Netflix Stock Market Analysis and Prediction

Paul Galli

Abstract:

Netflix is considered to be one of the largest entertainment services in the world. Over hundreds of millions subscribers all over the world. The stock market is a complex network of trading and has a significant contribution to the world economy. This study aims to discover how accurate stock market prediction models can be. Using different types of models such as ARIMA, LSTM, and GARCH. Combining ARIMA and LSTM models using ensemble techniques such as simple averaging and weighted averaging produced accurate predictions. The GARCH model is used to predict volatility. The final results showed that the predicted volatility is generally inline with the daily returns. Main interpretation is that while perfectly predicting the stock market is essentially impossible. It is possible to create incredibly accurate models that can help investors, traders, and even help the economy.

Introduction:

Netflix is one of the largest entertainment services in the world with over one hundred million subscribers across the world. Founded in 1997 by Reed Hastings and Marc Randolph, the two had the idea of sending dvd rentals through the mail. They successfully tested the idea on themselves by mailing a DVD to themselves and it arrived fully intact. In 1998 the two launched Netflix.com as a DVD rental company with a flat rate subscription service. At first it started out as no late fees or due dates which helped the company grow in popularity. In 2000, the company created a movie recommendation feature to target and retain customers as well as predict users preferences. In 2002, the Netflix stock was opened at one dollar per share on NASDAQ. The company reached one million users in 2003 and by 2006 it had grown up to five million. 2007 is when the company added the ability to stream movies online. By 2011, Netflix had opened operations in Canada, Latin America, The Caribbean, Nordic countries, and the UK. By 2020, it became the largest entertainment company by market capitalization. [1]
https://groww.in/us-stocks/nflx

The stock market is a complex network of trading where company shares are bought and sold. It plays a crucial role in economics by allowing money to move between investors and companies. Owning a stock means you own part of the company. However, there are many companies that have millions and even billions of shares. There are two types of people who invest in the stock market. Investors who focus more on long term investments and traders who focus on short term. In today's world the stock market is considered to be the center of the global economy.[2]
https://www.investopedia.com/terms/s/stockmarket.asp#:~:text=The%20stock%20market%20is%20also,the%20retirement%20you%20might%20plan

Being able to accurately predict what will happen in the stock market can be crucial to how successful the economy can be. Using tools and algorithms to predict the future stock price can allow investors to know what to expect and how shares should be allocated. The aim of this work is to successfully build those models that can accurately predict the stock market. By creating different models and comparing, contrasting, and combining we should be able to have models that perform well and are accurate.

Materials and Methods:

The dataset being used was created by using a Python package called yfinance. The data is loaded as a pandas dataframe and then saved as a csv file. The dataset consists of three thousand five hundred and fifty-two rows and eight columns. The columns include the date which is the year/month/day of that particular date, the open price which the price per share when the market opens at 9:30am, the high which is the highest recorded price per share during that date, the low which is the lowest recorded price per share during that date, the close price which is the price per share when the market closes at 4:30pm, the volume which is the recorded amount of shares that were traded on that particular date, dividends which is the distribution of a company's earning to its shareholders, and finally stock splits which is a ratio that is calculated when a company increases its number of shares in order to boost the stock's liquidity. Before creating any models there was a need to preprocess the data in order for it to be correctly implemented. The index of the dataset needed to be set to the date so the programs would be able to successfully graph the data. Since the Netflix stock went public in 2002 there appeared to be very little and insignificant growth until 2010. The features that are used are the close price which is displayed in figure 1. Other features that will be used are returns which is either the profit or loss of a share during that trading day. The return is calculated by subtracting the opening price by the closing price. This feature will be used to predict volatility. The last feature that will be used is the Relative Strength Index (RSI). The RSI is used to determine sell signals and buy signals.

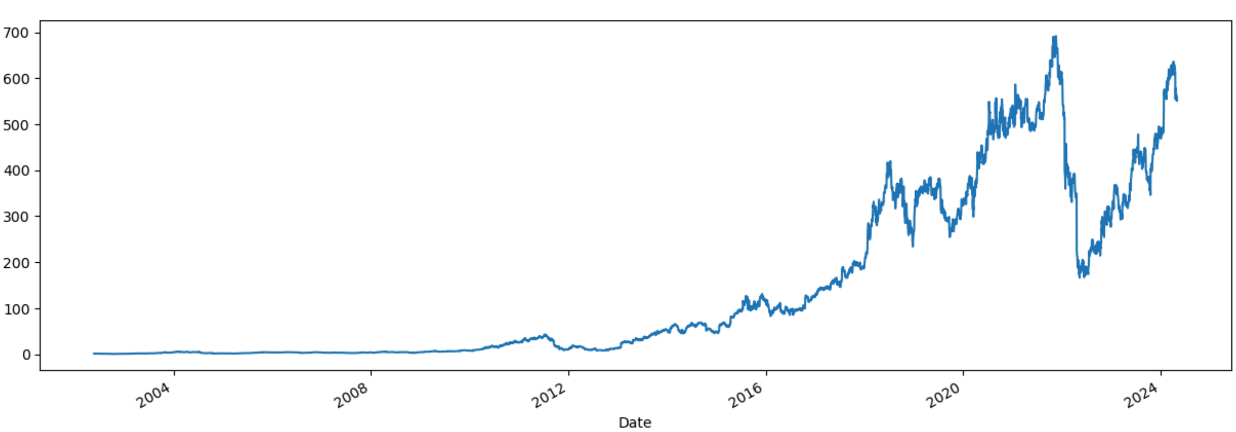

Figure 1.

Any data that was recorded before January 1st, 2010 was discarded. Our dataset includes every trading day from January 1st 2010 to December 31st, 2023. Displayed in figure 2.

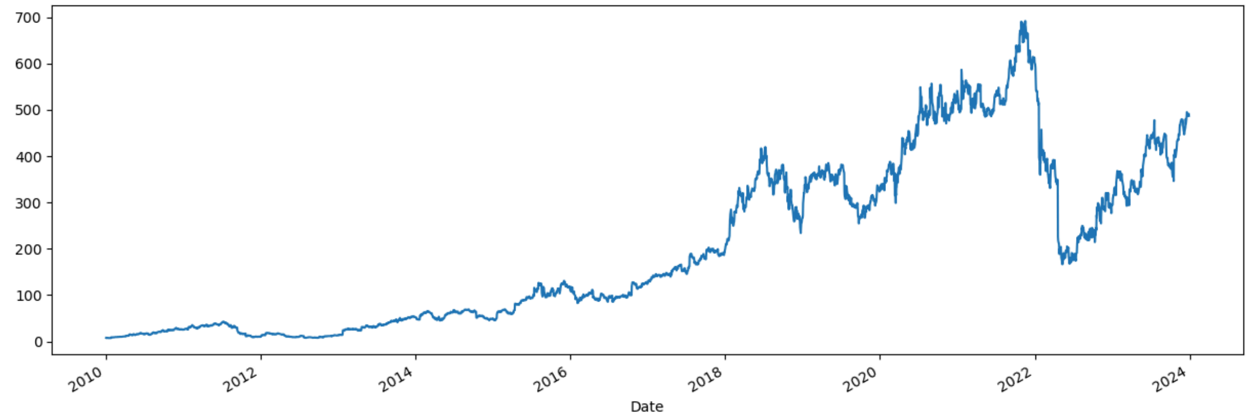

Figure 2.

Once the required data was gathered the next step for preprocessing was to test for stationarity using the first feature close price. The test used was the Dickey Fuller Test. This test is used to determine if our series has a unit root. If we fail to reject the null hypothesis then we can conclude that our series is not stationary. The results of the test are presented in figure 3.

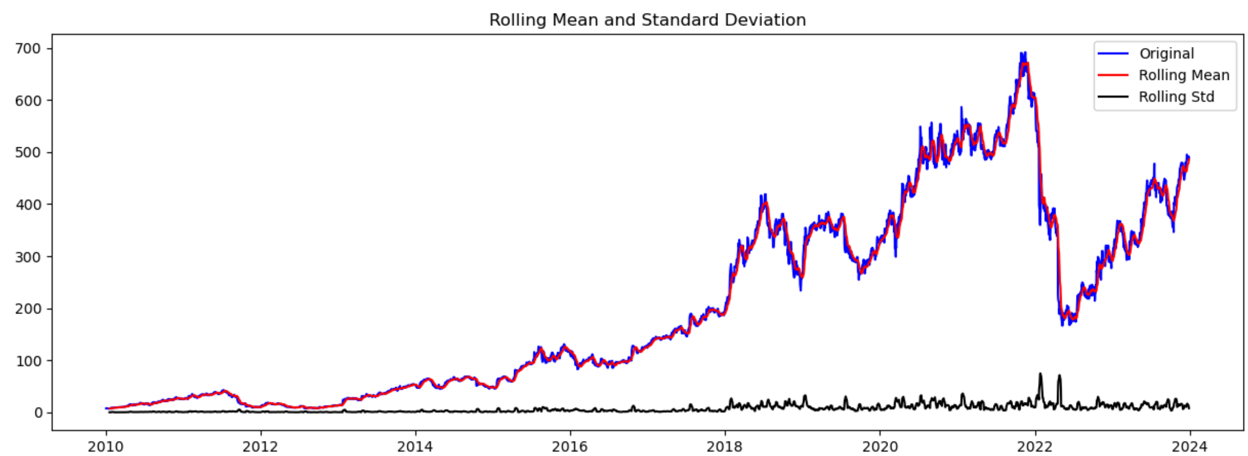

Figure 3.

The p-value of the test is 0.8 and with an alpha level of 0.05 we fail to reject the null hypothesis and conclude that this series is not stationary. In order to create a stationary series there will be a need to difference the data. I conducted a first difference which subtracts one period by the previous period. The results are shown in figure 4.

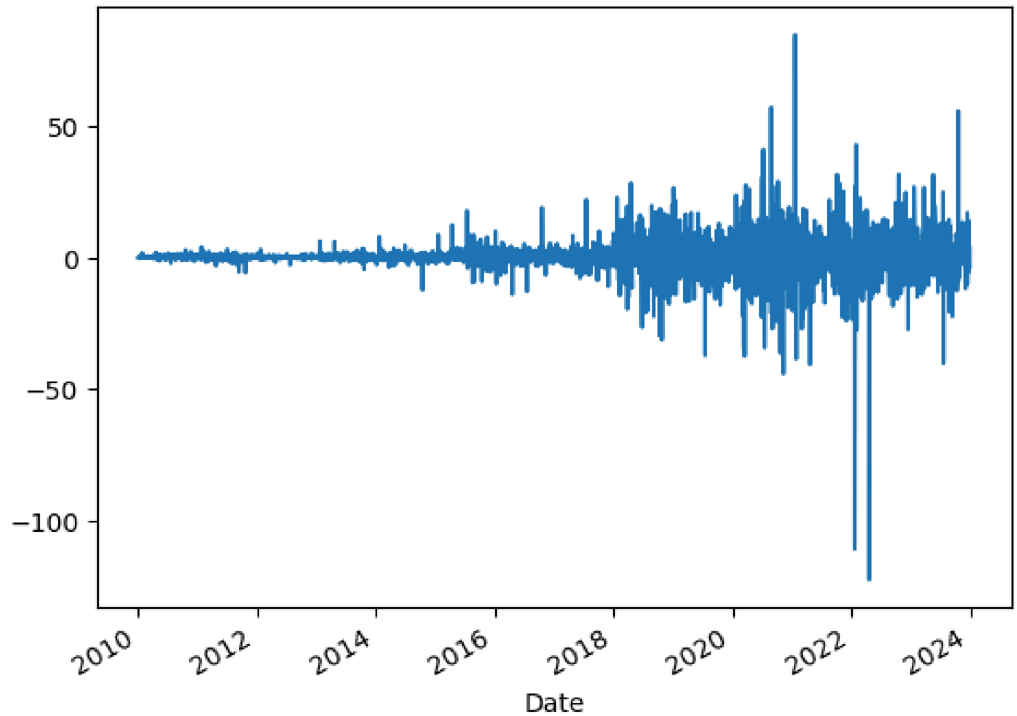

Figure 4.

With a p-value of 3.532446e-29 which is significantly less than alpha level of 0.05. We conclude that our series is now stationary. The first model that is being created is the ARIMA model which stands for Autoregressive Integrated Moving Average. ARIMA models can predict future values based on past values. They make use of lagged moving averages to smooth time series data. The Autoregressive (AR) uses the dependent relationship between an observation and some predefined number of lagged observations, the Integrated (I) means the model employs differencing raw observations in order to make the time series stationary, and the Moving Average (MA) exploited the relationship between the residual error and the observations. In order to create the ARIMA model we will need to determine three parameters. The first parameter is *p.* This value determines the number of past values used to predict the current value. In order to determine *p* I will examine the Partial Autocorrelation Function (PACF). The next parameter is *d.* This value is the difference value. Since I did a first difference this value will be 1. The last parameter is *q.* This value determines the number of past forecast errors used to predict the current value. This value will be determined using the Autocorrelation Function (ACF). The ACF and PACF are displayed in figures 5 and 6.

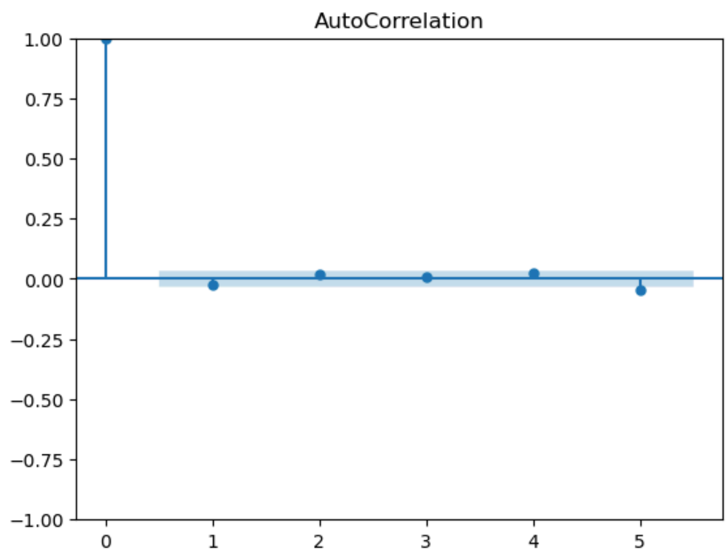

Figure 5.

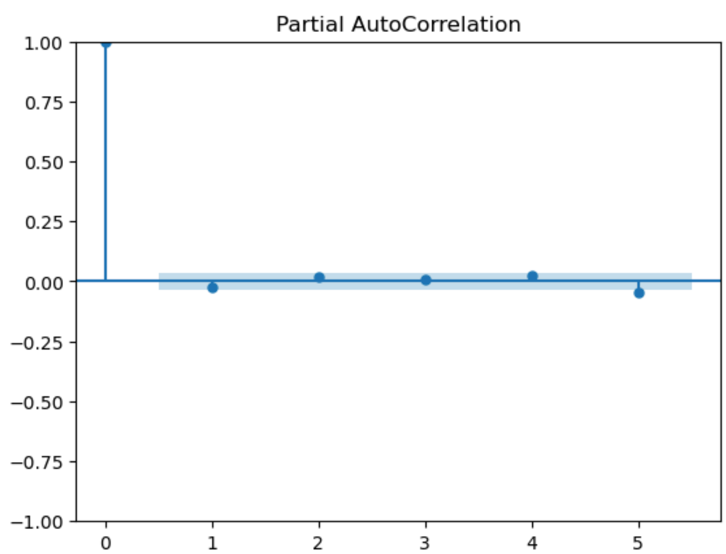

Figure 6.

Based on these results p=1 and q=1. Therefore, the ARIMA model will be an ARIMA(1,1,1) model. The final step before creating any of the models is to split the dataset into training and validating sets. The training set consists of all data from 2010 to December 31st, 2020 which consists of 2769 rows and 9 columns representing 80% of the data. The validating set consists of all the data from January 1st, 2021 to 2023 which includes 753 rows and 9 columns representing 20% of the data.

The second model that will be created is a Long Short Term Memory (LSTM) model. One issue with ARIMA models is that they are not built for long term forecasting. However, LSTM models are capable of remembering information over long periods of time. The LSTM has three components. First is the input gate, this lets new information into the cell. Second is the forget gate, this helps the cell remember important information from earlier in the sequence. Finally, the last gate is the output gate, this lets the cell produce an output that can be used for prediction. After both models have been created and tested they will then be combined into two different models. The first combined model will be a simple average model which combines the predictions by averaging them together. This will reduce the variance and improve prediction accuracy compared to individual models. However, its limitations is that it assumes all models contribute equally and independently which may not be optimal if models have correlated errors or differing performance levels. The second combined model will be a weighted average model. This combines predictions by weighting them according to each model's importance or accuracy. It can improve ensemble performance by giving more importance to better-performing models. In order to test the performance of the ARIMA, LSTM, and combined models I will be using the Root Mean Square Error (RMSE). The RMSE is calculated by first calculating the difference between the actual value and the predicted value for each observation. Then it squares each difference in order to remove negative signs and give more weight to larger errors. It then finds the average of the squared differences. Finally, it then takes the square root of the averaged squared difference. The next feature that will be used for creating these models is the RSI. The preprocessing for the RSI data if similar to that of the close price. The formula for the RSI is RSI = 100 * avg_upward_price / (avg_upward_price + avg_downward_price). An asset is considered to be overbought if the RSI is above 70% and oversold if below 30%. The comparison between close price and RSI is displayed in figure 7.

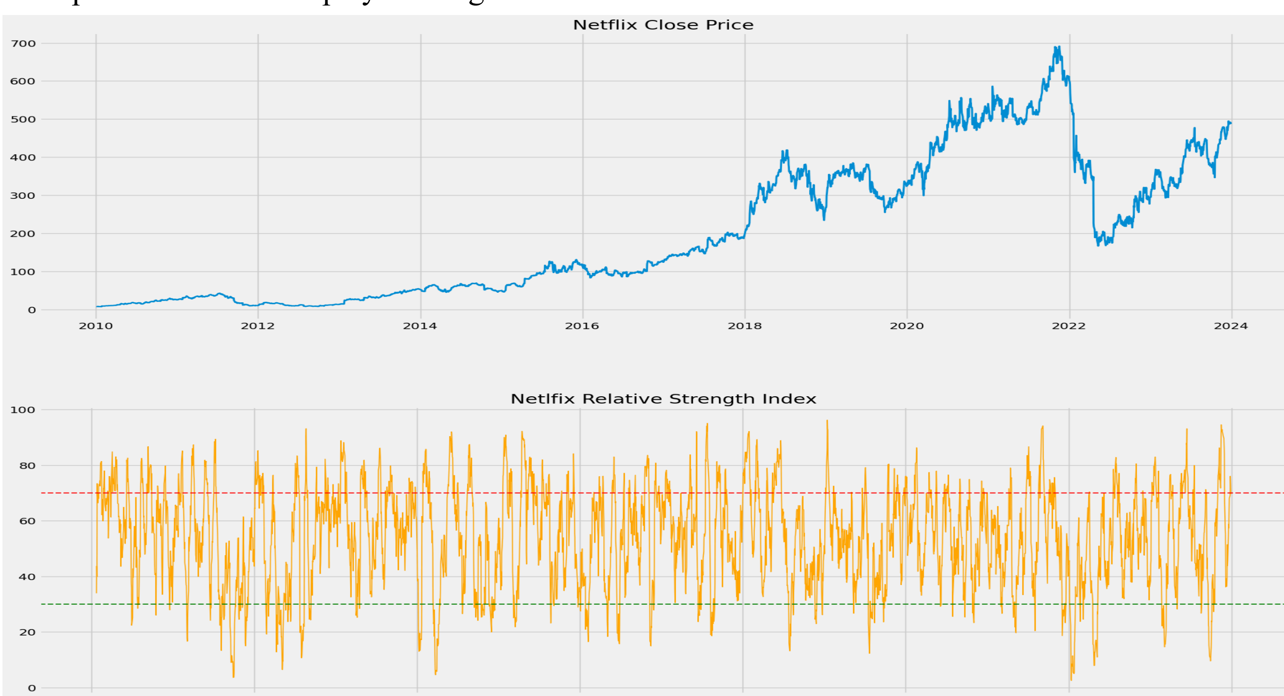

Figure 7.

The feature did consist of 13 nan values which were dropped. The values were dropped because interpolation was not successful as the values were being considered “None”. Since there were only 13 nan values I decided to drop them because of how small a percentage they were. Again this data was tested for stationarity using the Dickey Fuller Test with a p-value = 4.893007e-12 which means this series is stationary. The results are presented in figure 8.

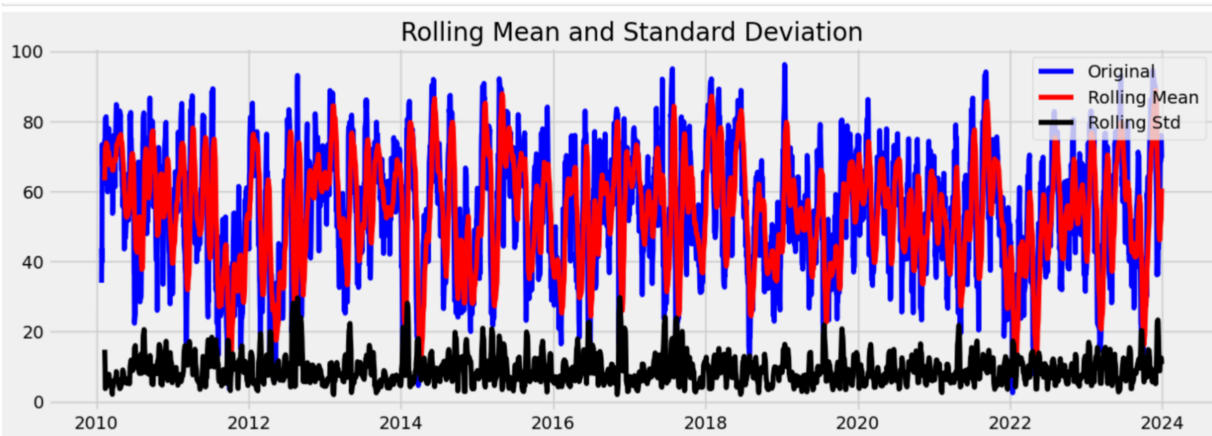

Figure 8.

The ACF and PACF are displayed in figure 9. The results conclude that the ARIMA model will be an ARIMA(1,1,0). The training and validating datasets are the same as the close price datasets.

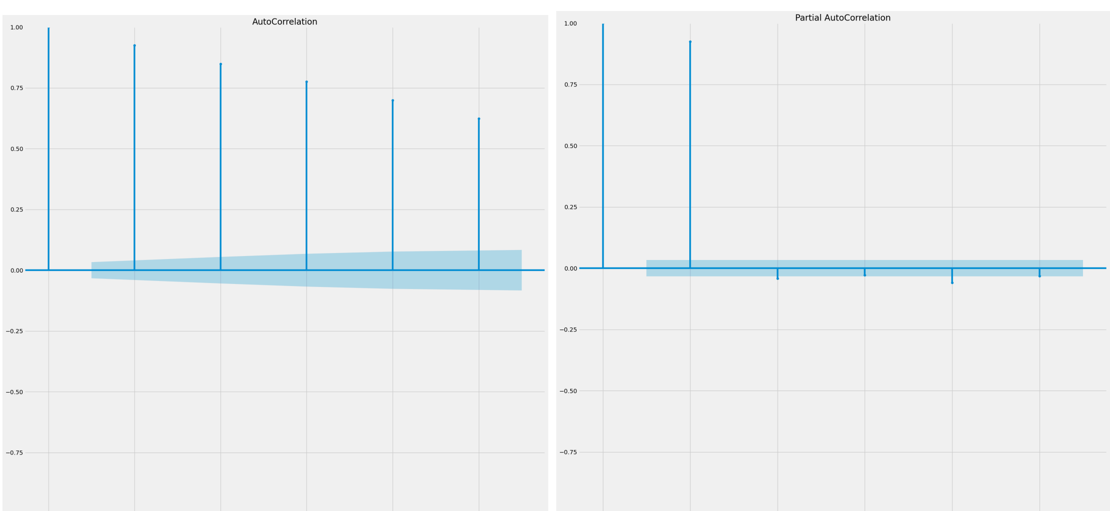

Figure 9.

The final feature that will be used is returns. The preprocessing for returns follows the same steps as the close price and RSI. Returns are calculated by subtracting the opening price by the closing price. Figure 10 displays the daily return over time.

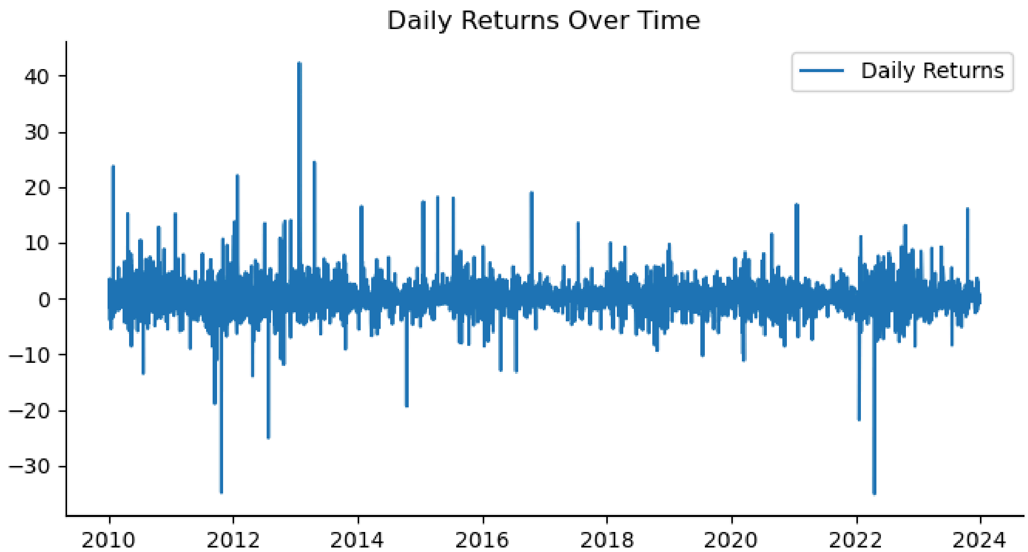

Figure 10.

Daily returns can be used to calculate the daily volatility. The daily volatility is the standard deviation of the daily return. The monthly volatility is calculated by assuming there 21 trading days in the month and multiplying the daily volatility by the square root of 21.The annual volatility is calculated by assuming there 252 trading days in a calendar year and multiplying the daily volatility by the square root of 252. The daily volatility is 3.22%, monthly volatility is 14.76%, and annual volatility is 51.13%. In order to predict the volatility of returns I will be using a GARCH model. GARCH stands for Generalized Autoregressive Conditional Heteroskedasticity. It is able to capture the long-term persistence of volatility.

Results:

The first model we will examine is the ARIMA(1,1,1) model which uses the close price from the dataset. The model's results are displayed in figure 11. The RMSE score for this model is 11.731

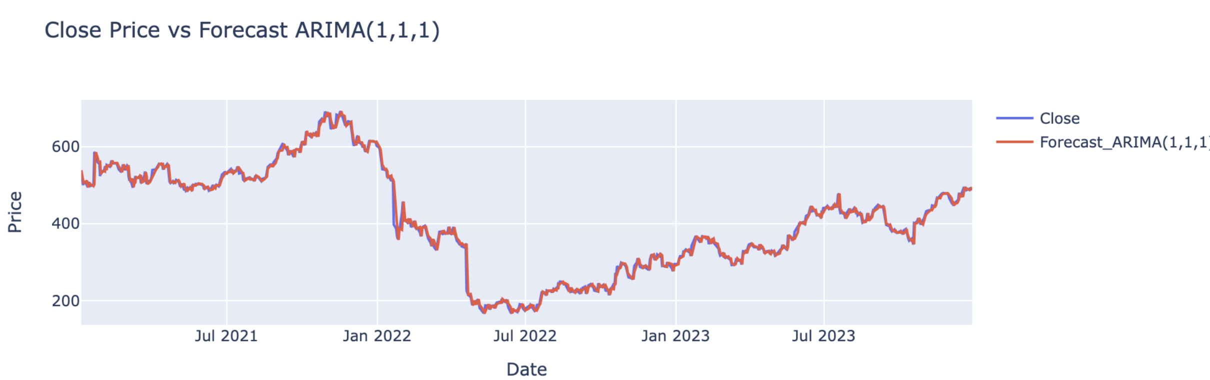

Figure 11.

The results for the LSTM model are displayed in figure 12. The RMSE score for this model is 9.634.

Figure 12.

Looking at both results we can see that the ARIMA(1,1,1) model shows better forecasts when the close price is higher. However, the LSTM model has a better performance score. The simple average model is displayed in figure 13. The RMSE score for this model is 7.662.

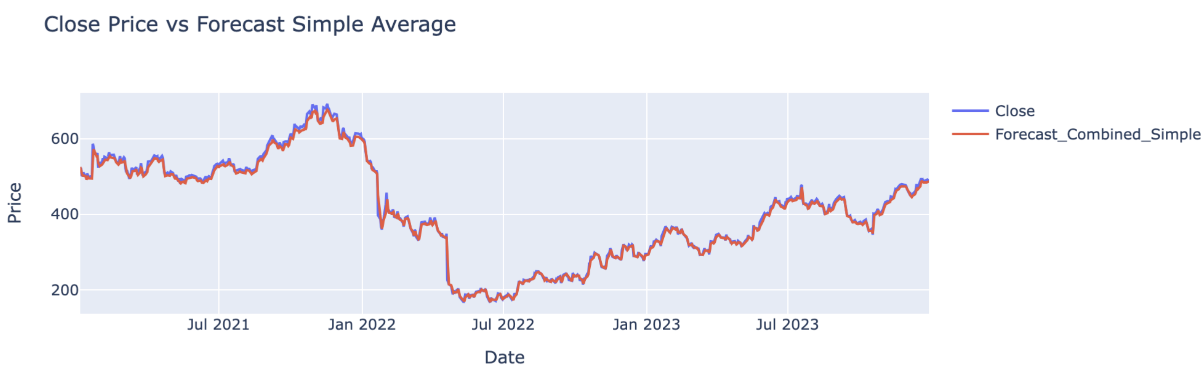

Figure 13.

The weighted combined model was created by giving the LSTM model 60% weight and the ARIMA(1,1,1) 40%. This is because the LSTM model performed better. The results of the weighted average model are displayed in figure 14. The RMSE score for this model is 7.516.

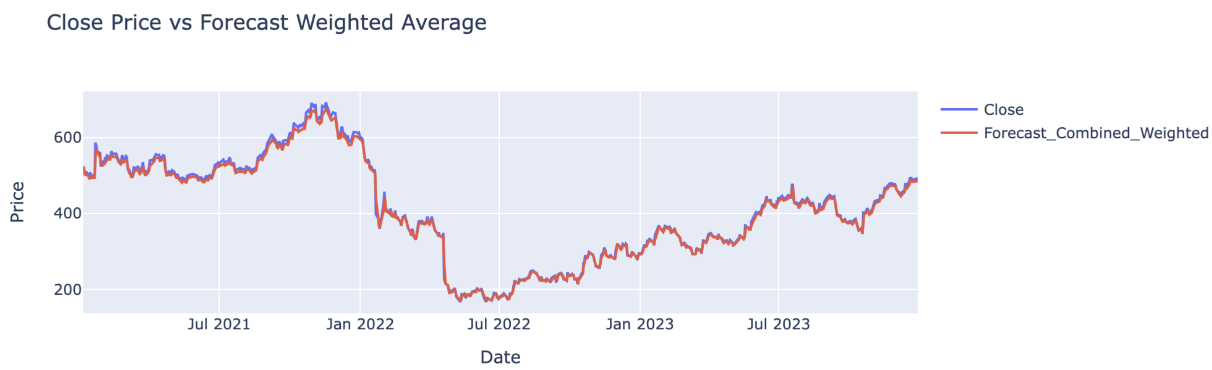

Figure 14.

In figure 15 we can see all of the RMSE scores amongst the different models. The best performing model was the weighted average model. Since the LSTM model was given more weight it allowed the model to perform better since the LSTM model performed better than the ARIMA(1,1,1) model.

	Model	Root Mean Squared Error
3	Weighted	7.516141
2	Simple	7.662436
1	LSTM	9.634213
0	ARIMA	11.731414

Figure 15.

The next results that will be presented are the results for the models created using the Relative Strength Index. The first model created using the RSI feature is the ARIMA(1,1,0) model. The RMSE score for this model is 6.677. The results are shown in figure 16.

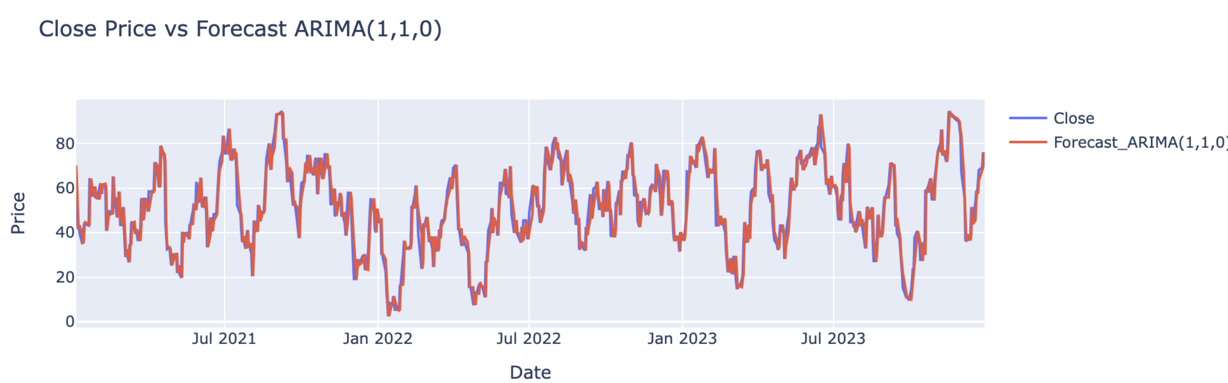

Figure 16.

The LSTM model using the RSI performed much better than the ARIMA(1,1,0) model. It received a RMSE score of 1.33. The results are shown in figure 17.

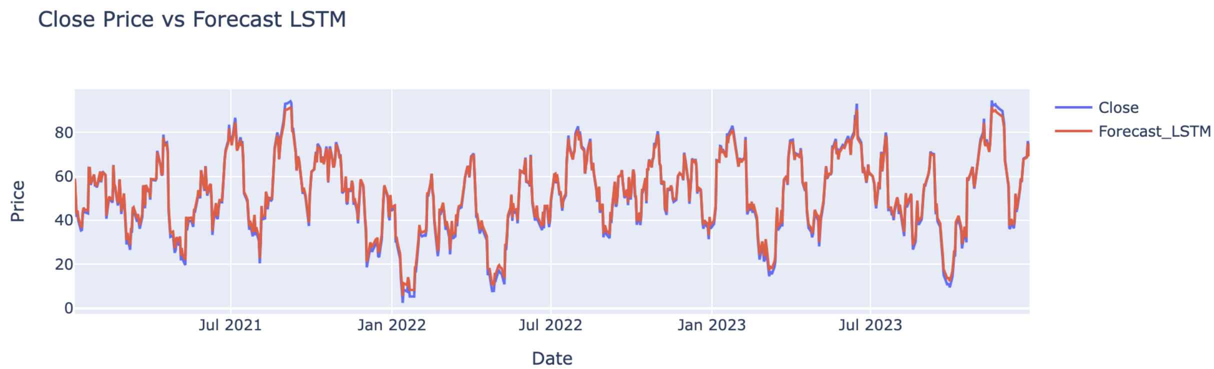

Figure 17.

The simple average model using the RSI received a RMSE score of 3.506. The results are presented in figure 18.

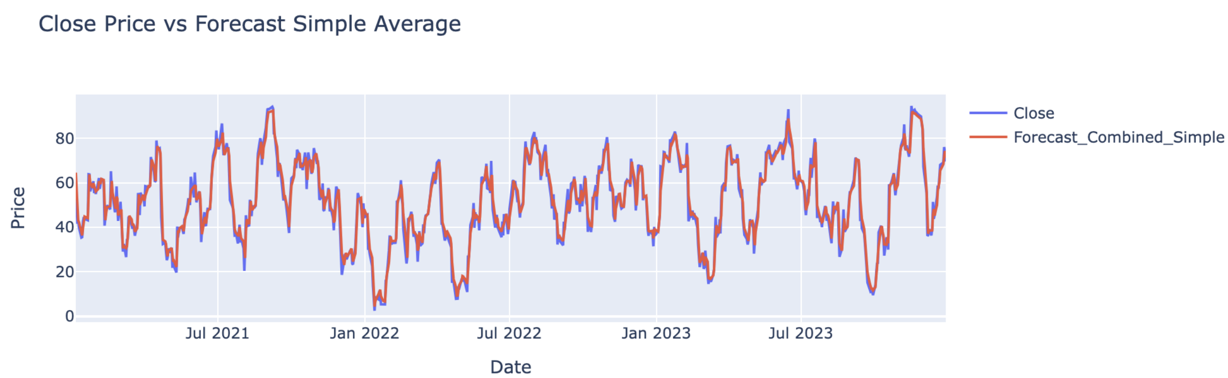

Figure 18.

Surprisingly, the weighted average model did not receive the best RMSE score. Its score is 2.906. I believe these results are because the LSTM performed significantly better than the ARIMA(1,1,0) that the ARIMA predictions depreciated the values used in the weighted average model. The results are shown in figure 19.

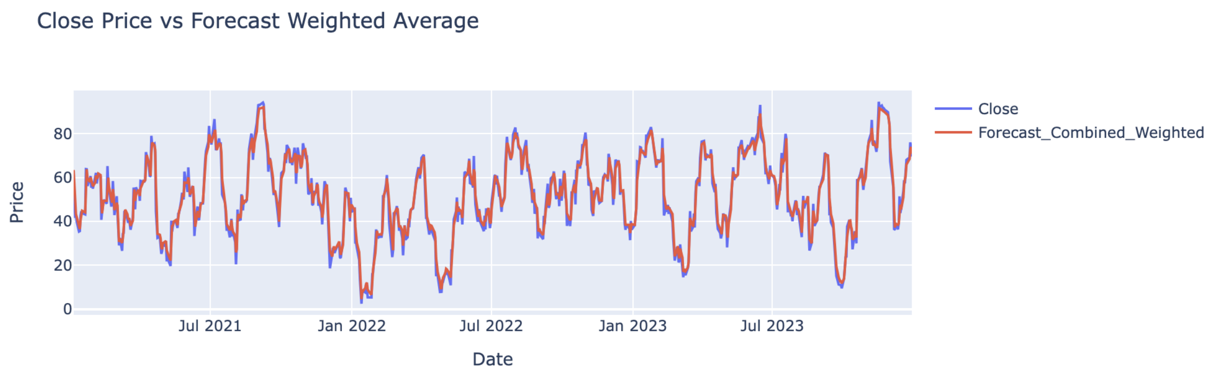

Figure 19.

Figure 20 shows a summary of all the RMSE scores for each model. The best performing model was the LSTM model.

	Model	Root Mean Squared Error
1	LSTM	1.333306
3	Weighted	2.906103
2	Simple	3.505820
0	ARIMA	6.677419

Figure 20.

The final model that was created is the GARCH model used to predict volatility. Figure 21 shows the significant parameters of the GARCH model. The alpha of a stock indicates how well or poorly it has performed in comparison to a benchmark index. The bigger it is the stronger the immediate volatility impact is. Beta indicates how volatile a stock's price has been in comparison to the market as a whole. The bigger it is the longer the volatility impact is.

```
alpha[1]     0.012193
beta[1]      0.982261
```

Figure 21.

Using the GARCH model we can train a rolling prediction of the volatility. In figure 21, we can view the results. After creating the predictions we can compare the predicted volatility to the daily returns in order to see how well our model's predictions are. In figure 22, we can see that the predicted volatility is generally inline with the daily returns. Daily returns are high in areas where volatility is expected to be high.

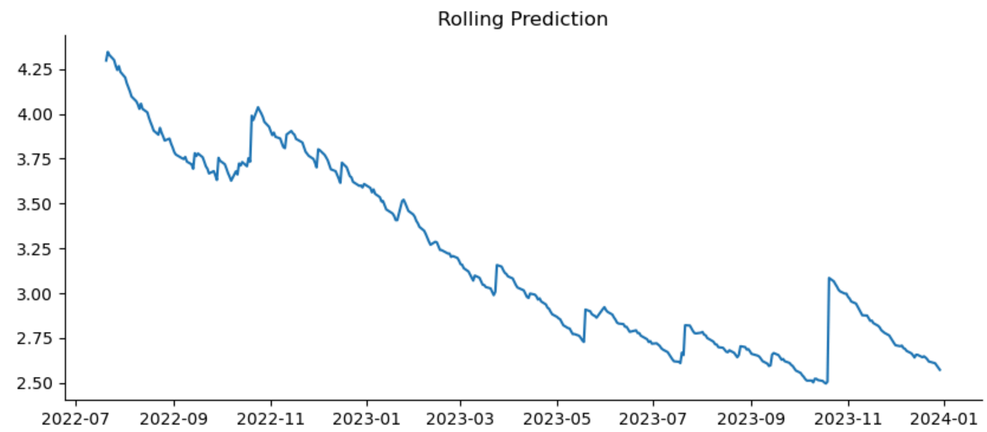

Figure 21.

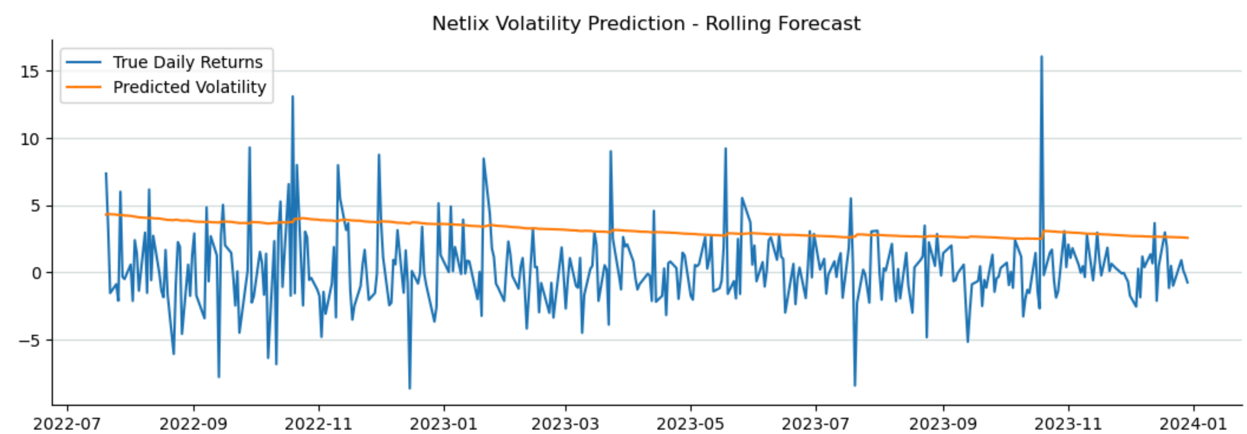

Figure 22.

Discussion:

To summarize, the best performing model for predicting the close price was the weighted average model. I believe this may be because the ARIMA(1,1,1) and LSTM both had RMSE scores that were not too far off from each other. Combining the predictions from both models led to a better performing model. The model which performed the best for the RSI was the LSTM model at a RMSE score of 1.33. I believe this model performed better than the weighted average model because of how different the RMSE scores were between the ARIMA(1,1,0) and LSTM models. The ARIMA(1,1,0) caused the weighted average to produce underwhelming results. Creating the perfect model is almost impossible especially with stock market data. There are many economic factors that can occur. The GARCH model showed to produce accurate results. We can see that daily returns are high when volatility is expected to be high.

Conclusion:

In conclusion, these analyses were to predict Netflix stock market prices. Using features such as close price, relative strength index, and daily returns. These models can help us forecast stock market prices and volatility that can be used to benefit investors, traders, and even the economy. Of course, being able to perfectly predict the stock market is basically impossible. However, we can create models that can be extremely accurate by using multiple features and models. Using other types of algorithms such as Linear Regression, Random Forest, or XGBoost can lead to the creation of better models. To further my research using other company stock features would be a good way of seeing how the models perform using other data instead of Netflix data.

References:

1. "Netflix Inc Share Price, NFLX Stock Price Quote Today." groww.in/us-stocks/nflx.
2. Gratton, Peter. "What Is the Stock Market and How Does It Work?" *Investopedia*, Investopedia, www.investopedia.com/terms/s/stockmarket.asp#:~:text=The%20stock%20market%20is%20also,the%20retirement%20you%20might%20plan