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

Adelle Driker, Bo Yan, Yuan Hu

Advisor: Professor Rose Yu

Modeling

Analytic Approach:

The data points that are to be ingested by the model currently contain the following attributes: location (country), time stamp (day), and number of flights. The first two columns will be the inputs that the model will use to calculate the target number of flights. Please note that we will be adding more inputs as we work to improve our model's performance. Currently, we are utilizing the three models: two traditional time series models - AR and ARIMA - and a deep learning model - LSTM.

Another type of model that we have experimented with is the Sequence to Sequence (Seq2Seq) model. This model is part of a class of RNNs that is typically used to solve complex language problems like Question Answering, Text Summarization, etc. The algorithm, originally developed by google for Machine Translation, relies on an Encoder-Decoder LSTM (or GRU) model, which helps avoid the problem of the vanishing gradient. Each component is turned into a corresponding hidden vector by the Encoder, which is then decoded into an output component that is used as input in the next layer.

Model Description:

The full dataset was broken out into a 90-10 train-test split and scored with the r^2 and RMSE values, using the MSE and MAE as loss functions. Both the Deep Learning models (LSTM and Seq2Seq) and the traditional baseline models (AR, ARIMA) are considered supervised learners, and each model was parameterized as follows:

Baseline Models:

- Autoregressive (AR) Model
  - Training: AR(11), CMLE
- ARIMA Model
  - Training: ARIMA(1, 1, 0), CSS-MLE

Deep Learning Models:

- LSTM
  - Training: sequential, LSTM
  - Scoring: RMSE score
  - Learner Parameterization:
    - Optimization algorithm: Adam optimizer
    - Drop-out rate: 0.2, 0.25
    - Batch size: 25
    - Units: 100
- Seq2Seq with LSTM
  - Scoring: MAE
  - Major hyper-parameters:
    - Drop-out rate: 0.8

- Horizon: 12
- Batch size: 64
- Input/Output dimension: 1/1

Model Results:

The following three figures are our prediction results against different models. Figure. 1 is the prediction results for the AR model. Figure. 2 is the prediction results for the ARIMA model. Figure. 3 is the prediction results for the LSTM model. As we can see from these three figures, the trend of the predicted data is in line with the test data.

Results and Evaluation:

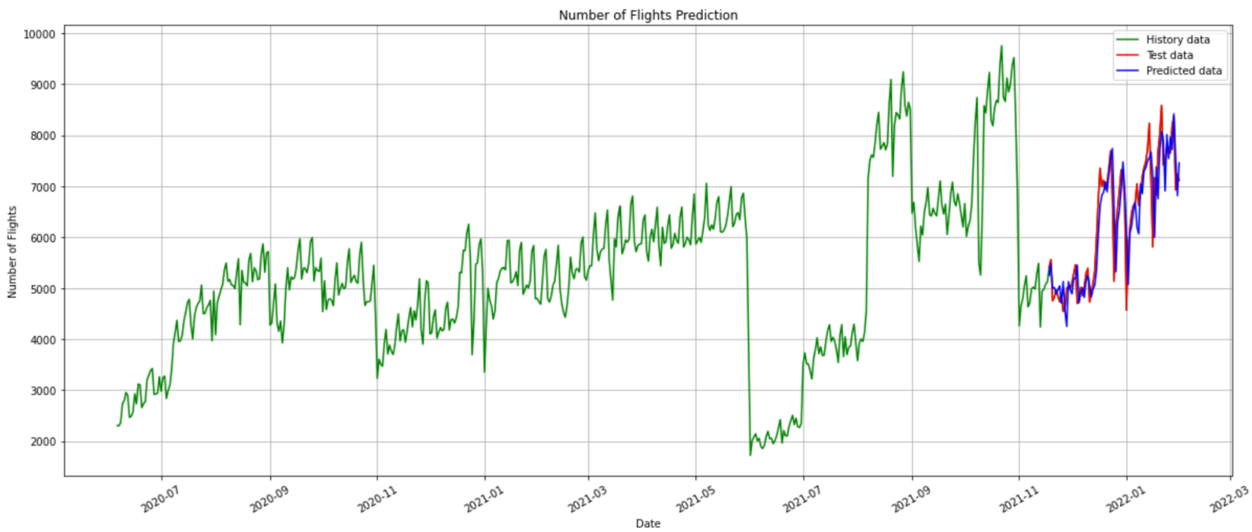

Figure. 1 Prediction Results for AR Model

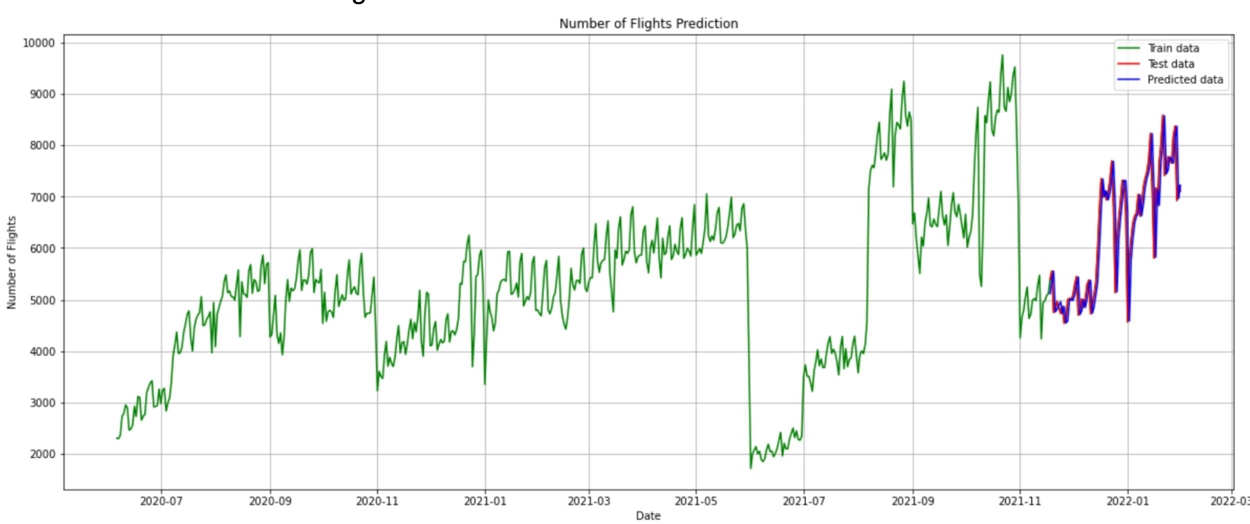

Figure. 2 Prediction Results for ARIMA Model

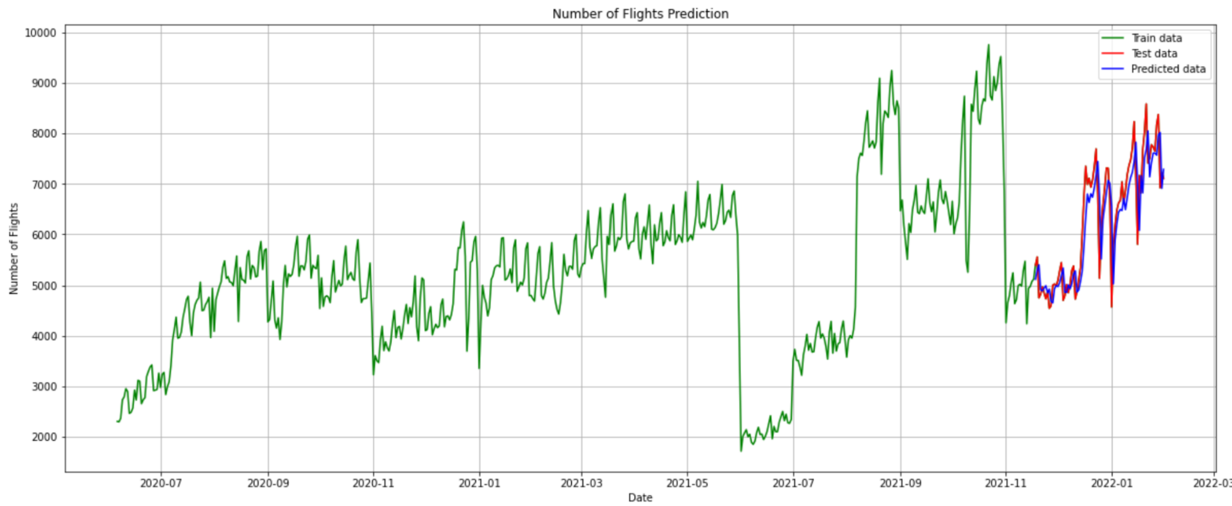

Figure. 3 Prediction Results for LSTM

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 used the existing real data to learn from. For training, 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.
- 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.
- 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	1530.10	259351.99	381.05	509.27	0.81
ARIMA	2121.74	364047.23	429.24	603.36	0.73
LSTM	1973.63	319303.64	433.86	565.07	0.76

Table 1. Evaluation Results

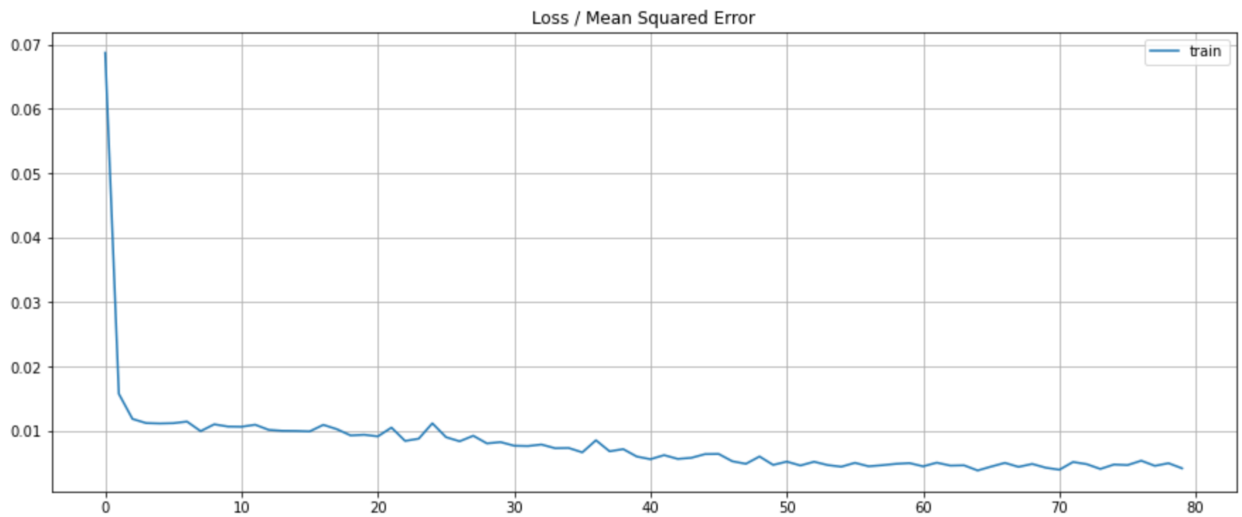

Figure. 4 MSE Loss over Training Epochs for LSTM

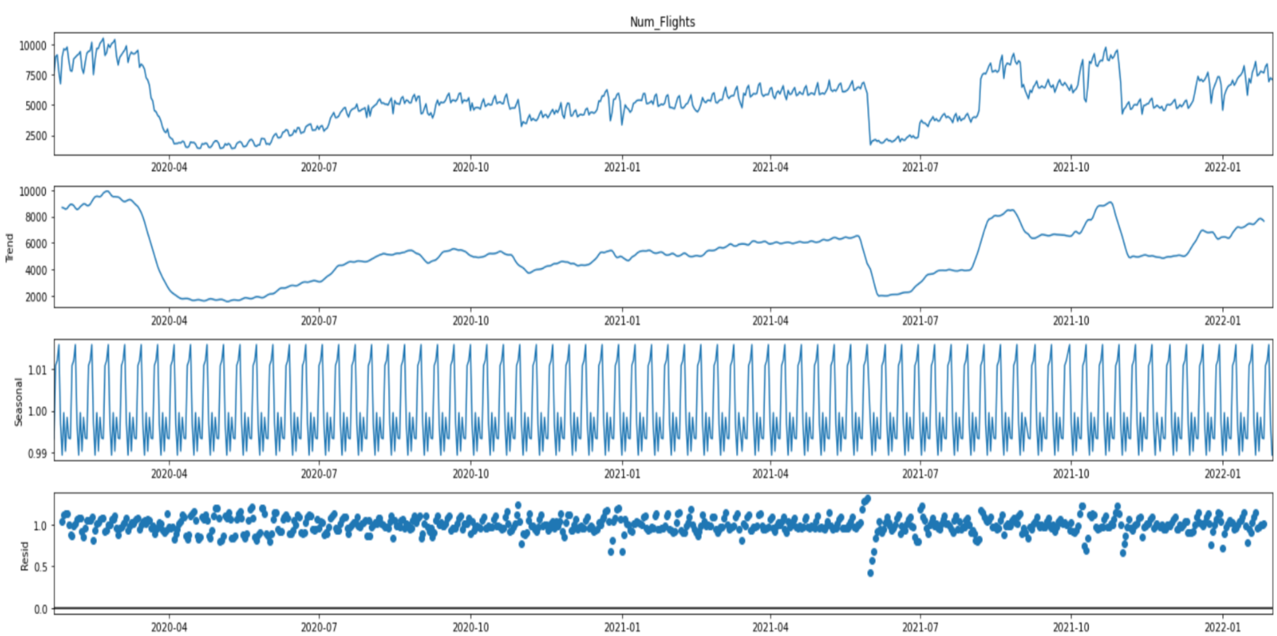

Figure. 5 Decomposed Dataset

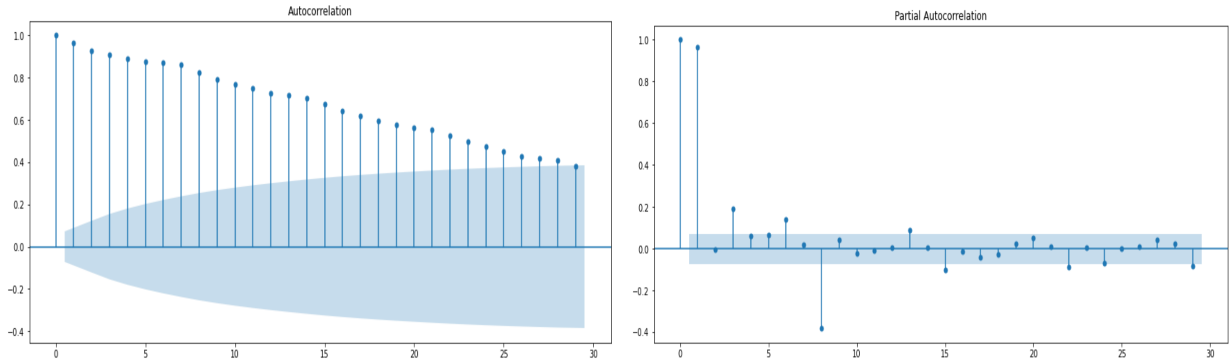

Figure. 6 Autocorrelation and Partial Autocorrelation

Model Interpretation:

Insights Derived from Results:

- From Figure. 1, Figure. 2 and Figure. 3, we can see that using AR model and ARIMA model as the baseline models can reflect the trend information. And the LSTM model performs better with training against the whole dataset compared to training the model with one quarter of the whole dataset. Also the seasonal component which has a cycle of 10 days, reflects the seasonal information. And the variance in the data keeps on increasing with time.
- From Figure. 5, the decomposed data shows that the trend and seasonality information extracted from the series does seem reasonable. The trend shows the number of flights with time. The seasonality information shows a 10 days cycle.
- From Figure. 6 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.
- From Table. 1, the R^2 scores show the predicted data is strongly correlated.
- From Figure. 4, we can see that the best epoch for LSTM is 60.

Significance of Results:

- Currently the prediction results for the AR and ARIMA models set the baseline for our project, which are used 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 training with the whole dataset produces better performance.
- Our deep learning models still need 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.

Conclusion and Discussions for Next Steps:

- Having trained our model on the full set of data, we can already see a definite improvement in the prediction results using the traditional time series models (AR, ARIMA) without having added any additional features so far. The LSTM DL model performed sufficiently well, but some improvements can still be made. Compared to our previous trial with the partial dataset which contained a single quarter of data, the r^2 value for the LSTM increased from 0.06 to 0.77 (which is close to the best value of 0.80 generated by the AR model) - a significant improvement.
- It appears that all models show the most deviation between actual and predicted results around December 2021 - January 2022, which was the peak of the Omicron wave. To help reduce this error, we are devising a set of additional features that we will incorporate into our model - most importantly, the Covid counts. These counts are already incorporated into our preprocessed dataset which feed into our model, and we are ignoring that column at the moment to test the performance of the model with just the flight data. An additional feature that we think would benefit the model's performance would be a holiday schedule, which will certainly have an effect on people's travel schedules.
- Further updates to the dataset include modifying the location column to include an airport name along with its latitude and longitude coordinates so that a more granular analysis may be conducted.

Visualization I

- Figure. 7 shows the prediction results for AR, ARIMA, and LSTM models against whole datasets.
- Notebooks: AR.ipynb, ARIMA.ipynb, LSTM.ipynb, Visualization.ipynb

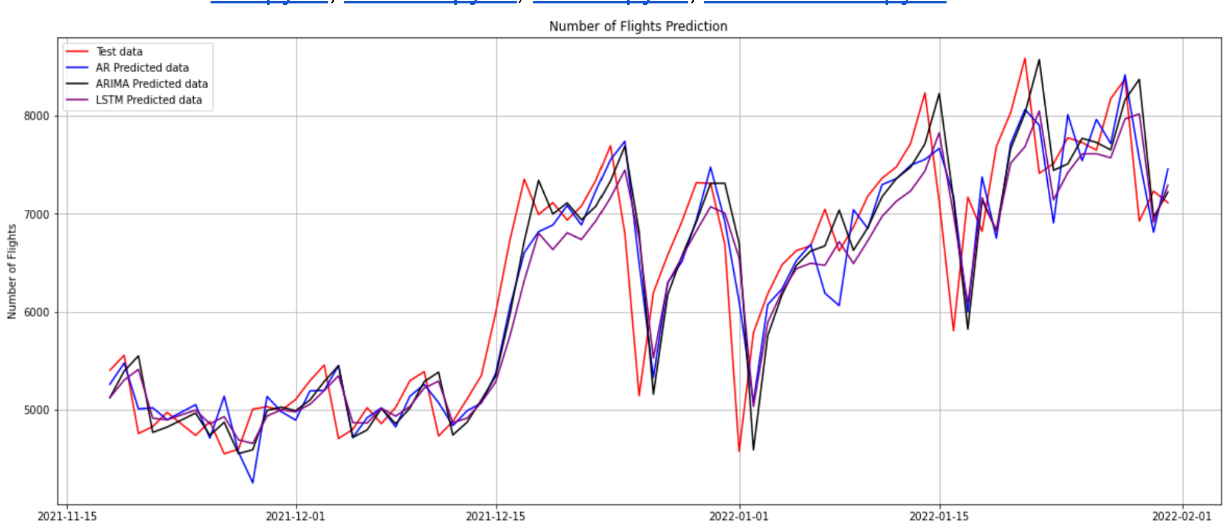

Figure. 7 Prediction Results for AR, ARIMA, and LSTM

Visualization II

Notebook:[seq2seq.ipynb]

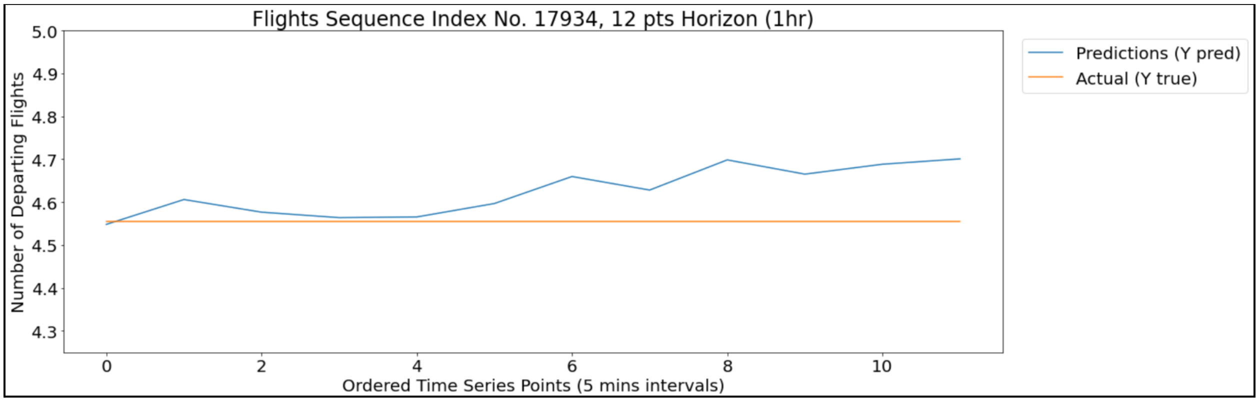

Figure. 8 Prediction Results for Single index, Seq2seq Model

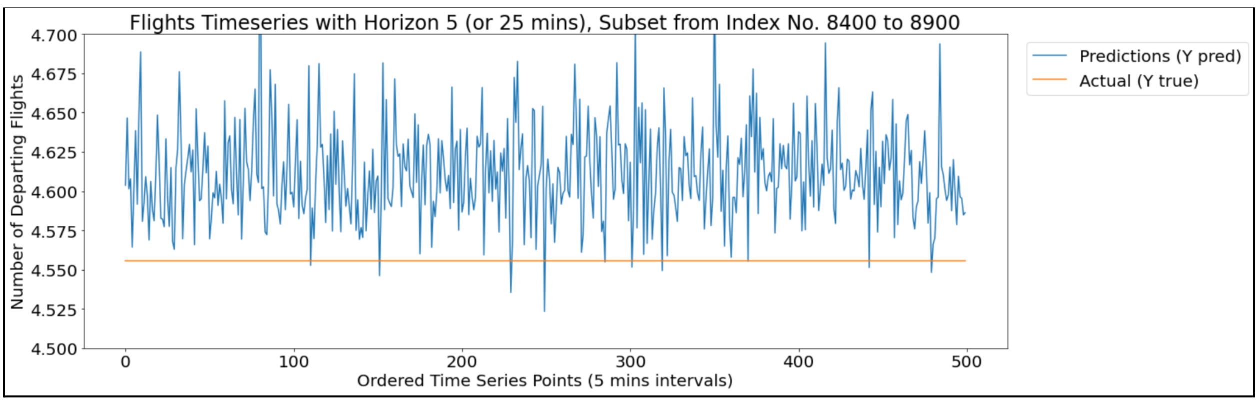

Figure. 9 Prediction Results for Sequenced index, Seq2seq Model

Team Member Contributions

Bo:

- Wrote scripts for AR, ARIMA, LSTM, and Seq2Seq models and performed debugging and performance tuning. Wrote script for Visualization.
- Contributed to “Modeling” and “Visualization I”
- Maintained our Capstone Project Planning spreadsheet and GitHub

Yuan:

- Debugged script for Seq2Seq model
- Contributed to “Visualization II”
- Set up Sagemaker for DL notebooks

Adelle:

- Reached out to advisor to coordinate new meeting schedule for the quarter
- Consolidated code to batch process all monthly flight files, combine datasets, and transform into the format required for the time series models (both DL and traditional) to use
- Contributed to “Modeling” and “Major Updates” sections

Major Updates to Steps 1-5:

Since the last report, we have been able to preprocess the full dataset ranging from late January 2020 through February 2022. Since the process of filling in missing airport names took 2 hours total per file, we have excluded those kinds of rows altogether, retaining over 90% of the original dataset. With this update, we still have plenty of records to incorporate into our deep learning model.