

Time-Series Forecasting for NASDAQ Index

ANLY 560 Final Project

Group members:

Yingjie(Chelsea) Wang, Jiaqi Hu, Ju Huang,

Qiaowei Jiang, Xinyue (Ivy) Liu

Table of Contents

[INTRODUCTION 3](#_Toc39098804)

[Basis 3](#_Toc39098805)

[Background 3](#_Toc39098806)

[Goal and Motivation 3](#_Toc39098807)

[History 3](#_Toc39098808)

[State-of-art 3](#_Toc39098809)

[THE ANALYSIS/STATISTICAL METHODS 4](#_Toc39098810)

[DATA INTRODUCTION & DATA PREPARE 4](#_Toc39098811)

[WHY EACH METHOD / MODEL? 4](#_Toc39098812)

[ARIMA 5](#_Toc39098813)

[ARIMA + Regressor 6](#_Toc39098814)

[GARCH 6](#_Toc39098815)

[LSTM 8](#_Toc39098816)

[Intro to LSTM 8](#_Toc39098817)

[Feature Engineering 9](#_Toc39098818)

[RESULTS 10](#_Toc39098819)

[ARIMA 10](#_Toc39098820)

[ARIMA + Regressor 12](#_Toc39098821)

[GARCH 14](#_Toc39098822)

[LSTM, Deep Learning 15](#_Toc39098823)

[Result Compare Table 16](#_Toc39098824)

[CONCLUSION 17](#_Toc39098825)

[Appendix 18](#_Toc39098826)

[NASDAQ.csv: 18](#_Toc39098827)

[EDA\_ARIMA\_Xreg\_GARCH.html: 18](#_Toc39098828)

[LSTM.html 18](#_Toc39098829)

[NASDAQ\_Presentation.pptx: 18](#_Toc39098830)

[References 18](#_Toc39098831)

# INTRODUCTION

## Basis

The stock market is a market that facilitates a unified trade of buying and selling of company stocks. Every stock exchange has its own index value, which is an average value that is calculated by combining several stocks. The index acts as a representation of the entire stock market and predicting the market’s trend over time. The stock market can have a huge impact on people and a country’s economy as a whole. Therefore, predicting the stock patterns and trends in an efficient manner can minimize the risk of loss and maximize profit during an exchange.

The values of stock prices are not randomly generated and can be thought as a discrete-time series model which is based on a set of numerical data elements gathered at continuous points during regular intervals of time. In this project, we are going to help our audiences better understand and forecast the NASDAQ Composite Index with time series analysis. Interest rate is an important economic variable that interacts with the stock market but sometimes not considered in stock market forecast models. The models that were used for this project will detail how critical the interest rate variable can be for better performance.

## Background

The NASDAQ Composite Index is one of the most widely-watched indexes in the world, and can be seen as a stand-in for the technology sector, due to its heavy weighting in technology. The NASDAQ Composite Index was also originally launched in 1971, with only a starting value of 100 (Investopedia, 2020).

## Goal and Motivation

There are three major goals of this project. First, was to identify find the best model to predict the NASDAQ Composite Index and provide new perspective ideas to data scientists. Second, the study could be used to help investors compare the current price level from the previous and include an overall evaluation of the market performance. Third, this project can also assist forecasting practitioners to predict the technical industry’s trend of development.

## History

In the past, researchers focused the stock market on the S&P 500 index, which is another stock index that is considered by economists to be more comprehensive and stable. However, these studies only considered the index itself. Compared to this analysis, that focused on the technology domain and additionally added interest rate in the models to improve the performance results.

## State-of-art

At this time, the economy and financial market remain to be unstable, since the Covid-19 virus sweeping the world. A potential risk that may result from this analysis would be poor forecasts for 2020 compared to forecasting 2019 index values. Despite this concern, the models could learn from the variation of the NASDAQ Composite Index and eventually increase the overall accuracy.

# THE ANALYSIS/STATISTICAL METHODS

## DATA INTRODUCTION & DATA PREPARE

The data used in this analysis is comprised of ten years’ NASDAQ Indices from 2010 to 2019 (Data source from [Yahoo Finance](https://sg.finance.yahoo.com/quote/%5EIXIC?p=%5EIXIC&.tsrc=fin-srch)). We use data in 2010 to 2018 as training data and data in 2019 as test data.

Interest rates were also reviewed to assist the analysis of the NASDAQ Index (Data source from [Economic Research](https://fred.stlouisfed.org/series/BAMLH0A0HYM2)). Statistical testing later demonstrated that the rates show a close relationship with the stock market, which is further detailed in a following section.

The two sets of data were relatively neat and clean, therefore, did not require much subsequent manipulation. Both the stock market and interest market do not trade during weekends or holidays. During the preprocessing stage of the two datasets, several data points in normal trading days were identified to be missing. To resolve the data gaps, it was decided to impute the missing values with the average value of the days before and after.

## WHY EACH METHOD / MODEL?

ARIMA:ARIMA, short for 'Auto Regressive Integrated Moving Average' is actually a class of models that 'explains' a given time series based on its own past values. The ARIMA model learns from its own lags and the lagged forecast errors, so that the equation can be used to forecast future values. It is the very basic method in time series analysis and the statistical theory is explained in greater detail in the resultsection.

ARIMA+REGRESSOR:The ARIMA model is a common model for time-series forecasting but can be improved upon. The ARIMA model can additionally extend to incorporate information provided by leading indicators and other exogenous variables: you simply add one or more regressors to the forecasting equation. Interest rate was the best performing regressor introduced to the ARIMA model.

GARCH:Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model is also commonly used in economic and financial studies to estimate volatility of the data. The GARCH model was added to the ARIMA model because it is one of the most used models in stock market analysis and could further capture the pattern of fluctuations in the data, ultimately improving the original ARIMA model.

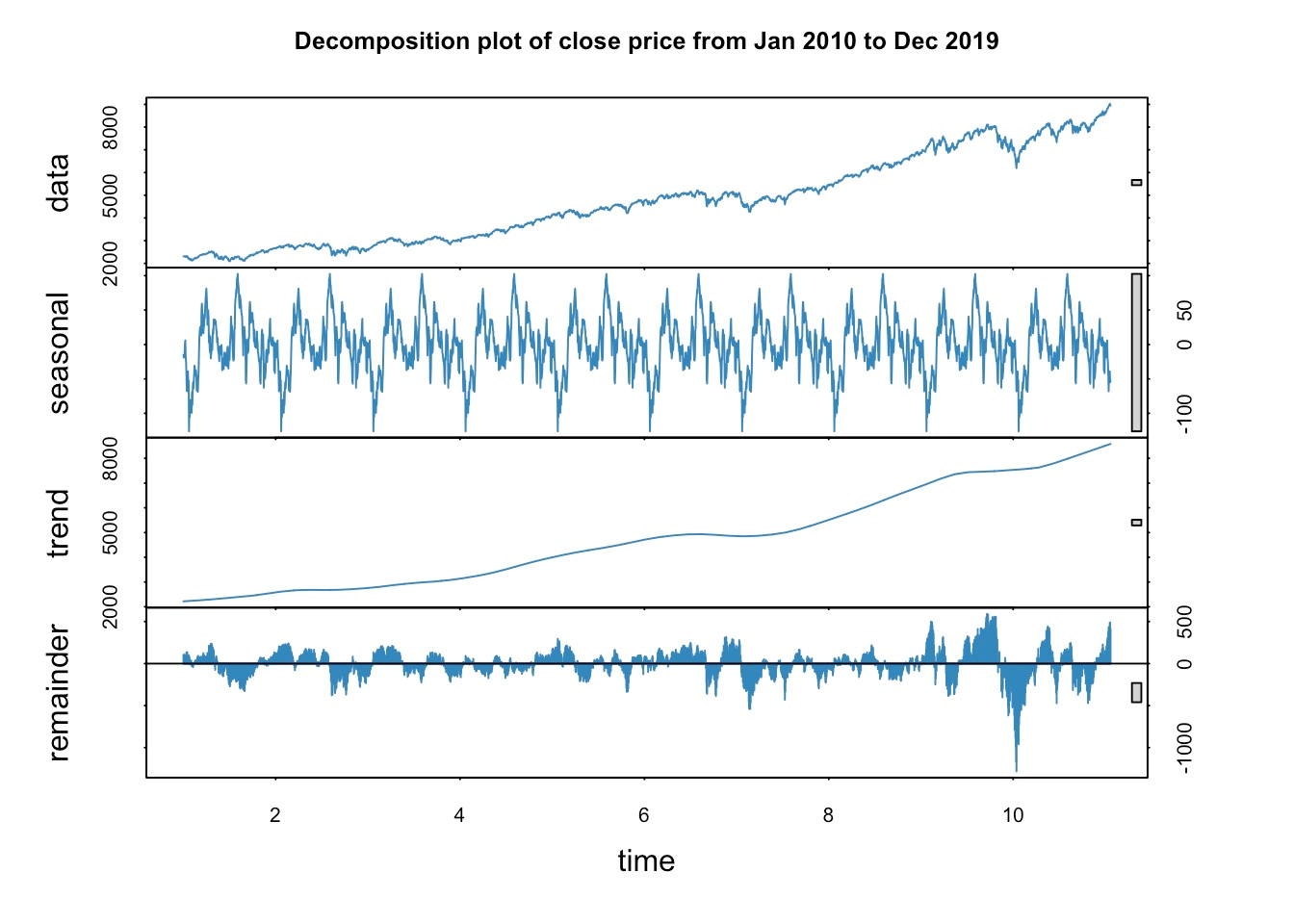
LSTM:Long Short-Term Memory (LSTM) network is an improved version of recurrent neural networks, which is able to automatically learn arbitrary complex mappings from inputs to outputs and support multiple inputs and outputs. These are powerful features that offer a lot of promise for time series forecasting, particularly on problems with complex-nonlinear dependencies, multivalent inputs, and multi-step forecasting.

## ARIMA

ARIMA modeling is based on the assumption that the series is stationary, when the mean and variance are constant over time. Before going into the ARIMA modeling, the series needs to be transformed to be stationary. Each time series can be thought as a mix between several parts :

* A trend (upward or downwards movement of the curve over the long term)
* A seasonal component
* Residuals

The decomposition plots of the NASDAQ index can help us find a pattern and transform the series to become stationary.



There is clear evidence in the above illustration of an increasing trend as well as seasonal patterns, therefore applying differencing is a good way to remove these patterns. The plots of the series after the 1st differencing transformation becomes stationary. The Augmented Dickey–Fuller test is used to double check the stationarity, with a p-value of 0.01 and can be interpreted that the transformed series is suitable for ARIMA modeling.

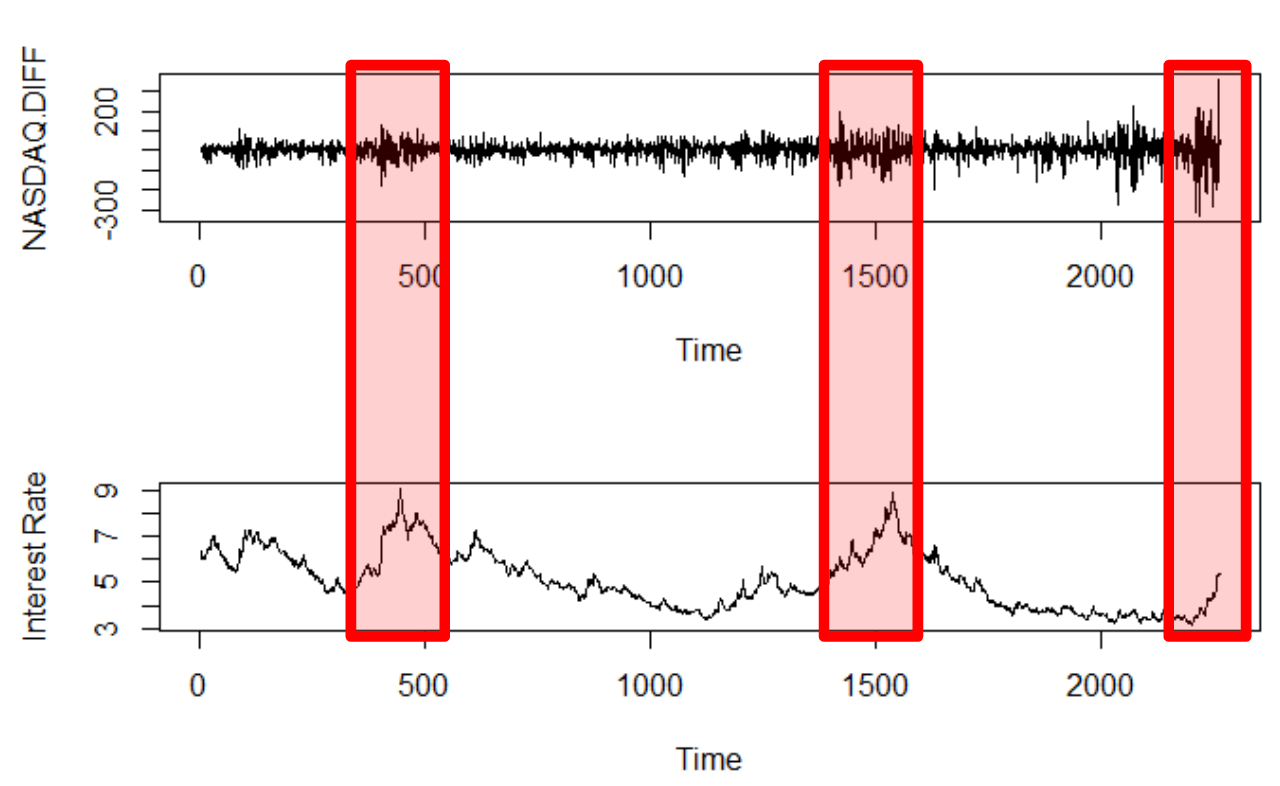
An ARIMA (p,d,q) model is parametrized by the following three integers:

* p: number of autoregressive terms (AR order), which could be interpreted by PACF.
* d: number of nonseasonal differences (differencing order). And in this case, d=1.
* q: number of moving-average terms (MA order), which could be interpreted by ACF.

## ARIMA + Regressor

Based on the purpose of further improving our model, we consider adding on a regressor -- INTEREST RATE, which is closely related to stock price.

As mentioned before, the interest rate market has a close relationship with the stock market. By plotting the difference of the NASDAQ Index and the interest rate below, the two variables almost fluctuate at the same time. These similar fluctuations occurring during the same time intervals demonstrate a significant relationship.

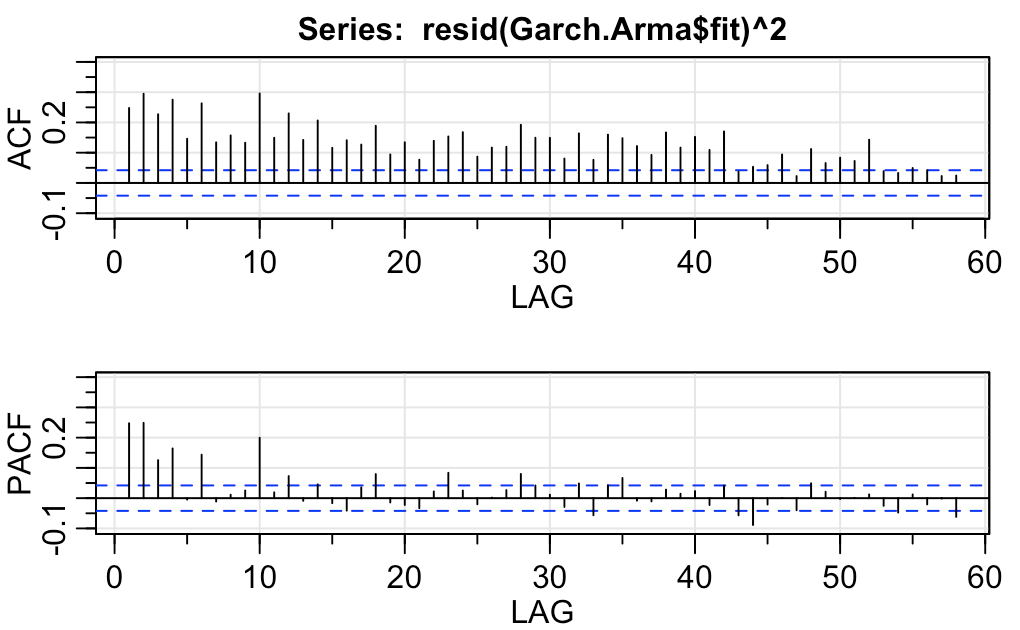


Again, the *auto.arima()* function can be used to provide a better approach to the dataset. The function returns a model as ARIMA(4,1,1), which is further broken down in this section.

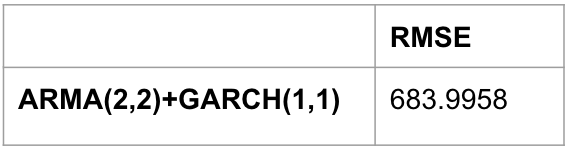
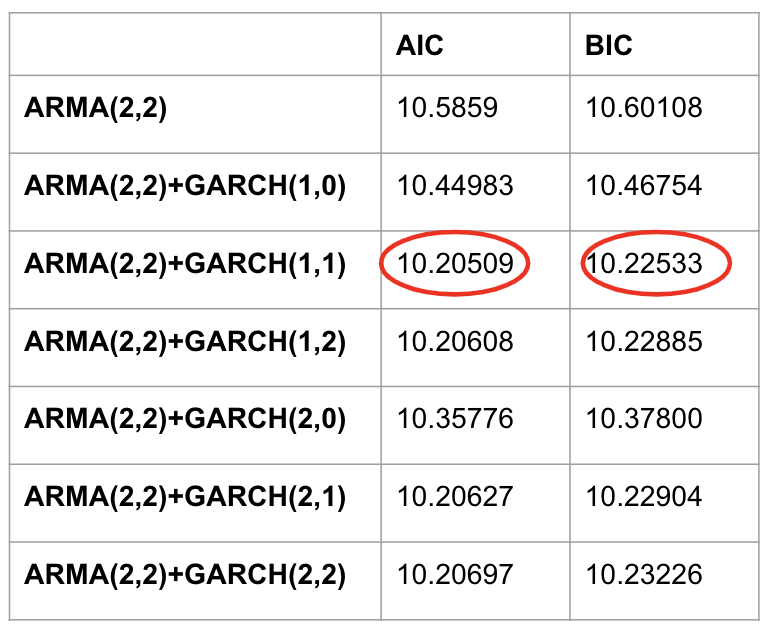
## GARCH

Another approach we used to improve the performance of the basic ARIMA model is the GARCH model. Since the ARIMA model is ARIMA(2,1,2) with a d = 1, the GARCH model can only be added to ARMA model by using the garchFit function from the fGARCH package in R. Before implementing, a differentiation must first occur in the original data and then apply the ARMA(2,2) model on the results. This can also be equivalent to applying a ARIMA(2,1,2) model on the original data.

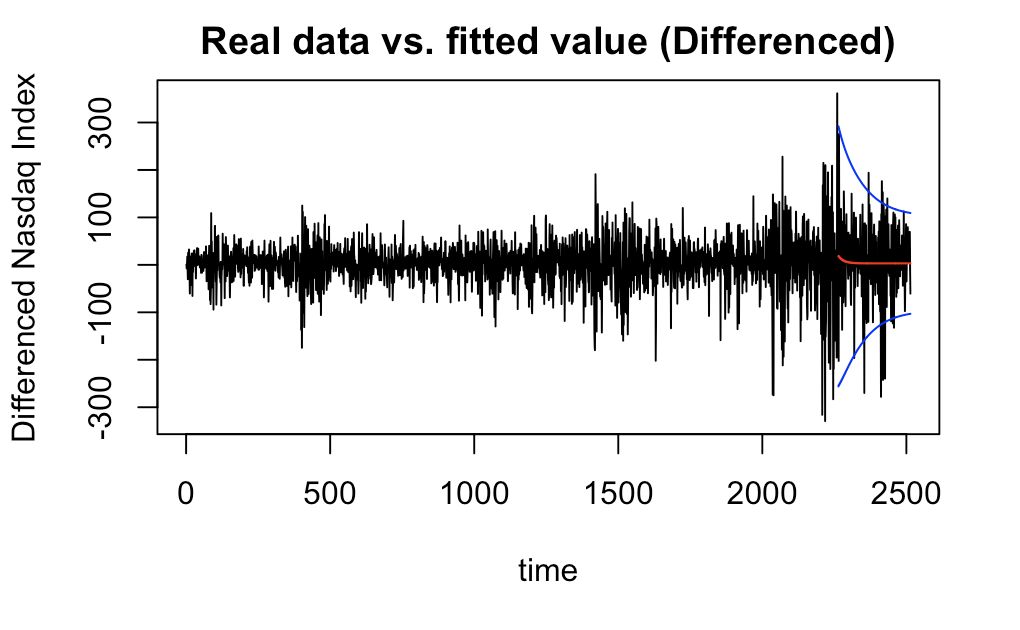
ACF (auto-correlation function) and PACF (partial auto-correlation function) plots were subsequently developed to understand the squared residuals of the ARIMA model. Based on the plots below, the squared residuals die down while the number of lags grows, and the lags with the highest values are within the first few lags.



Therefore for the ARMA(2,2)+GARCH(p,q) models, it was decided to try p = 1, 2 and q = 0, 1 and 2, then fit each of them to the data, and finally compare the resulting AIC and BIC values to find the best fit.



From the above chart, it is obvious that all the ARIMA+GARCH models’ AIC and BIC values are smaller than the original ARMA(2,2) model, which means all the GARCH models have some improvement to the original model. Among the six models, the best model is ARMA(2,2)+GARCH(1,1) with the smallest AIC and BIC values of 10.205 and 10.225. Therefore, ARMA(2,2)+GARCH(1,1) was chosen for prediction on the 2019 test data. In the graph below, the black line is the real data, the red line is the forecast value, and the blue lines are the upper and lower confidence intervals of the prediction. Although the line of the forecast sticks in the middle, the lines of the confidence intervals seem to effectively capture the pattern of volatility and reflect the range of the actual data.

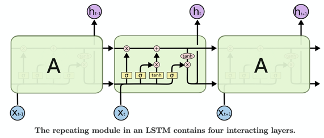


## LSTM

### Intro to LSTM

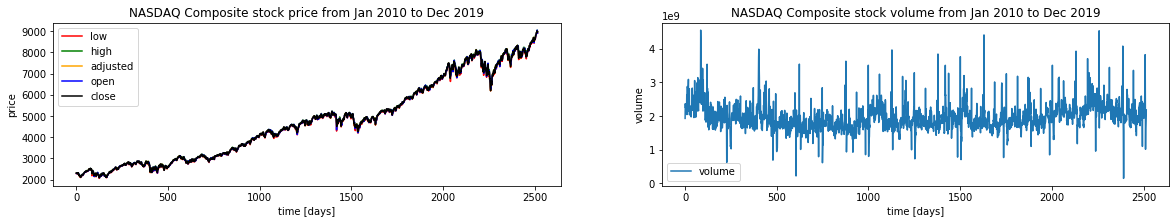
Time series forecasting can be considered difficult. Unlike the easier problems of classification and regression, time series problems add the complexity of order or temporal dependence between observations. Traditionally, time series forecasting has been dominated by linear methods, like ARIMA, because they are well understood and effective on many problems. But these classical methods also suffer from some limitations, such as assuming a linear relationship that excludes more complex joint distributions.

An LSTM model is an improved version of recurrent neural networks, which is capable of automatically learning the temporal dependence from the data. Similarly, RNNs are networks with loops in them, which allow them to use past information before arriving at a final output. However, RNNs can only connect recent previous information and cannot connect information as the time gap grows. This is where LSTMs come into play; LSTMs are a type of RNN that remember information over long periods of time, making them better suited for predicting stock prices.

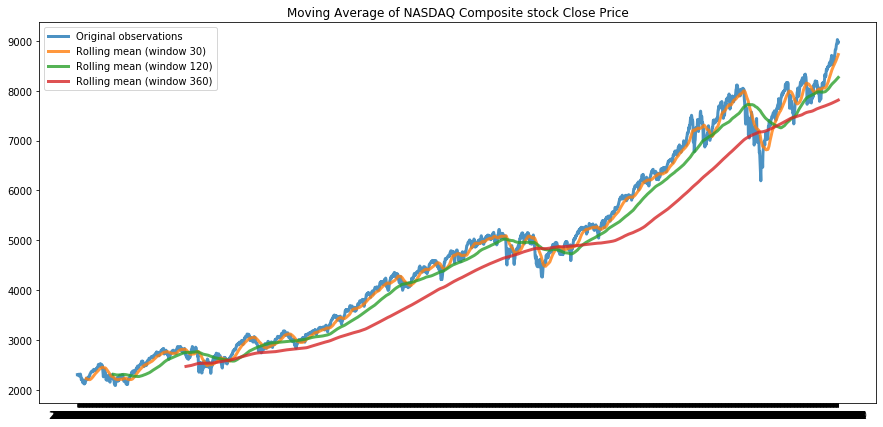


### Feature Engineering

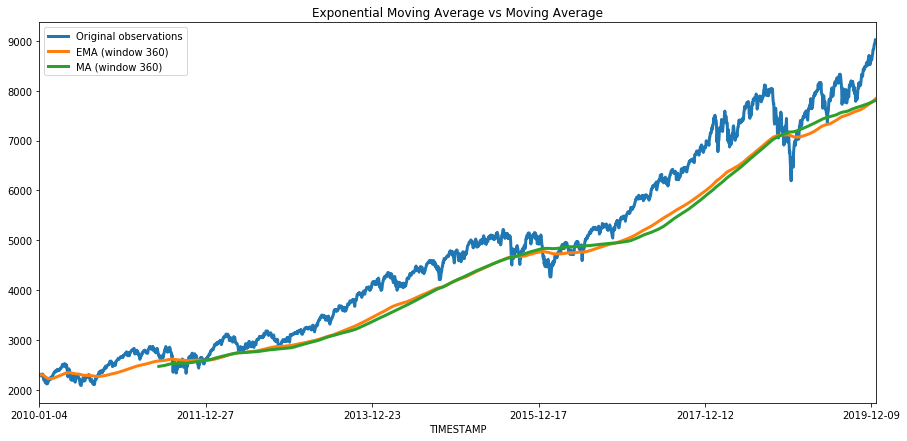
Since LSTM neural networks are able to take in multiple input variables, data preparation methods were implemented to improve the performance. Features such as open, close, low, high, adjusted price and volume from the stock data with interest rate were also considered. From the plots below, it can be inferred that there is barely any difference among these prices, but the volume seems to be an insightful variable to be included in the model.



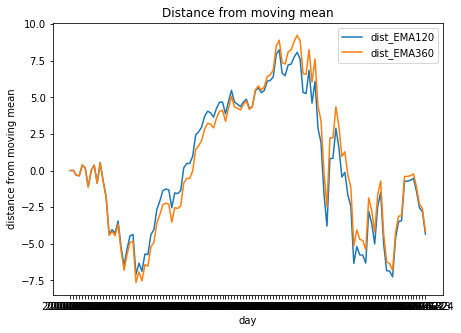
Secondly, smoothing was added to the closing price data using a moving average and exponential moving average. This feature engineering helps to further remove the inherent raggedness of the data and produce a smoother curve.



From the plot below, the exponential moving average with an orange line that catches the downwards movement in Dec 2018 faster than the moving average with the green line. In addition, in early 2016, an exponential moving average started to catch an upwards movement, while the moving average still appeared as a downwards trend.



Therefore, it was decided to use the distance from the exponential moving average mean as an additional feature in the analysis to better catch the fluctuation of data.



By adding in the additional features mentioned above to the original stock indices and interest rates, the overall performance of the LSTM model resulted beyond expectations. The results are further broken out in the next section.

# RESULTS

## ARIMA

Adapting the ACF and PACF plots helped find the possible combination of orders.

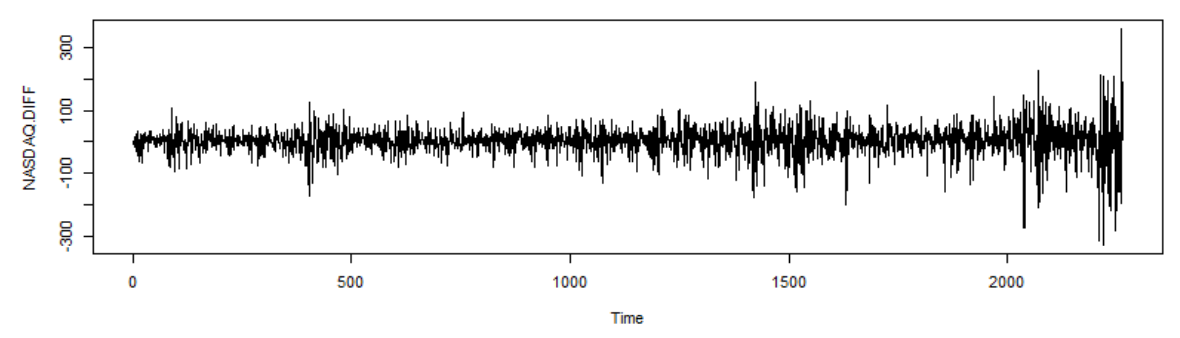
* **Autocorrelation function plot (ACF)**: Autocorrelation refers to how correlated a time series is with its past values, whereas the ACF is the plot used to see the correlation between the points up to and including the lag unit.

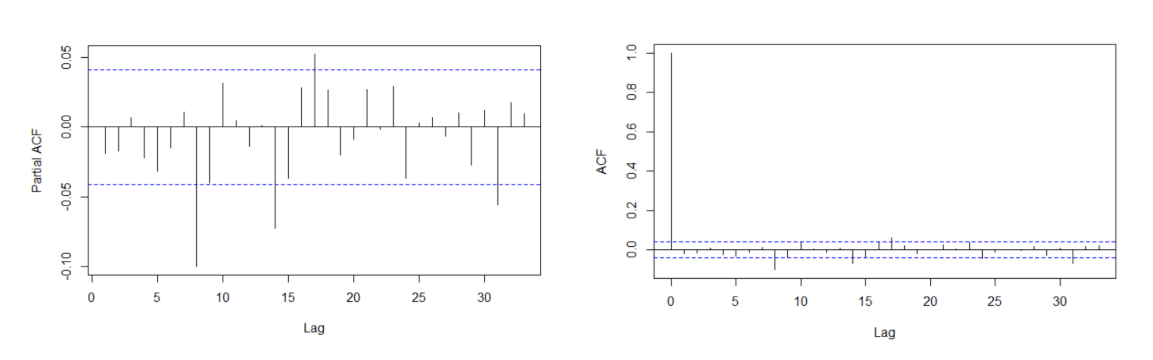
In this case, the peak cuts off after lag>0, so the possible q could be 0, 1 or 2.

* **Partial Autocorrelation Function plots (PACF)**: A partial autocorrelation is a summary of the relationship between an observation in a time series with observations at prior time steps with the relationships of intervening observations removed.

In this case, the values seem to be within the confidence band, so the possible p could be 0, 1 or 2.

Six models are fitted based on the possible p and q values. AIC and BIC are then used to choose the best model. Based on the AIC and BIC, ARIMA(0,1,0) has the lowest BIC while ARIMA (2,1,2) has the lowest AIC. By comparing the significance of coefficients to further explore, the results show the coefficient of ARIMA(0,1,0) is insignificant, while the ARIMA(2,1,2) model has most of the significant coefficients. Thus, ARIMA(2,1,2) is the best model. Adapting *auto.arima()* also gives this result, indicating the ARIMA(2,1,2) model is best suited for ARIMA modeling.





The Ljung-Box test additionally resulted in the following:

* H0: the model does not show lack of fit (or in simple terms—the model is just fine)
* HA: the model does show a lack of fit. In other words, the residuals from the time series model resemble white noise.

Thus, a good forecasting model needs to have zero correlation between its residuals or forecasting would not be possible. A significant p-value in this test rejects the null hypothesis that the time series isn’t autocorrelated.

The test gives the result as follows:

*Box-Pierce test*

*data: res.auto*

*X-squared = 0.0031431, df = 1, p-value = 0.9553*

In the above test it can be inferred that the p-value is far from 0.05 which would mean that the residuals are independent. The independence is desired for the model to be corrected and improved, which is why a regressor is introduced in the next steps.

## ARIMA + Regressor

With an ARIMA(4,1,1) model and from using an auto function, finding additional model parameters are not necessary.

The Ljung-Box p-values of the model with the regressor are updated with the following:

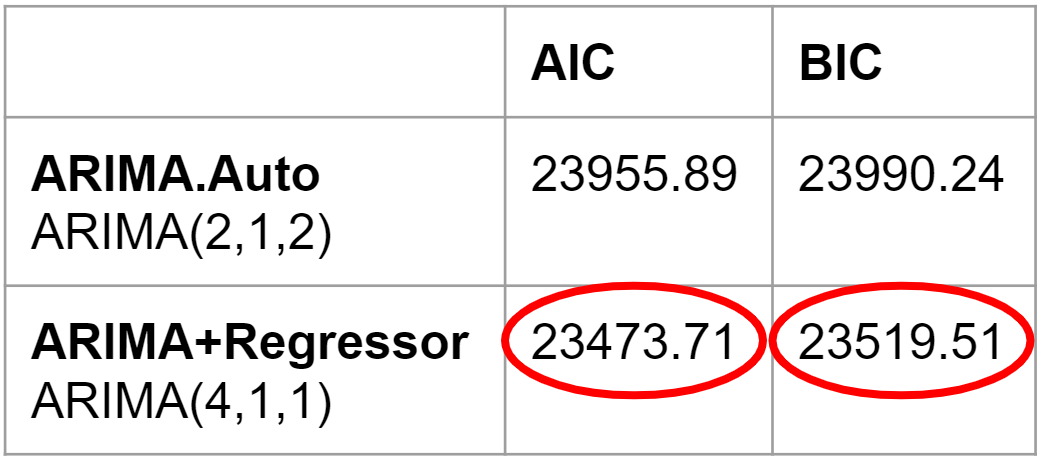
*Box-Pierce test*

*data: res.xreg*

*X-squared = 3.6246e-06, df = 1, p-value = 0.9985*

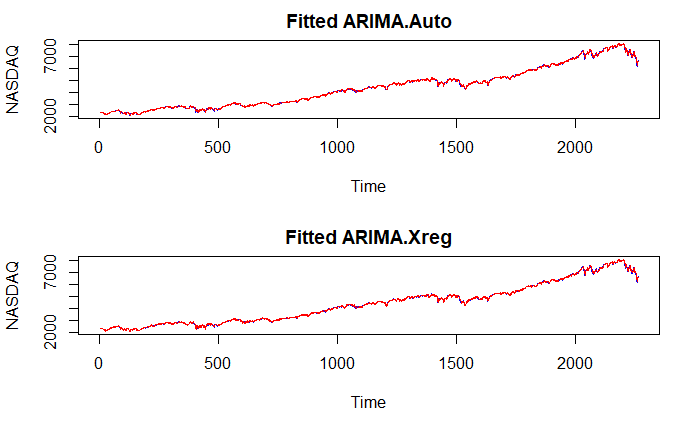
The p-value is far from 0 so that the null hypothesis is again not rejected, allowing to continue the study with a solid motivation.

Which one is better, with or without a regressor? An answer can be found by comparing the AIC and BIC of the two models.

**

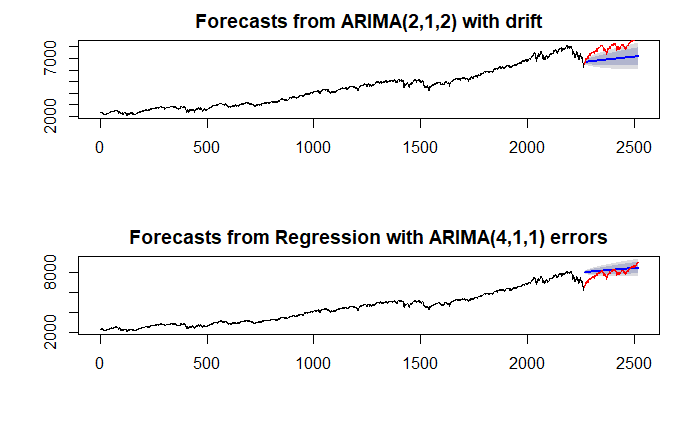
The ARIMA.Xreg model has both smaller AIC and BIC. Therefore, adding a regressor benefits the performance of the model.

Having ARIMA models applied and analyzed, visualizations can be developed to plot the model prediction with a red line over the predicted stock value and a blue line over the actual stock value.

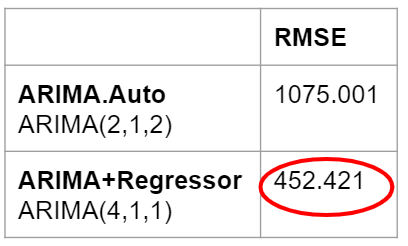


Both plots indicate that the models fit the actual data quite well. However, these are still auto-regressive models, so very good past predictions are expected. Now with the model fitted, forecasting can be generated from daily close price values. Forecasting on the close Nasdaq index for the next year, 2019, can be focused.

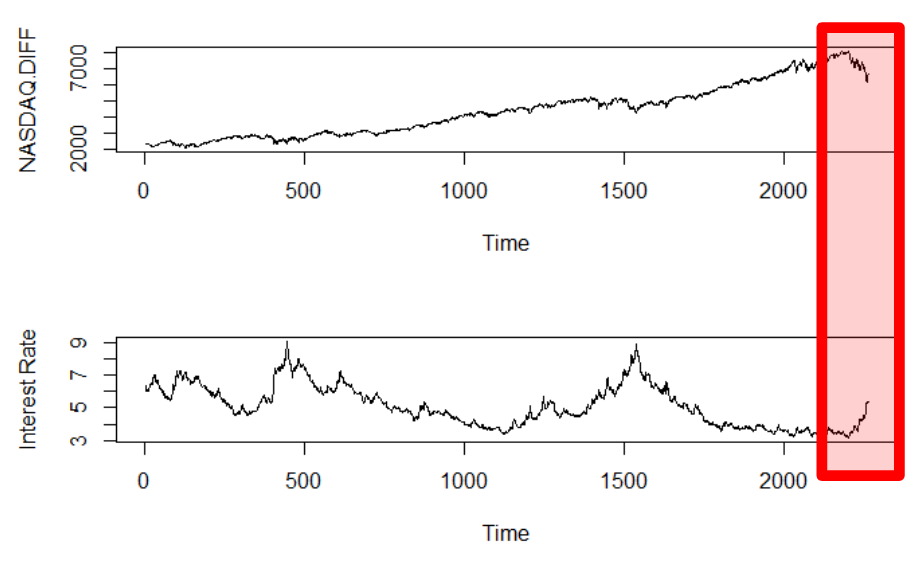
In the below illustration, a blue line represents the mean of the prediction and a red line represents the actual data of 2019. With the blue line explained, darker and light darker areas can also be viewed, representing 80% and 95% confidence intervals respectively in lower and upper scenarios.



Both models indicate an upward trend. There is a downward trend at the end of the training data. The first model is affected by this and gives a lower prediction in 2019. While the second model overcomes this misleading information and gives a relevantly accurate prediction, much better than the first one. The RMSE of prediction results also shows the following:



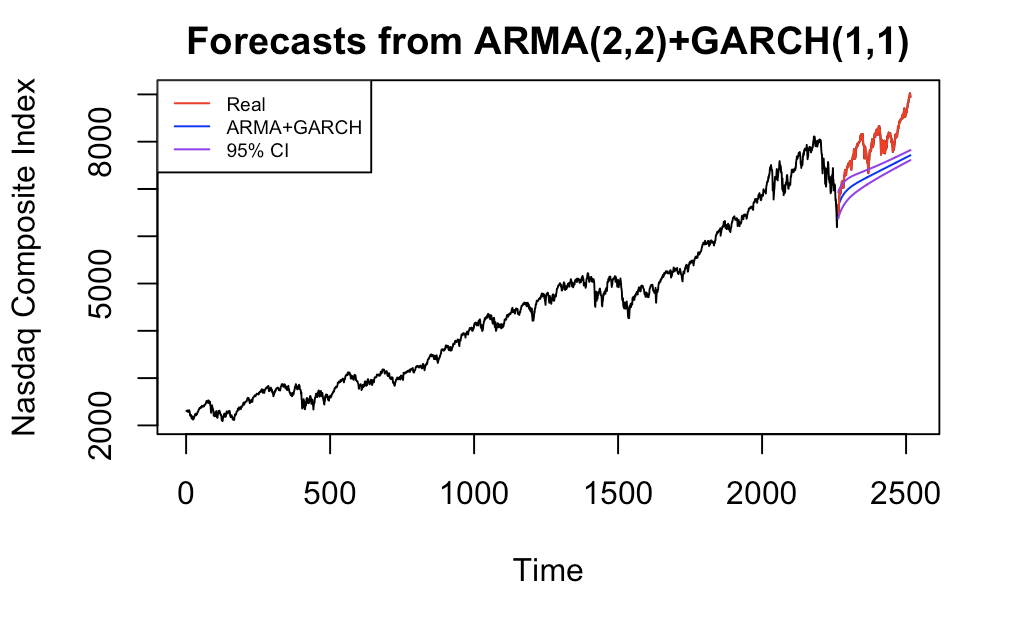
Based on the above RMSE, the ARIMA model with a regressor can make better predictions.



When the NASDAQ Index descended in late 2018, the interest rate starts on an upward trend. It is possible to assume that this wave of stock descending is due to the increase in the interest rate. Since interest rate would be adjusted in the near future, it's impossible to increase consistently, otherwise, the stock price would quickly resume an increasing trend.

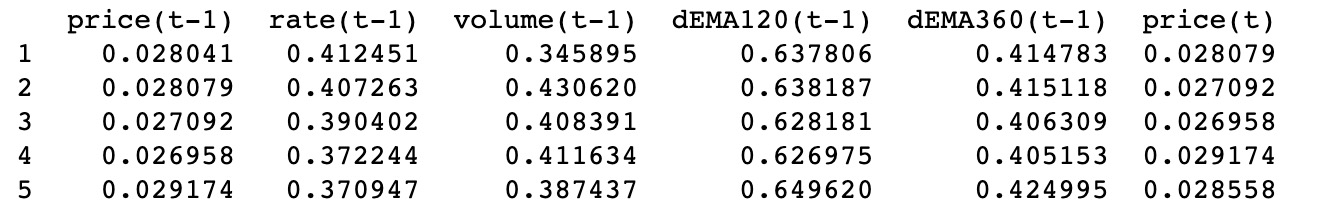
## GARCH

In order to see the actual prediction result and compare the ARMA+GARCH model with other models built, an inverted differencing process was implemented on the prediction result. The figure below shows the actual prediction result after the inverse-differencing, where the blue line is the forecasted value and the purple lines are the 95% confidence interval. The GARCH model estimates the volatility, by adding curvature to the straight line forecast of the ARIMA model and appears closer to the actual data than the basic ARIMA model.



## LSTM, Deep Learning

In the LSTM model, the supervised learning problem can be framed as predicting the closing stock price at the current day, given all the features of the prior day.

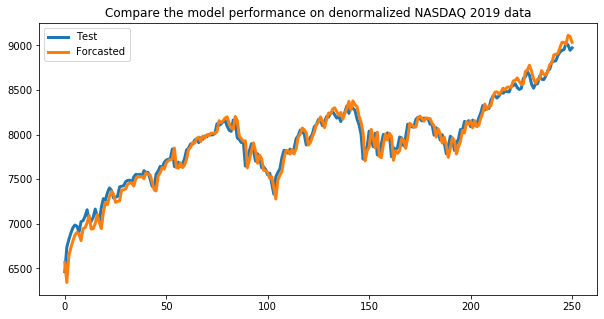


Different features are added from selections to compare how they work. From the bar plot, the model with the interest rate is better than the one without. Ultimately, the best model is the one with the interest rate, volume, and two features generated by the exponential moving average.

A picture containing screenshot

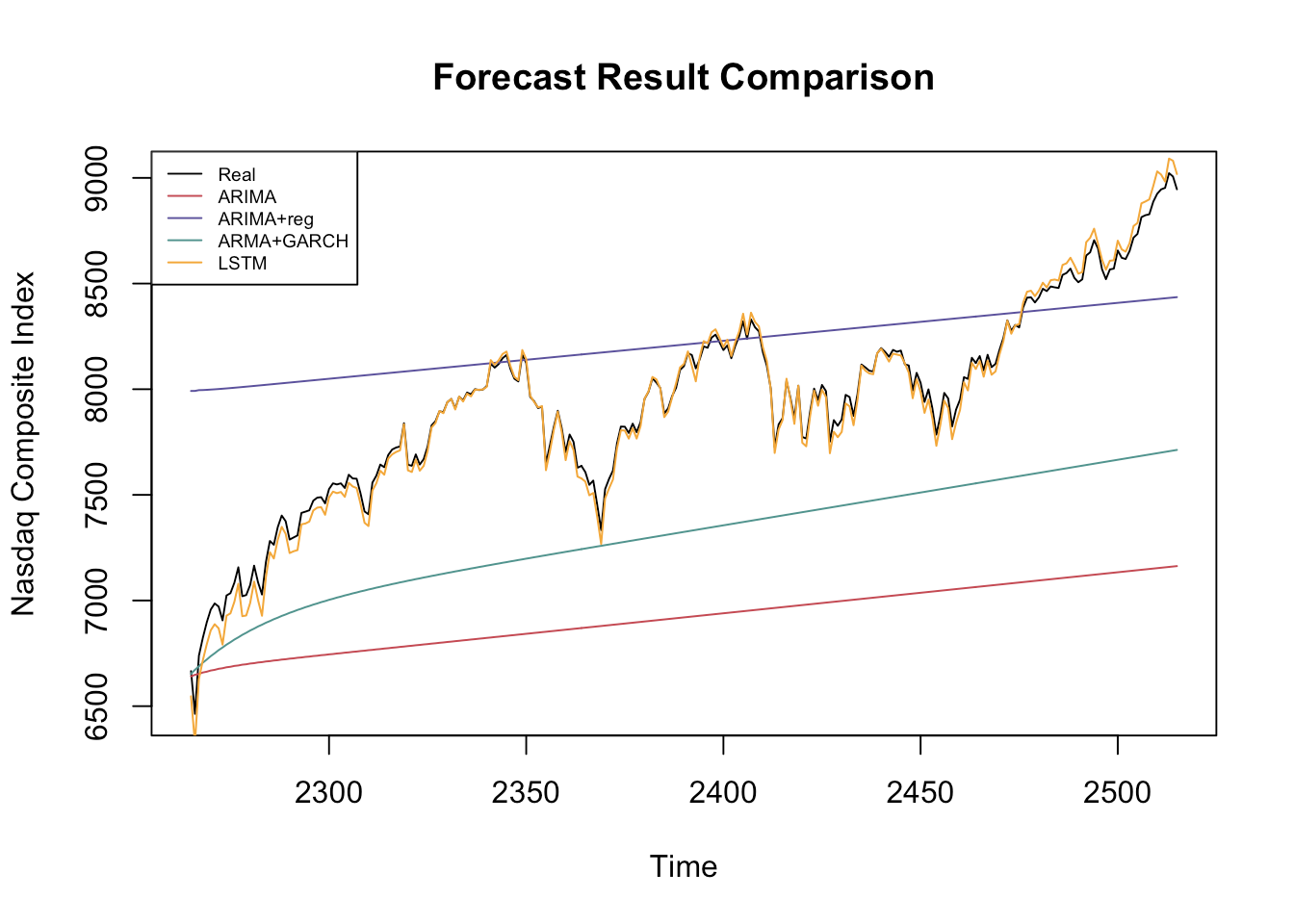
Description automatically generated

In the line plot below, the LSTM model can be seen how well it fits with the 2019 data.



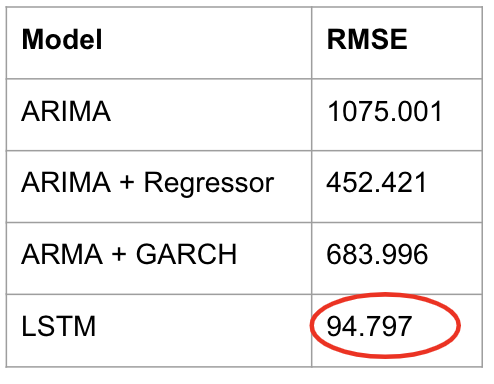
## Result Compare Table

After finishing all of the mentioned models in this analysis, evaluation and inference can be made from the following plot that compares each model’s performance:



As illustrated from the comparison plot, the prediction result of the basic ARIMA model (red) is a straight line, and it is the farthest from the actual data. The ARIMA model with regressor (purple) is still a straight line, but it is much closer to the actual data. The ARMA+GARCH model (green) better captures the trend and adds curvature to the straight line, which works great for the first 50 data points, but it goes further and further away from the actual data afterwards. On the other hand, the LSTM model (orange) has amazing performance compared to all other models.

After calculating the RMSE, a more precise comparison can be made between the performance of all models. The table below lists the RMSE of the basic ARIMA model of around 1000, while the regressor and GARCH models are in the middle with an RMSE of around 450 and 680, and the LSTM model with a quite remarkable value that is less than 100.



To sum up, the ARIMA model is the most straightforward and interpretable model which can be widely used. By adding regressors and the GARCH model to ARIMA, the prediction result gets better in different ways: the regressor model helps find a better fitted straight line for a longer term, while the GARCH model takes volatility into account and works better in the short term. However, all these time series models can only predict the general trend. By innovating with cutting-edge stepwise, deep learning models, in our case, LSTM, it has very high flexibility, and can learn and train itself. An advantage of LSTM is that it is a nonlinear function with neural networks, therefore, it works very well for fluctuating series like the data used in this analysis.

# CONCLUSION

All the models were able to predict the future NASDAQ index to some extent, and the LSTM architecture stands out as the best model. Since the deep learning model gave the best performance on the 2019 test data, additional steps can be made to see how it would perform on the NASDAQ closing stock price for 2020. However, the performance of the 2020 data is significantly worse than that of 2019. As many of you are aware, the current COVID-19 pandemic has created lots of uncertainty, especially in the global economy. As such, even the best model didn’t pick up on such a large external factor for its prediction but appears to perform not terrible in terms of the overall trend.

Since a deep learning approach performed so well on the NASDAQ index, it would be interesting to see how the model would work with other stock datasets like the Dow Jones for further analysis. If additional time was allowed to work on this analysis, it could be worth exploring external factors in depth that might impact stock data, such as news sentiment data, then combine with the best deep learning models. As data scientists and academia further explore time series analyses using this new field of artificial intelligence, it’s likely that time series forecasting standards will change in the near future.

# Appendix

Please include your codes->html or pdf knitted from red. In the appendix

## NASDAQ.csv:

This is the data file we used for this project.

* Data Source:
  + NASDAQ Composite Index: [https://sg.finance.yahoo.com/quote/%5EIXIC?p=^IXIC&.tsrc=fin-srch](https://sg.finance.yahoo.com/quote/%5EIXIC?p=%5eIXIC&.tsrc=fin-srch)
  + Interest Rate: <https://fred.stlouisfed.org/series/BAMLH0A0HYM2>

## EDA\_ARIMA\_Xreg\_GARCH.html:

This is the Rmd -> html output file for EDA and our time series models.

## LSTM.html

This is the Jupyter notebook -> html output file for LSTM models.

## NASDAQ\_Presentation.pptx:

This is the Powerpoint for our presentation.

# References

[1] NASDAQ Composite (^IXIC) charts, data & news – Yahoo Finance. (2020, April 30). Retrieved March 29, 2020, from https://sg.finance.yahoo.com/quote/^IXIC?p=^IXIC&.tsrc=fin-srch

[2] “ICE BofA US High Yield Index Option-Adjusted Spread.” FRED. 2. (2020, April 30). Retrieved March 29, fred.stlouisfed.org/series/BAMLH0A0HYM2.

[3] Fuqua School of Business, ARIMA models with regressors. [Online]. Available: https://people.duke.edu/~rnau/arimreg.htm. [Accessed: 30-Apr-2020]

[4] S. Prabhakaran, “ARIMA Model – Complete Guide to Time Series Forecasting in Python,” Machine Learning Plus, 28-Apr-2020. [Online]. Available: https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/. [Accessed: 30-Apr-2020]