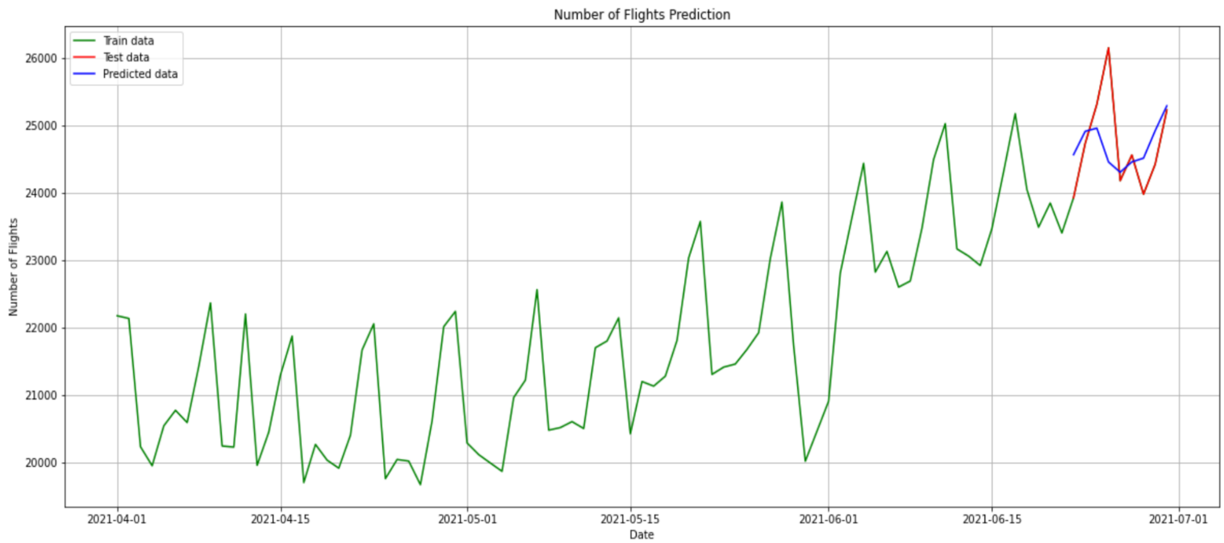

Fig. 3 Prediction Results for LSTM

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

Model Interpretation: Insights Derived from Results, Significance of Results

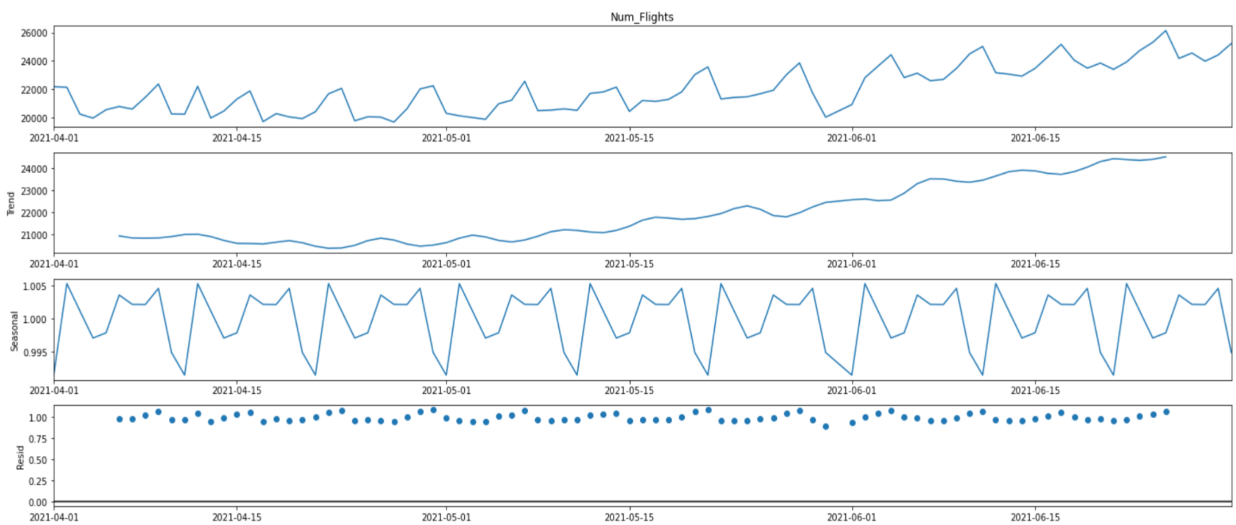

Fig. 4 Decomposed Dataset

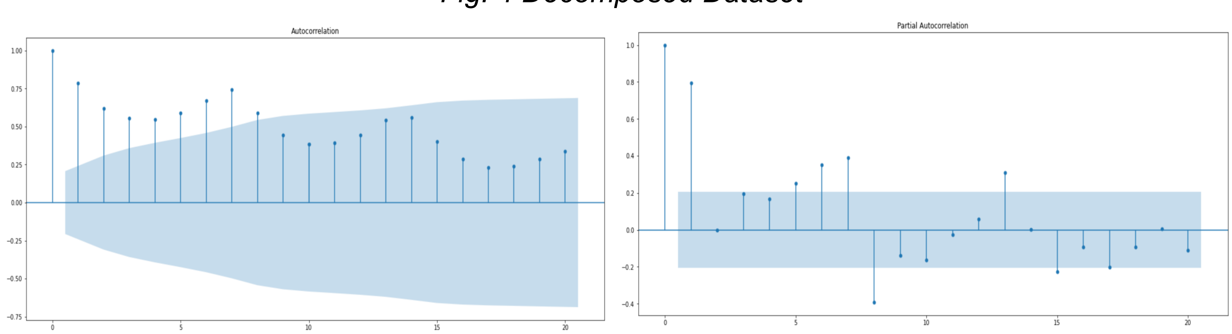

Fig. 5 Autocorrelation and Partial Autocorrelation

Insights Derived from Results:

- From Fig. 1 and Fig. 2 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 Fig. 3 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 Fig. 4 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 Fig. 5 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.

Next Steps: New Features, Datasets, Techniques Based on Results

- 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.

Team Member Contributions

Bo

- Wrote scripts for AR, ARIMA, LSTM models

- Contributed to “Models”, “Results and Evaluation”, “Model Interpretation”, and “Next Steps for Modeling”
- Maintained our Capstone Project Planning spreadsheet

Yuan

- Conducted time-series decomposition and hypothesis testing on whether the outbreak of Omicron affects the number of departing flights from Los Angeles
- Worked on finalizing data modeling for loading to influxdb database

Adelle

- Wrote script to combine flight and covid datasets
- Contributed to “Hypothesis Definition”, “Analytic Approach for MVP”, and “Updates to Steps 1, 2, and 3” sections
- Coordinated meetings between our team and advisor

Updates to Steps 1, 2, and 3

Since our first three reports, we have added another step in our data preprocessing method, specifically to combine both the flight and covid datasets. Below are the links to the new notebooks we have created:

Combine_Flight_and_Covid.ipynb
AR.ipynb
ARIMA.ipynb
LSTM.ipynb