ISE537 Final Project
Price Prediction: Comparison between ARIMA Model and LSTM

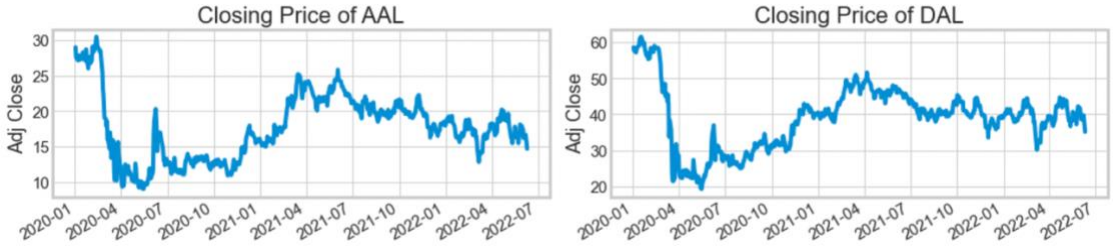
Yu Yun Tsai

Abstract

In this project, we do the price prediction of the two stocks utilizing both ARIMA model and LSTM model. The goal is to compare the out-of-sample performance of the two stocks during the pandemic period between ARIMA model and LSTM model and investigate the performance of two stocks we chose. We find that both LSTM and ARIMA model performs well on the datasets.

Data

We collected the historical financial data of stock price from Yahoo Finance during the period of 2020-01-01 to 2022-06-12, which refers to the COVID19 pandemic period. I set the start date to '2022-06-01' since around that date the pandemic began to prevail in the world, almost all the counties have restrictions of traveling abroad, this restriction significantly impacted the aviation industry, most flights were cancelled. Flights were cancelled then since there were only a few passengers. And I set the end date to '2022-06-12' since this is the date the US government announced that the vaccination record is no longer required to enter United States, people start to travel abroad and thus airlines started to add more flights. To understand the impact of aviation industry from the COVID19 pandemic period, I choose two Airlines stocks 'American Airlines' and 'Delta Airlines' to do the prediction. The data format we download from yahoofinance is data frame, and we transfer the 'Close' price to numpy.array to do the predictions below, since tensorflow accept the nnumpy.array as its data format to train the model. After utilizing the function aal.info to check the dataset, we find that there is no missing data during this period in both stocks AAL and DAL.



The plot shows the trend of closing price of the two stocks during the COVID19 pandemic period.

Introduction to LSTM model

LSTM model refers to long short-term memory networks, which is a variety of recurrent neural networks (RNNs). LSTM is a network with loop to make it capable to learn long-term dependence in sequence prediction problems, which is different from RNN since RNN can predict more accurate based on current data. LSTM has feedback connection so that it can processing entire sequence of data. LSTM is composed of a sigmoid neural net layer and a cell state, an input gate, an output gate, and a forget gate, and the flow of information can be removed

or added from the cell state and regulated by gates. We use LSTM to predict stock price since the prediction of a future stock price is related to the previous stock price.

Data processing

- We use the close price of the stock AAL and DAL to do the LSTM model.
- We split the train-test dataset. After extracting the 95% of dataset as the training set, and 5% of dataset as the testing set, we can calculate the length of training set is 586.
- Utilizing the function MinMaxScaler from scikit-learn to normalize the data to interval [0,1]. Then we extract the training and testing dataset from the dataset after normalized.
- I pick a lookback window with 60 timestamps. Create 60 timestamps of data as x_train, and the following 60 timestamps as y_train. Next, we reshape the dataset x_trian and y_train into a three-dimensional array to train the LSTM model later.
- Repeat the steps with the testing dataset to obtain x_test and y_test.
- Do the same process above for the two stocks AAL and DAL.

LSTM model choice (layer)

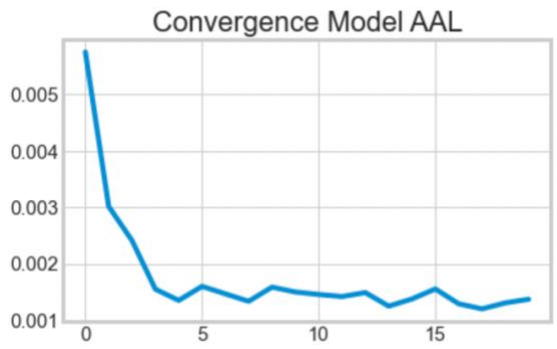
We first define a sequential model with layers. Then we add a layer to the sequential model using model.add. We add 128 units of nodes to the input layer and set the return_sequence to true to return the full sequence as input, people often choose a higher number of units for this layer, thus I pick 128 units from the common value (32,64,128,256). The parameters of input_shape is the number of timestamps and the number of input unit. Next, we add another layer to the model with 64 units of node and set the return_sequence to false to return only the last output in the output sequence. Then we add a densely connected network layer with 25 units. Lastly, we add a dense layer and set to 1 unit to make the model more robust.

Fit train model

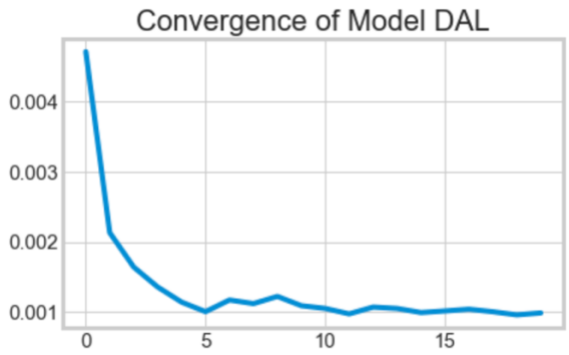
Before fitting the LSTM model with the training set, we compile the model by setting the ‘adam’ optimizer to optimize the algorithm and set the ‘mean squared error’ as loss function for our model to reduce the loss. Then we can train the model with the training dataset for 20 epochs and in batches of 1 sample.

Convergence of LSTM model

Stock AAL Stock DAL



According to the plot above, we use loss (mean squared error) to draw the plot, and we can see that the LSTM model of stock AAL converges.



According to the plot above, we use loss (mean squared error) to draw the plot, and we can see that the LSTM model of stock DAL converges.

Prediction

After training the model, we use the model to predict the price with training dataset and testing dataset. The predictions fall in the interval [0,1] since we do the Minmaxscaler to scale the data in the beginning, we need to do inverse transform to denormalized the predicted price.

LSTM model evaluation

To evaluate the performance of the LSTM model, we apply the root mean square error (RMSE) to the trained model. For stock AAL, we get the RMSE of testing set = 0.7322, and the RMSE of the training set = 0.7366, both RMSE is low, which means that the performance is satisfactory. For stock DAL, we get the RMSE of testing set = 1.6246, and the RMSE of the training set = 1.7858, both RMSE is low, which means that the performance is satisfactory.

Evidence of over-fitting

- RMSE of stock AAL (training) = 1.0505
- RMSE of stock AAL (testing) = 0.7811

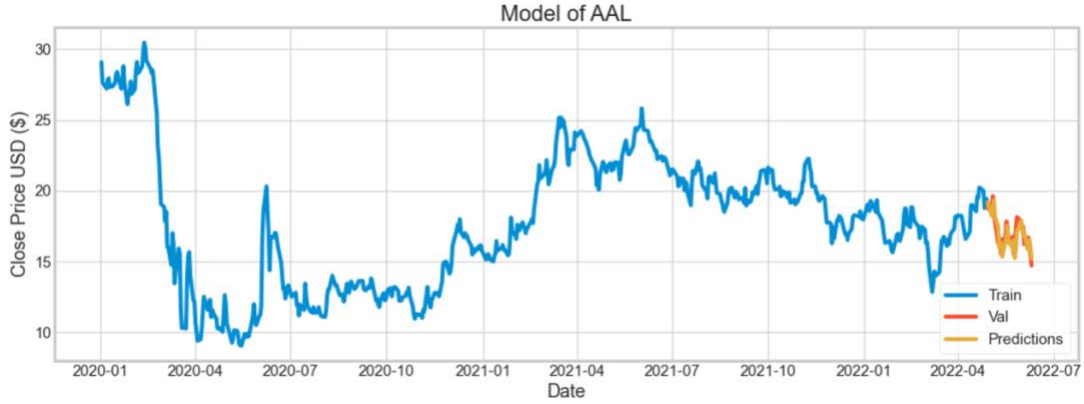
There is no evidence of over-fitting. RMSE of training dataset and RMSE of testing dataset perform well. Although the RMSE of training dataset of stock AAL is larger than the RMSE of the testing dataset, the gap is small.

- RMSE of stock DAL (training) = 1.2139
- RMSE of stock DAL (testing) = 1.5637

There is no evidence of over-fitting. RMSE of training dataset and RMSE of testing dataset perform well.

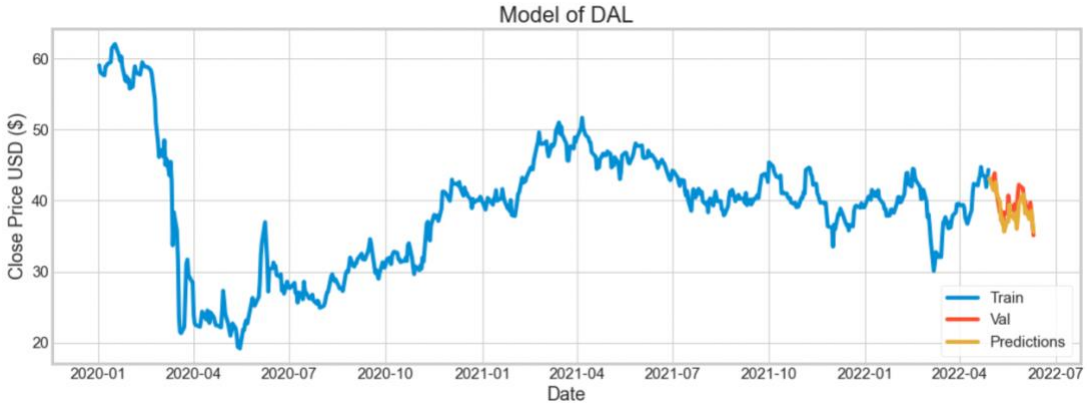
Result of LSTM model

- Stock ‘AAL’ (American Airlines)



From the plot above, we can see that the predicted price overall follows the trend of the true price. The model performs well with the dataset.

- Stock ‘DAL’ (Delta Airlines)



From the plot above, we can see that the predicted price overall follows the trend of the true price. The model performs well with the dataset.

Financial Interpretation

By plotting the time series data, we observe the prices were fluctuated between 2020/01-2021/06 the COVID 19 pandemic. The prices dropped severely when the pandemic prevailed all over the world and in slower growth after the travel restrictions were lifted in United States since airlines started to add more flights to fit those passengers. The LSTM model works well on the prediction during the COVID pandemic.

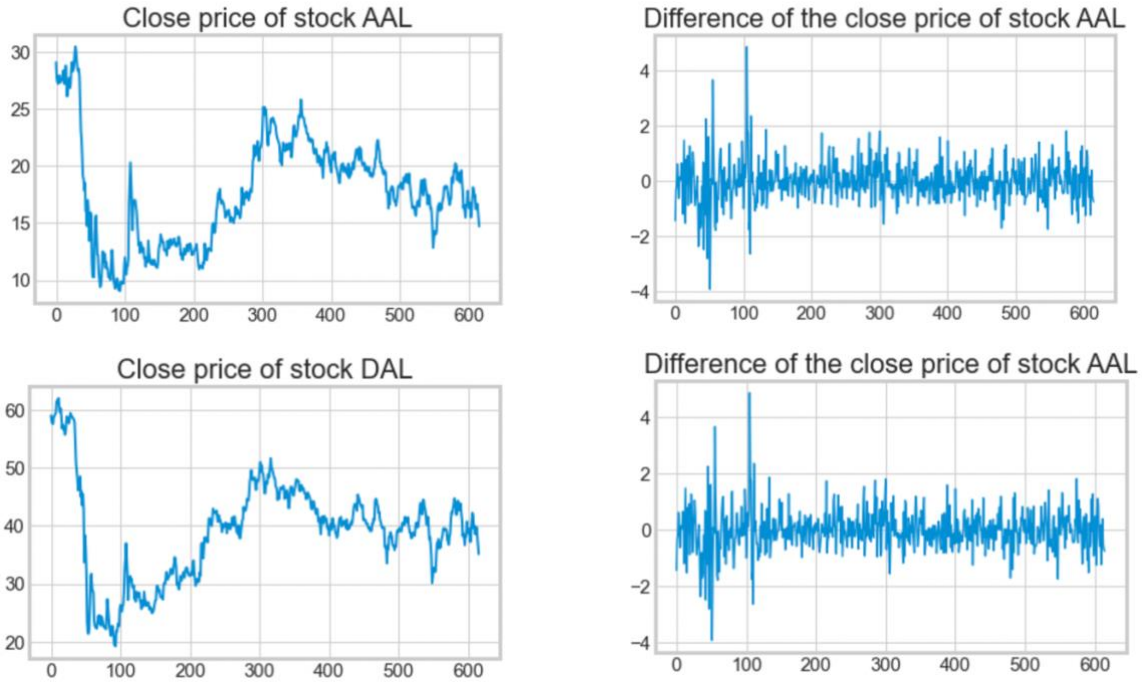
Introduction to ARIMA model

ARIMA is one of the classic time series forecasting models. ARIMA model refers to Autoregressive integrated moving average model, which is based on the autoregressive moving average (ARMA) model fitted on d-th order differenced time series that the final differenced

time series is stationary. Auto Regressive (AR) regression model indicates that the variable depends on the lags of previous value. 'I' in the ARIMA model refers to integrated (non-stationary differencing), which performs the prediction on the difference between the values and the previous values. The 'MA' in the ARIMA model indicates moving average involves modeling the error terms as a linear combinations of error terms that occurs contemporaneously and at various times in the past, which means that the model predict future values based on the previous forecasting errors. ARIMA models are denoted ARIMA(p,d,q), p indicates the order (number of time lags) of AR model, d indicates the order of difference of the original dataset (the number of times the data have had previous values), and q refers to the order of the MA model.

Data processing

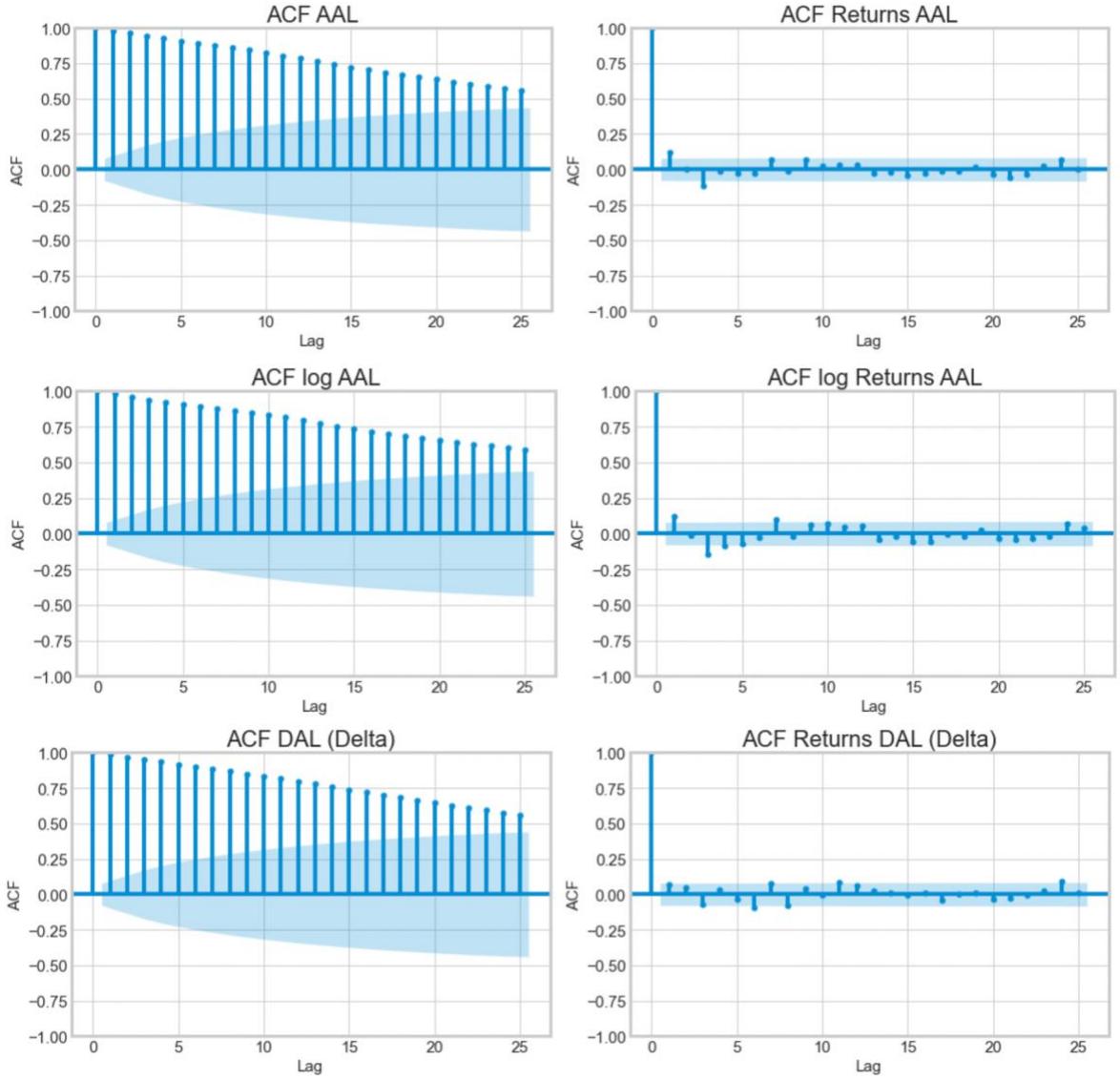
- We use the close data to do the ARIMA model.
- We split the train-test dataset. After extracting the 95% of dataset as the training set, and 5% of dataset as the testing set, we can calculate the length of training set is 586.
- We find the difference of the close price, the log of close price, and the difference of the log close price to do the stationarity test (ADF test) seperately.

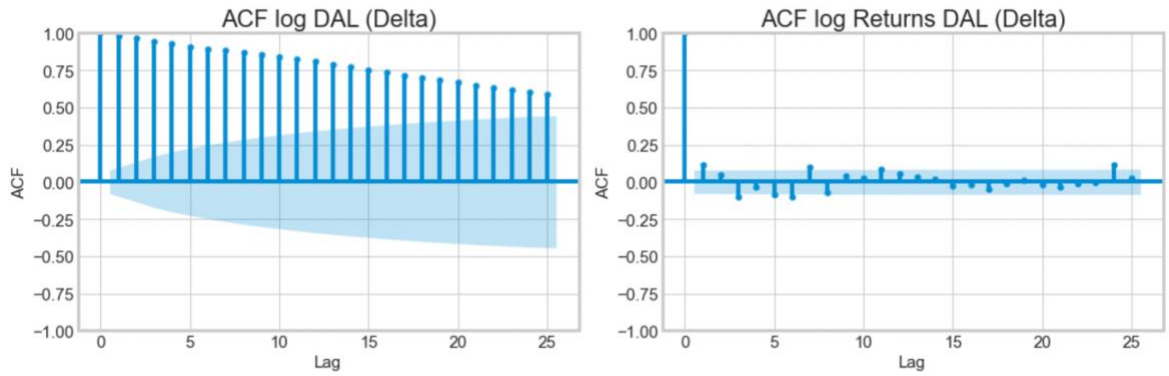


Stationarity (ADF test)

ADF test refers to Augmented Dicky-Fuller test, which used to test the null hypothesis that a unit root is present in a time series. From the test statistic and p-value, we can determine whether the time series is stationary or not. At the level of confidence, the existence of a unit root means the time series is non-stationary and does have time-dependent structure. Furthermore, the more negative it is, the larger power of the rejection of hypothesis that the existence of a unit root is.

Since p-values of stock AAL (the difference of close price and the difference of the log close price) and stock DAL (the difference of close price and the difference of the log close price) equal to 0.000000, which is smaller than 0.05 (5% significance level or 95% confidence interval), which means that the data does not have a unit root and can reject the null hypothesis, thus it is stationary. Therefore, we choose the difference of close price and the difference of the log close price to do the ARIMA model because both are stationary.





ARIMA Model Choice:

To find the optimal ARIMA(p,d,q) model to fit the data, we would like to select the model that has a lower AIC. First we determine the d-value of ARIMA(p,d,q), which refers to the number of differences that required to make a given time series stationary. Utilizing ndiffs to the data to estimate the order of differences of the dataset, and we result of both stocks are d-vlaue=0. Next, we use the auto.arima and set the maximum value of p and q to 10 to find out the optimal ARIMA(p,d,q) model. After analysis, we can find out that ARIMA(4,0,5) would be the optimal model for both the difference of the close price and the difference of the log close price of stock AAL. ARIMA(2,0,2) would be the optimal model for the difference of the close price of stock DAL, and ARIMA(0,0,8) would be the optimal model for the difference of the log close price of stock DAL.

Prediction

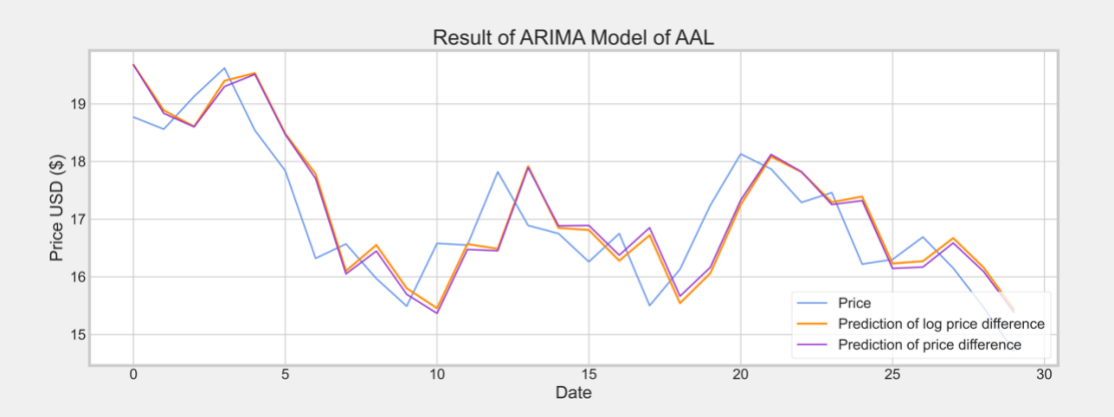
According to the stationarity test above, we will do the two predictions for each stock, we will use the price difference and the log price difference to predict.
If we respect the observations from AIC, the best model choice for the price difference of stock AAL is ARIMA(4,0,5). After we select our optimal model, we start to fit the ARIMA model using the training dataset and the optimal order to fit the data. Next, we predict the data with the model. Since we use the price difference to do the prediction, the prediction value is also based on the difference. We need to reverse the difference value back to the price to do the comparison. We create a list to append the value we calculate and use for loop to do the calculation. We add each of the prediction of price difference to each day of the true closing price (except for the first day) and append to the list. This list would be the prediction of the close price we predict from LSTM model. We also do the prediction using the log price difference, thus we need to reverse the predicted value back to the non-log full price to do the comparison. We first reverse the predicted log price difference back to the log full price by creating a list to append the value we calculate and use for loop to do the calculation. We convert the close price to log close price and add each of the prediction of price difference to each day of the log closing price (except for the first day) and append to the list. Then we reverse the list back to the non-log full price by adding the list into exponential (np.exp()).
We repeat the steps above for stock DAL to do the comparison.

Financial Interpretation

By plotting the time series data, we observe the prices were fluctuated between 2020/01-2021/06 the COVID 19 pandemic. The prices dropped severely when the pandemic prevailed all over the world and in slower growth after the travel restrictions were lifted in United States since airlines started to add more flights to fit those passengers. The ARIMA model works well on the prediction during the COVID pandemic.

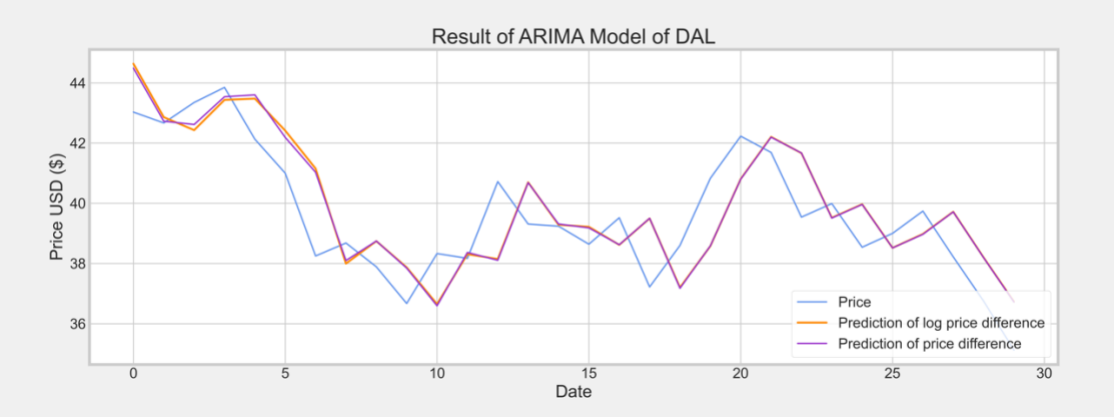
Result of ARIMA model

- Stock AAL



From the plot above, we can see that the predicted log price difference and predicted price difference overall follows the trend of the true price. The two predicted value are almost overlap, the difference is very small. The model performs well with the dataset.

- Stock DAL



From the plot above, we can see that the predicted log price difference and predicted price difference overall follows the trend of the true price. Also, the two predictions overlap in the end. The model performs well with the dataset.

ARIMA model evaluation

We can obtain the model predicted price values and evaluate the trained model based on the testing dataset and use the inverse_transform function to denormalized the predicted price. To

evaluate the performance of the LSTM model, we apply the root mean square error (RMSE) to the trained model. For stock AAL, we get the RMSE of testing set = 0.7322, and the RMSE of the training set = 0.7366, both RMSE is low, which means that the performance is satisfactory. For stock DAL, we get the RMSE of testing set = 1.6246, and the RMSE of the training set = 1.7858, both RMSE is low, which means that the performance is satisfactory.

Comparison

For stock AAL, we can compare the two predictions of ARIMA model (prediction of price difference and prediction of log price difference) and find an optimal one to do the comparison with LSTM. RMSE of price difference = 0.7475 and RMSE of log price difference = 0.7588. Although both RMSE is low, we still can find out that root mean square error (RMSE) of the price difference is lower, which means it performs better. We also calculate out RMSE of LSTM model = 0.7811. In this case, performances of both LSTM model and ARIMA model are well.

For stock DAL, we can compare the two predictions of ARIMA model (prediction of price difference and prediction of log price difference) and find an optimal one to do the comparison with LSTM. RMSE of price difference = 1.3962 and RMSE of log price difference = 1.4152. Although both RMSE is low, we still can find out that root mean square error (RMSE) of the price difference is lower, which means it performs better. We also calculate out RMSE of LSTM model = 1.5637. In this case, performances of both LSTM model and ARIMA model are well.

Advantages and Disadvantages

LSTM model
Advantages:

- Solve vanishing gradients problem.
- Can predict future values based on prior sequential data.

Disadvantages:

- Require many resources and it is time consuming to get trained.
- LSTM is prone to overfitting.

ARIMA model:
Advantages:

- It works well for short term predictions.
- Requires only previous data of time series to do the prediction.

Disadvantages:

- High cost to implement.
- Cannot works well at predicting series with turning points.

Conclusion

We predict the stock price during the COVID19 pandemic period (2020-01-01 to 2022-06-12) by LSTM model and ARIMA model. LSTM can predict future prices based on previous data, and the result of LSTM overall follows the trend of the true price, thus it suitable for predicting stock price. As for ARIMA model, we do the ADF test to check the stationarity and choose the

stationary data to fit the model and do the prediction. From the results we can find that the model performs well. Prediction of price difference and prediction of log price difference overall follows the trend of the true prices. To summarize, RMSE of LSTM model and ARIMA model are low, thus for this kind of airlines stock price, ARIMA model and LSTM model are appropriate to work and perform well.

Reference

https://databasecamp.de/en/ml/lstms
https://colah.github.io/posts/2015-08-Understanding-LSTMs/
https://medium.com/the-handbook-of-coding-in-finance/stock-prices-prediction-using-long-short-term-memory-lstm-model-in-python-734dd1ed6827
https://en.wikipedia.org/wiki/Long_short-term_memory
https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average
https://medium.com/analytics-vidhya/a-thorough-introduction-to-arima-models-987a24e9ff71
https://en.wikipedia.org/wiki/Augmented_Dickey%E2%80%93Fuller_test
https://en.wikipedia.org/wiki/Autoregressive%E2%80%93moving-average_model
https://towardsdatascience.com/introduction-to-arima-for-time-series-forecasting-ee0bc285807a#:~:text=Auto%20Regressive%20Integrated%20Moving%20Average,to%20a%20time%20series%20data.
https://www.capitalone.com/tech/machine-learning/understanding-arima-models/