Assessing Time Series Models in Predicting Stock Market Trends

Jae Yun
University of California: Santa Barbara
jaewoong@ucsb.edu

I. Introduction

The field of long-term stock market analysis and predictive modeling presents a compelling landscape for investigation, particularly within the technology sector where entities such as Amazon, Microsoft, Nvidia, and Google have established dominance. These corporations not only boast substantial market capitalization but have also exhibited pronounced growth over the last decade, outpacing other industries and securing their position at the forefront of the economic arena. However, creating a predictive model that accurately captures the stock market's future prices is not a simple endeavor. It requires not only a thorough understanding of the dynamics of the market, but also a high dependency on market sentiment, financial domain knowledge, external influences, and various uncertainties that drive the market.

This study selects these four technology giants as the focal point, driven by a desire to critically evaluate the performance of some of the largest companies within a homogeneous sector. By comparing stocks that are within the same industry, the analysis aims to discern whether model effectiveness varies significantly with the variance in individual stock returns, regardless of non-dynamic factors. This approach facilitates an in-depth understanding of how well the model captures sector-specific dynamics and whether its predictive accuracy is consistent across different yet closely related stocks. Essentially, the goal is to determine the model's robustness in forecasting within a singular sector, providing insights into its utility and reliability in predicting stock trends in the technology industry.

The objective of this analysis is to apply time series forecasting models to the stock prices of Amazon, Microsoft, Nvidia, and Google, spanning a decade of 552 weekly opening price observations from January 1st, 2010, to January 1st, 2020. The focus is to evaluate the efficacy of these models, utilizing only the market trends and seasonal patterns, and intentionally excluding external, macroeconomic events such as the COVID-19 pandemic, Ukrainian-Russian War, The Great Recession, and other major economic crises which could skew the data and compromise the integrity of the technical-only focused forecast.

The rationale for this approach stems from a desire to ascertain the intrinsic predictability of stock prices based on historical data alone, thus evaluating the robustness of time series models as a tool for financial forecasting. By delineating the influence of major unpredictable occurrences, the study aims to distill the efficacy of time series analysis in capturing the underlying patterns and tendencies inherent to the stock prices of these leading technology firms.

To this end, study will transform our data to a stationary process in order to determine the parameters to employ a time series forecasting model, called the SARIMA (Seasonal ARIMA) model, to project future stock prices of Amazon, Microsoft, Nvidia, and Google. The effectiveness of these models will be gauged through the data from 2010 to 2018 to forecast stock prices for the year 2019. The predictive accuracy will be evaluated based on how closely the forecasted values align with the actual stock prices observed in 2019, thus providing a measure of the models' performance using AIC (Akaike Information Criterion) and visual representations of confidence intervals to assess predictive accuracy.

This analysis seeks to not only test the validity of time series forecasting in a stock market context but also to explore the potential of these models as a significant component of a strategic investment framework. The study intends to evaluate how well time series models can perform in a generally volatile stock market setting.

II. Key Terms

Augmented Dickey-Fuller Test (ADF Test)

The ADF Test is a common statistical test to determine if a time series is stationary. This is an important assumption, as SARIMA models must assume that a time series is stationary in order to accurately estimate the model parameters and provide reliable forecasts. A time series is considered stationary if its statistical properties, such as mean, variance, and autocorrelation, are constant over time. The ADF test is given by the expression[1]

$$\Delta y_t = \alpha + \beta t + \gamma y_{t-1} + \delta_1 \Delta y_{t-1} + \cdots + \delta_{p-1} \Delta y_{t-p+1} + \varepsilon_t$$

where Δy_t is the first difference of the time series at time t, α is a constant term, β is the coefficient of a linear time trend, and γ is the coefficient of the lagged value y_{t-1}. The terms δ_i are the coefficients of the lagged first differences, up to a maximum lag of $p-1$. The error term ε_t is assumed to be white noise.

Kwiatkowski-Phillips-Schmidt-Shin (KPSS) Test

Much like the Augmented Dickey-Fuller (ADF) Test, the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test is a statistical method used to analyze the stationarity of a time series. While the ADF test focuses on identifying a unit root indicating non-stationarity, the KPSS test takes the opposite approach by testing the null hypothesis that the time series is stationary around a deterministic trend. It's an essential tool for validating the stationarity of a series before fitting models like SARIMA, which require a stationary input.

The KPSS test checks for stationarity by assessing the presence of a stochastic trend, and is formulated as follows[^2]:

$$\text{KPSS} = \frac{S_T^2}{T\hat{\sigma}^2}$$

where: - S_T^2 is the sum of squared deviations of the cumulative sum of residuals (or deviations from the trend), - T is the total number of observations, - $\hat{\sigma}^2$ is an estimate of the variance of the residuals.

In contrast to the ADF test, which subtracts lagged values of the time series to find evidence of non-stationarity, the KPSS test sums the squared deviations to test the strength of the trend component. A small p-value in the ADF test suggests non-stationarity, whereas in the KPSS test, a small p-value indicates stationarity. In practice, both tests are used in conjunction to provide a more robust assessment of the time series stationarity.

Box-Cox Transformation

The Box-Cox transformation is a statistical technique employed to normalize the distribution of a time series and stabilize variance, particularly when the data exhibits non-constant variance (heteroscedasticity). It can also help in making the residuals of a model more homoscedastic and normally distributed, which is an important step in meeting the stationarity requirements necessary for accurate time series forecasting. The transformation is particularly beneficial when the assumption of equal variance is violated. The Box-Cox transformation is defined as follows:

For a time series y_t and a transformation parameter λ, the Box-Cox transformed series y_t' is computed using:

$$y_t' = \begin{cases} \frac{y_t^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0, \\ \log(y_t) & \text{if } \lambda = 0. \end{cases}$$

[1] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics, 4th ed.

Here: - y_t is the observed value at time t. - λ is the Box-Cox transformation parameter, which is determined empirically to best stabilize the variance of the time series. - y'_t represents the transformed value at time t.

The parameter λ is typically estimated to maximize the likelihood function, and it can vary across a wide range. When $\lambda = 1$, the transformation is equivalent to no transformation (the identity function). When $\lambda = 0$, the Box-Cox transformation corresponds to taking the natural logarithm of y_t. The goal is to find the value of λ that results in a time series with constant variance and approximately normal residuals.

Once transformed, the series should be assessed for improved homoscedasticity and normality. Subsequent time series analyses, including ARIMA modeling, are conducted on the transformed data. Predictions or forecasts obtained from the model should be back-transformed to the original scale, which can be done using the inverse Box-Cox transformation.

Kwiatkowski-Phillips-Schmidt-Shin (KPSS) Test

The Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test, is a statistical procedure used to assess the stationarity of a time series. Unlike tests that check for the presence of a unit root, the KPSS test examines the null hypothesis that a time series is stationary around a deterministic trend (level stationarity) or around a stochastic trend (trend stationarity). This test is particularly useful when a time series exhibits fluctuations around a drifting mean or an evolving trend. The KPSS test is defined as:

For a time series y_t, the KPSS statistic is computed as:

$$\text{KPSS} = \frac{\sum_{t=1}^{T} \hat{S}_t^2}{T^2 \hat{\sigma}^2}$$

where: - T is the number of observations in the time series. - $\hat{S}_t$ is the partial sum of the residuals (or deviations from the mean for level stationarity, or from the trend for trend stationarity), up to time t. - $\hat{\sigma}^2$ is an estimate of the long-run variance of the time series.

The null hypothesis (H_0) of the KPSS test is that the time series is stationary. A small p-value (typically less than 0.05) indicates that the null hypothesis can be rejected, suggesting the presence of a unit root, or in other words, that the time series is non-stationary. A large p-value suggests that the time series is stationary.

Time Series Decomposition

Decomposition is a method used in time series analysis to break down a time series (Y(t)) into several components, typically including trend, seasonal, and random or residual components. The random/residual component may result in a stationary time series. This breakdown helps in understanding, analyzing, and forecasting the data more effectively. The formula is as follows[2]

$$Y(t) = T(t) + S(t) + R(t)$$

- $Y(t)$ is the observed value of the time series at time t, given by $T(t) + S(t) + R(t)$.
- $T(t)$ is the trend component of the time series at time t, represented as $T_t = f(t)$.
- $S(t)$ is the seasonal component of the time series at time t, calculated as $S_t = \sum_{i=1}^{n} s_i$.
- $R(t)$ is the residual or random component of the time series at time t, defined as $R_t = Y_t - T_t - S_t$.

[2] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics, 4th ed.

Autocorrelation function (ACF)

The Autocorrelation Function (ACF) is a fundamental tool in time series analysis that measures the linear relationship between an observation in a time series and the observations at previous times (lags). By looking at the ACF plot, you can determine how many past values (lags) in the time series significantly influence the current value. This is crucial for identifying the 'Moving Average' (MA) component in ARIMA models.

The ACF is given by the expression[3]

$$\mathrm{ACF}(k) = \frac{\sum_{t=k+1}^{n}(Y_t - \bar{Y})(Y_{t-k} - \bar{Y})}{\sum_{t=1}^{n}(Y_t - \bar{Y})^2}$$

where k is the lag, n is the number of observations in the time series, Y_t is the observed value at time t, and $\bar{Y}$ is the sample mean of the time series.

Partial Autocorrelation Function (PACF)

The Partial Autocorrelation Function (PACF) offers a measure of the correlation between observations in a time series separated by various time lags, while controlling for the correlations at shorter lags. PACF is key in identifying the 'Autoregressive' (AR) component in ARIMA models. It indicates the extent of the direct relationship between an observation and its lag.

The PACF is expressed as[4]

$$\rho_k = \frac{\mathrm{cov}(Y_t, Y_{t-k})}{\mathrm{var}(Y_t)} - \sum_{j=1}^{k-1} \phi_j \rho_{k-j}$$

where ρ_k is the partial autocorrelation at lag k, $\mathrm{cov}(Y_t, Y_{t-k})$ is the covariance between Y_t and Y_{t-k}, $\mathrm{var}(Y_t)$ is the variance of Y_t, ϕ_j is the estimated coefficient of the autoregressive (AR) model at lag j, and k is the maximum lag for which the PACF is computed.

SARIMA Model

The SARIMA (Seasonal Autoregressive Integrated Moving Average) model is an advanced form of the ARIMA model. It's specifically designed to better model and forecast time series data that show seasonality – regular fluctuations or patterns that repeat over time. Seasonality can be observed in many real-world data such as monthly sales, daily temperature readings, weekly website traffic, or in our case – the stock market.

The SARIMA Model is formulated as follows[5]

$$\begin{aligned}(1 - \phi_1 B - \cdots - \phi_p B^p)(1 - \Phi_1 B^s - \cdots - \Phi_P B^{sP})^d (y_t - \mu)\\ = (1 + \theta_1 B + \cdots + \theta_q B^q)(1 + \Theta_1 B^s + \cdots + \Theta_Q B^{sQ})\varepsilon_t\end{aligned}$$

where y_t is the observed value of the time series at time t, μ is the mean of the time series, ε_t is the error term at time t, and B is the backshift operator. The parameters p, d, and q correspond to the order of the autoregressive (AR), integrated (I), and moving average (MA) components, respectively. The parameters P,

[3] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics, 4th ed.

[4] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics, 4th ed.

[5] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics, 4th ed.

D, and Q correspond to the order of the seasonal AR, seasonal I, and seasonal MA components, respectively. The parameter s represents the length of the seasonal cycle.

The ϕ and θ parameters are the coefficients of the AR and MA terms, respectively, while the Φ and Θ parameters are the coefficients of the seasonal AR and seasonal MA terms

Ljung-Box Test

The Ljung-Box Test is widely used type of statistical test primarily used to determine the absence of autocorrelation at multiple lag lengths in residuals from a time series analysis. The essence of the test is to verify the randomness of a time series, which is a crucial assumption in time series modeling, and assessing the SARIMA model's adequacy.

The Ljung-Box Test is expressed as[6]

$$Q = n(n+2)\sum_{k=1}^{h}\frac{\hat{\rho}_k^2}{n-k}$$

where Q is the test statistic, n is the sample size, h is the number of lags being tested, and $\hat{\rho}_k$ is the sample autocorrelation at lag k. The null hypothesis of the test is that the autocorrelations up to lag h are equal to zero, indicating that the time series is a white noise process. The alternative hypothesis states that the autocorrelations are not equal to zero, indicating the presence of serial correlation in the data. The test statistic Q follows a chi-squared distribution with $h - p$ degrees of freedom, where p is the number of parameters estimated in the time series model.

Box-Pierce Test

The Box-Pierce Test is another well-known statistical test used to check the independence of residuals in a time series model. It is similar to the Ljung-Box Test but predates it and is based on a simpler formulation. The primary purpose of this test is to detect any autocorrelation in the residuals of a fitted time series model, thereby assessing the SARIMA model's adequacy.

The Box-Pierce is formulated as[7]

$$Q^* = n\sum_{k=1}^{h}\hat{\rho}_k^2$$

where Q^* is the test statistic, n is the sample size, h is the number of lags being tested, and $\hat{\rho}_k$ is the sample autocorrelation at lag k. The null hypothesis of the test is that the autocorrelations up to lag h are equal to zero, indicating that the time series is a white noise process. The alternative hypothesis states that the autocorrelations are not equal to zero, indicating the presence of serial correlation in the data.

The test statistic Q^* follows an approximately chi-squared distribution with $h - p$ degrees of freedom, where p is the number of parameters estimated in the time series model.

III. Data

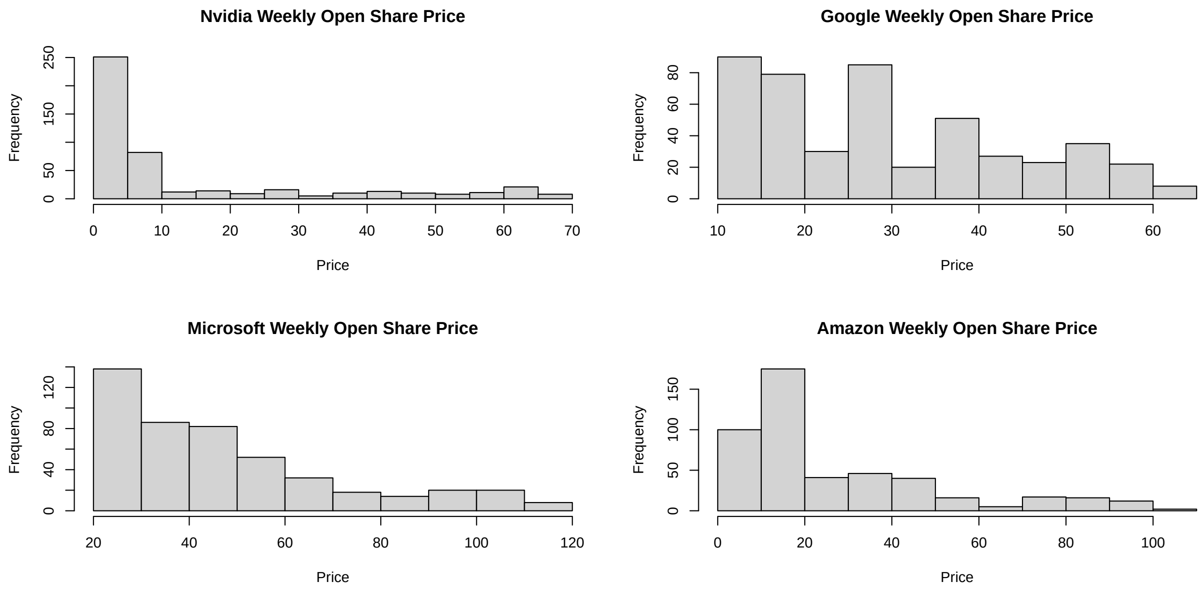
In our time series dataset, we observe five key dimensions, commonly referred to as the 5V's, that significantly enhance the depth and utility of our analysis:

[6] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics, 4th ed.

[7] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics, 4th ed.

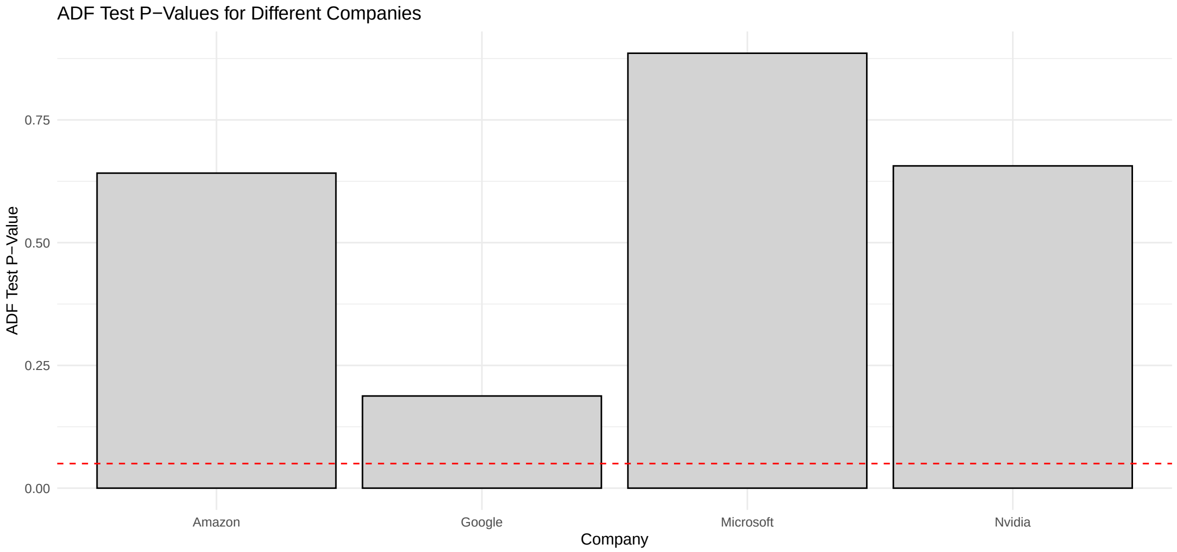
- **Volume**: Our dataset encompasses a substantial volume of data, spanning ten years of weekly stock market information. This extensive collection of data lays a solid foundation for making precise and effective predictions.
- **Velocity**: The dataset's velocity is characterized by its weekly frequency, providing a steady and consistent flow of information for timely analysis.
- **Variety**: Despite focusing on just four major stocks within the technology sector, our dataset is rich in variety. Each stock represents distinct market behaviors and price patterns, unique to each company, offering a diverse perspective.
- **Veracity**: The data, sourced from Yahoo Finance, is marked by its high level of veracity. Reflecting actual stock market prices, this dataset offers exceptional accuracy, crucial for reliable analysis.
- **Value**: Analyzing this data allows us to identify market trends and perform in-depth analysis. Such insights are invaluable not only for understanding past market behaviors but also for forecasting future trends. This is particularly significant given the influential nature of these technology stocks in the broader economic context.

Now, we can begin using R to import our data from yahoo, read csv, and plot the time series data:



Looking at the trends of each weekly opening price for our 4 technology stocks, we see a steady increase from 2010 to 2016, and a sharp, yet fluctuating increase from 2016 to 2020.

These plots show the presence of non-stationarity and non-randomness within our datasets and thus, we must perform differencing or decomposition to accurately utilize our forecasting model. We can also construct histograms to better visualize the distribution of our current non-stationary dataset:



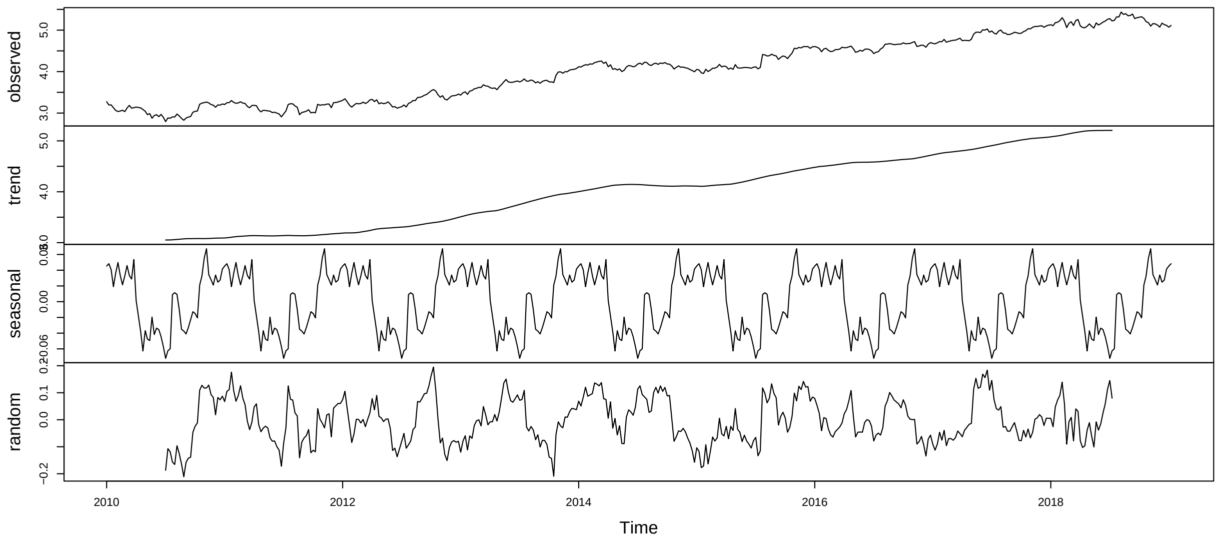
Using the ADF test, we extract the p-value for each dataset. Each p-value exceeds the significance value represented as the red dotted line where p = 0.05. Consequently, we fail to reject the null hypothesis and conclude that we do not have sufficient evidence to determine that each data is non-stationary.

IV. Data Augmentation

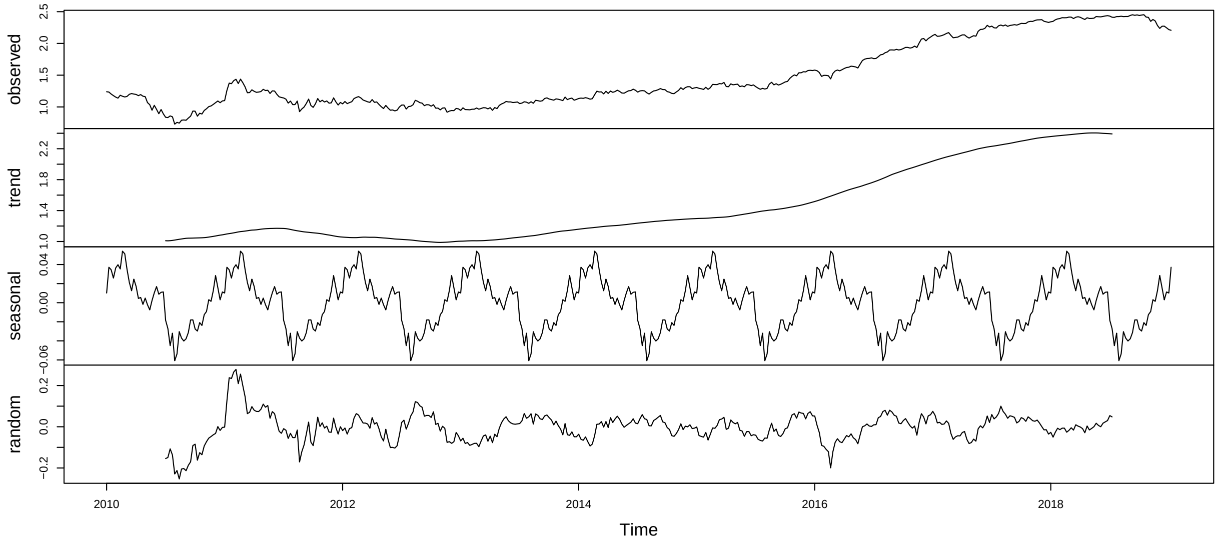
Through iterative examination during our study, it became evident that decomposition alone was insufficient for rendering the random component of dataset, and thus we first transformed the dataset using the Box-Cox transformation. This transformation helps in mitigating the effects of non-stationarity related to non-constant variance, which is a common characteristic of financial time series data. By doing so, we aim to transform the series into one that has a stable variance and a more symmetrical distribution, thereby creating a better foundation for the subsequent decomposition to isolate and remove the trend and seasonal components effectively.

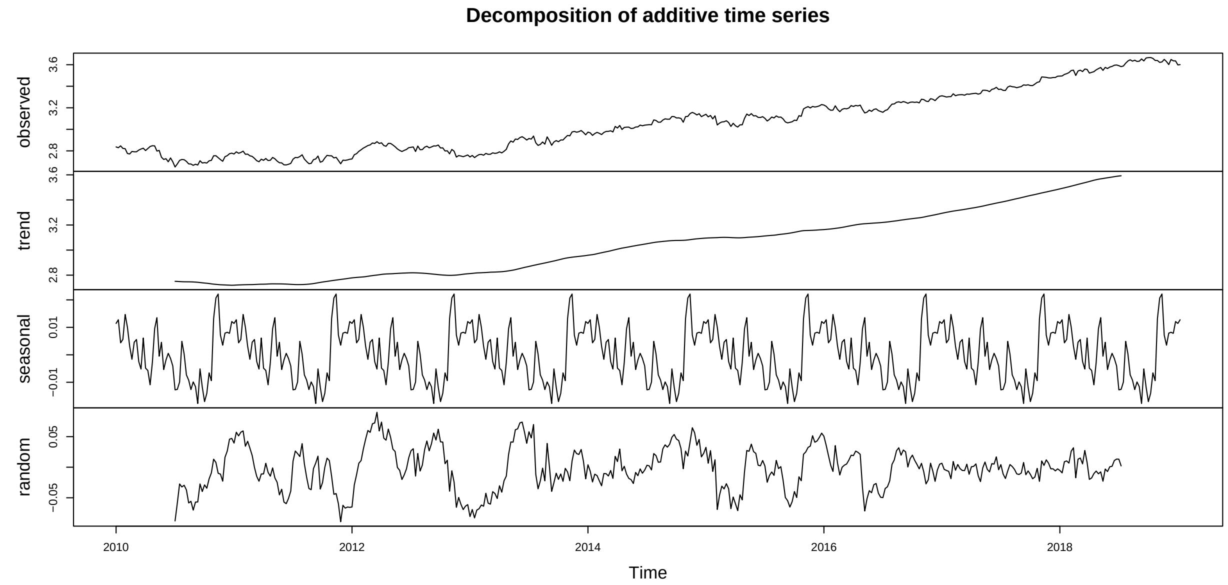
We can then plot a decomposition of the Box-Cox, to see the Box-Cox observed values, and see if there are any trends or seasonality patterns.

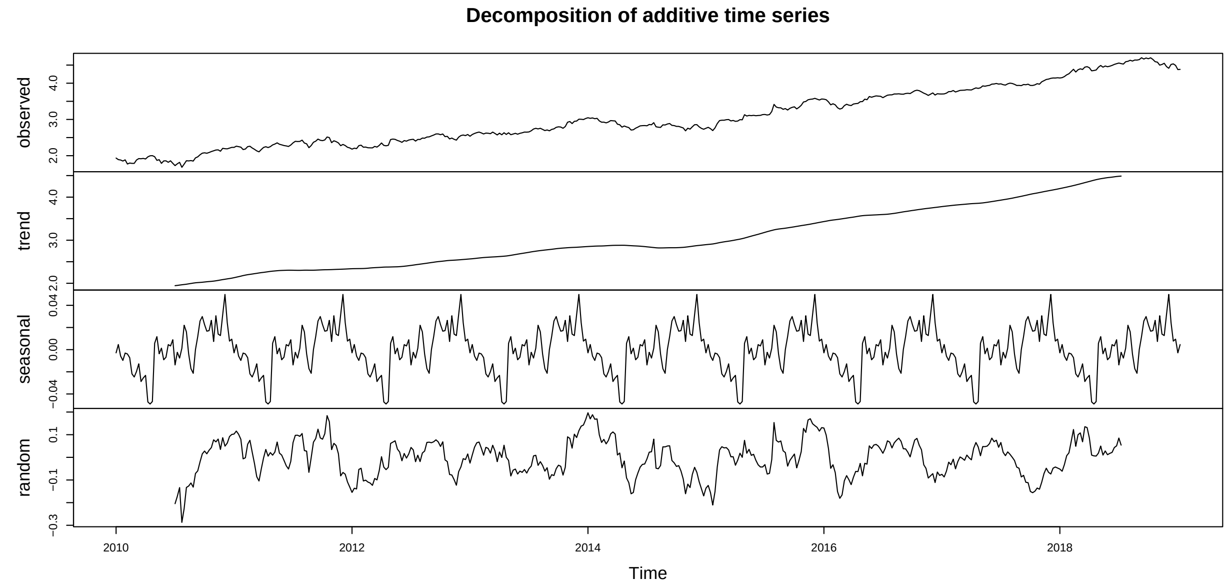
Decomposition of additive time series



Decomposition of additive time series

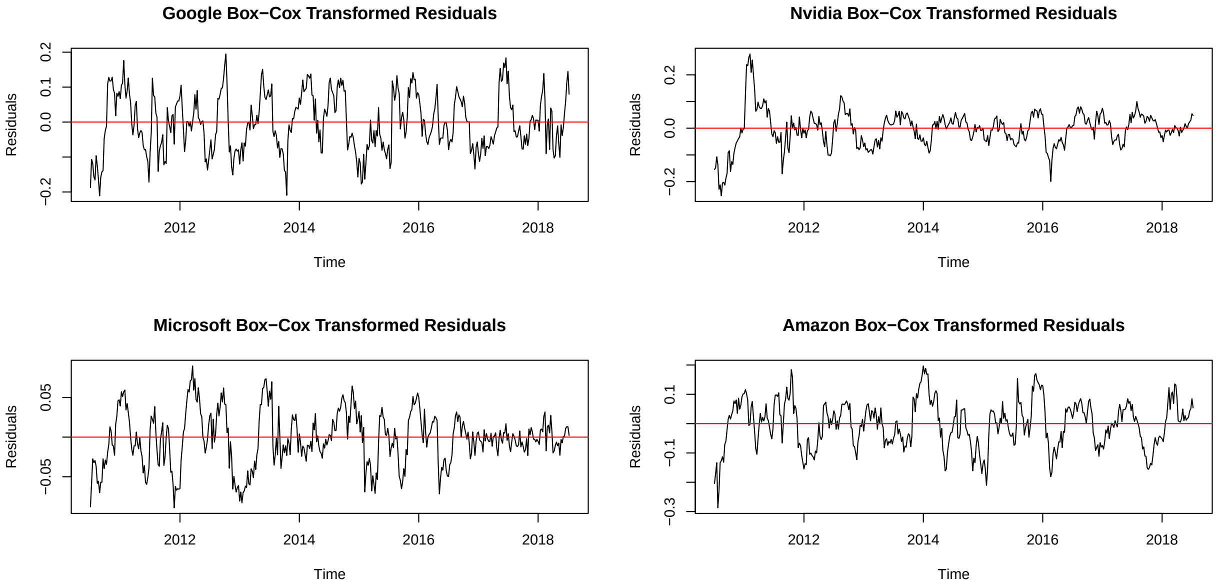




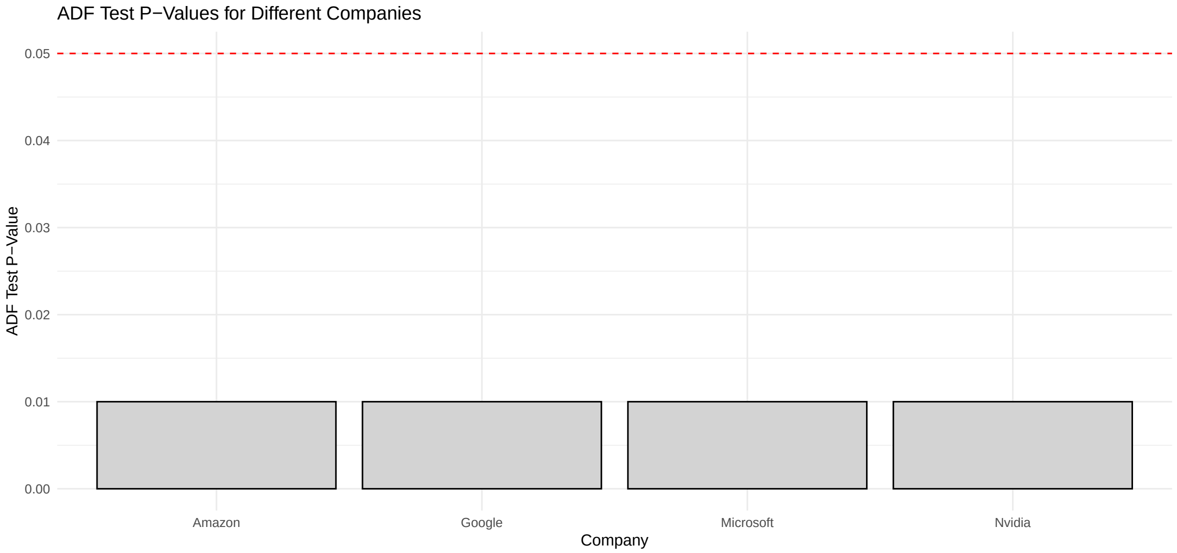


The examination of the random components across all four Box-Cox transformed time series suggests that they closely approximate a white noise distribution. This resemblance indicates that the random components are stationary, providing a solid foundation for the identification of parameters for our SARIMA models with greater confidence. To confirm further, we can perform the ADF test on each decomposed random component of the box-cox transformed data.

Here are our plots for our Box-Cox transformed residual-extracted components to check for stationarity.



It seems we have achieved stationarity within our twice-augmented dataset. We can also test the statistical hypothesis for stationarity using the ADF test.

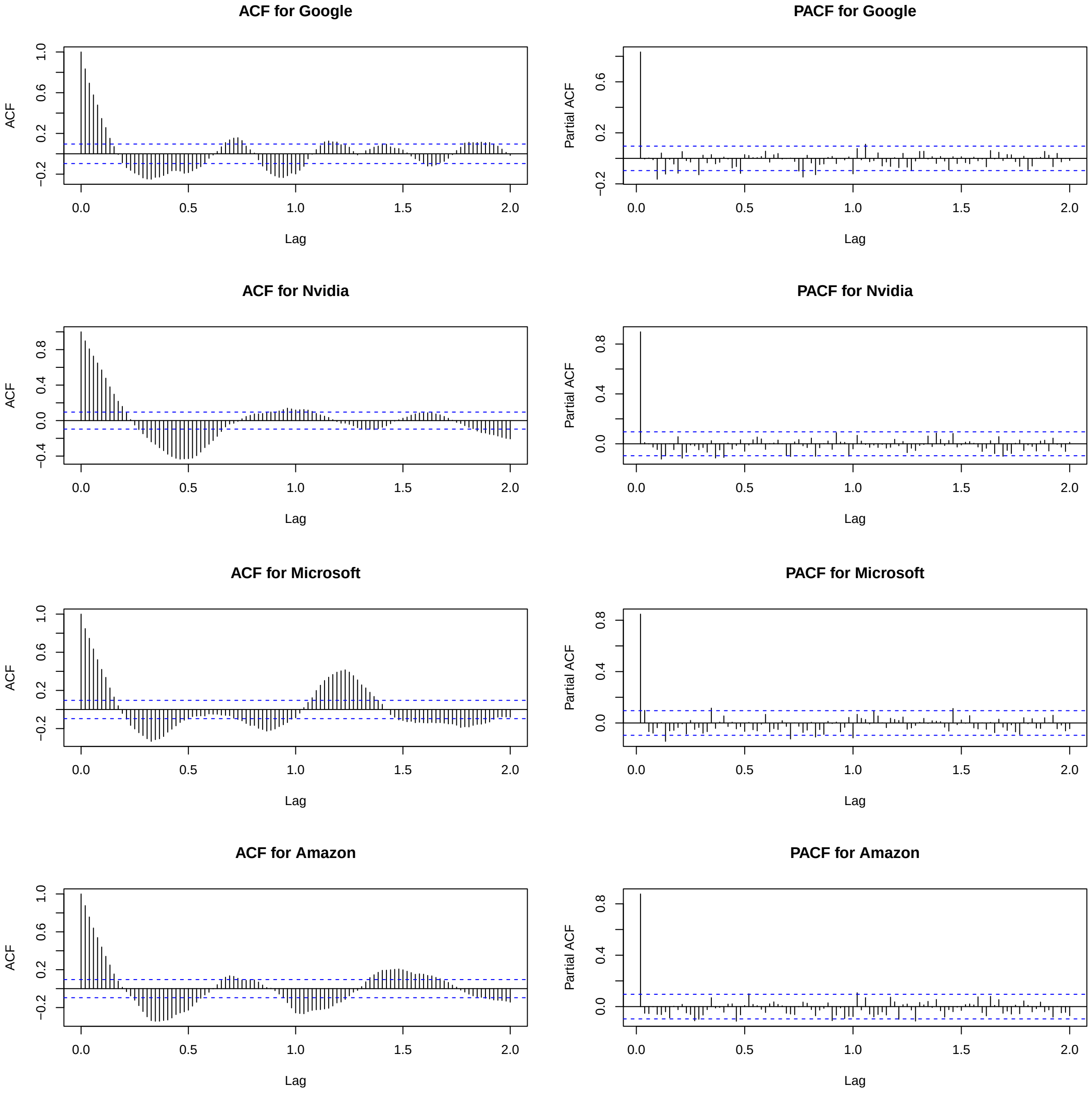


Our ADF Test P-values for each twice-augmented datasets are now below the significance level of 0.05, and thus we reject the null hypothesis for non-stationarity. Therefore, we conclude with significant evidence that each of our time series data are stationary. Consequently, we can move on with the next steps in selecting parameters in our SARIMA model using PACF and ACF plots.

V Models and Parameter Selection

We will employ the SARIMA model to evaluate performance, selecting the optimal model based on the lowest Akaike Information Criterion (AIC) values. Model parameters will be determined through analysis of the ACF and PACF plots.

Considering SARIMA model follows parameters Sarima(p, d, q, P, D, Q), we will use this in our evaluation.



- A significant spike at lag 1 (outside the confidence interval) may suggest an AR(1) component (p = 1) for the non-seasonal part of the model.
- The need for differencing (d) may not be necessary since the tests indicate the series is already stationary (d=0).
- The exact number (q) can be challenging to determine solely from the ACF plot when there are many significant lags.
- The sine-wave like features represent our data still contains a seasonal pattern. Thus, we need to difference it further using seasonal differencing. Thus we will set D = 1, and set P = 0 or 1, and Q = 0 or 1.

Thus, we choose a SARIMA model with parameters SARIMA($p = 1$, $d = 0$, $q = 1$ or 2, $P = 0$ or 1, $D = 1$, $Q = 0$ or 1).

- We also see very similar patterns for the other ACF and PACF plots

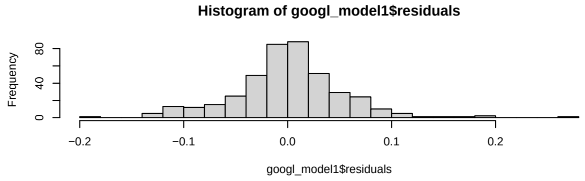
The chosen SARIMA model parameters for the NVIDIA, Microsoft, and Amzon residuals could be SARIMA($p = 1$, $d = 0$, $q = 1$ or 2, $P = 0$ or 1, $D = 1$, $Q = 0$ or 1).

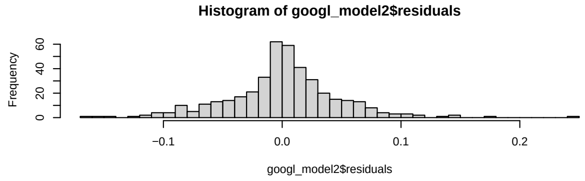
```
##   Google Model 1 AIC Google Model 2 AIC Google Model 3 AIC Nvidia Model 1 AIC
## 1          -1041.696          -1080.486          -1080.745          -1345.771
##   Nvidia Model 2 AIC Nvidia Model 3 AIC
## 1          -1389.588          -1387.605
```

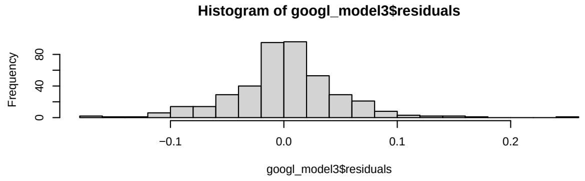
```
##   Microsoft Model 1 AIC Microsoft Model 2 AIC Microsoft Model 3 AIC
## 1             -1710.289             -1759.648             -1758.092
##   Amazon Model 1 AIC Amazon Model 2 AIC Amazon Model 3 AIC
## 1          -1150.074          -1205.664           -1205.35
```

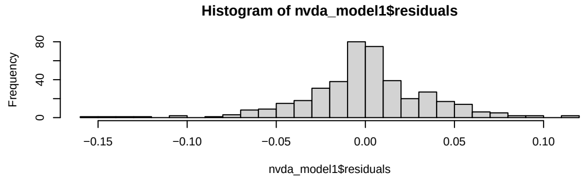
VI. Model Evaluation on Train Data

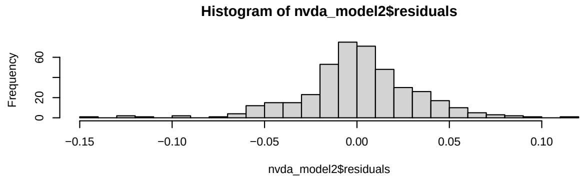
Our residuals should look like a normal white noise to represent an adequate model fit. We can test this using histograms.

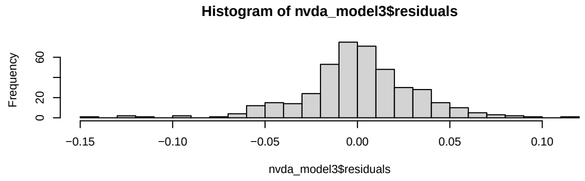


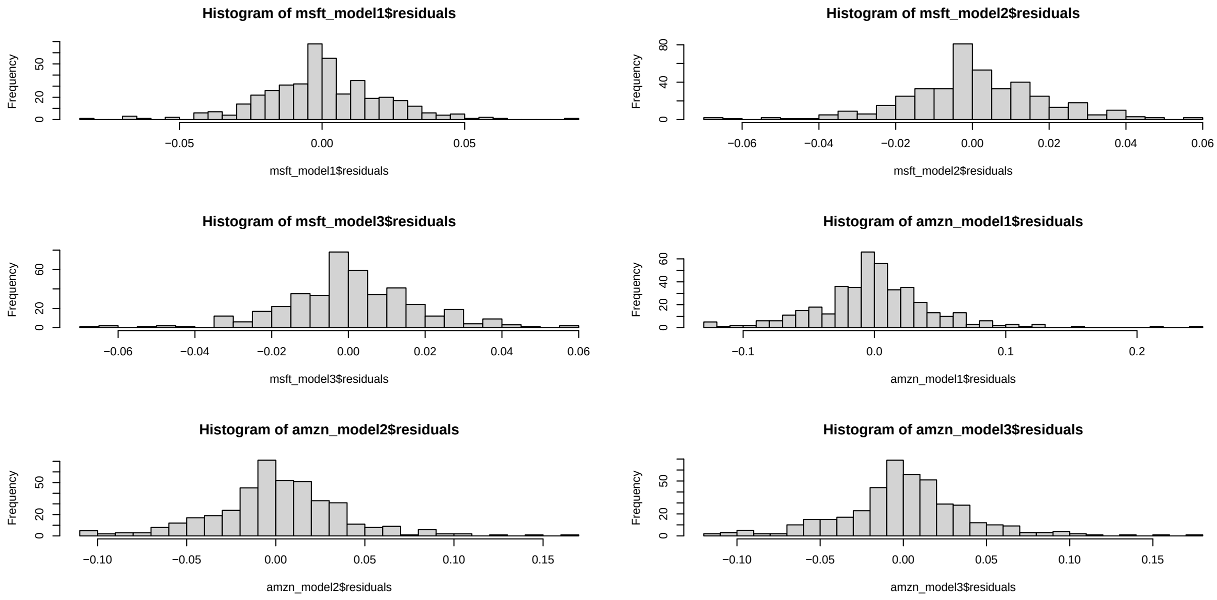






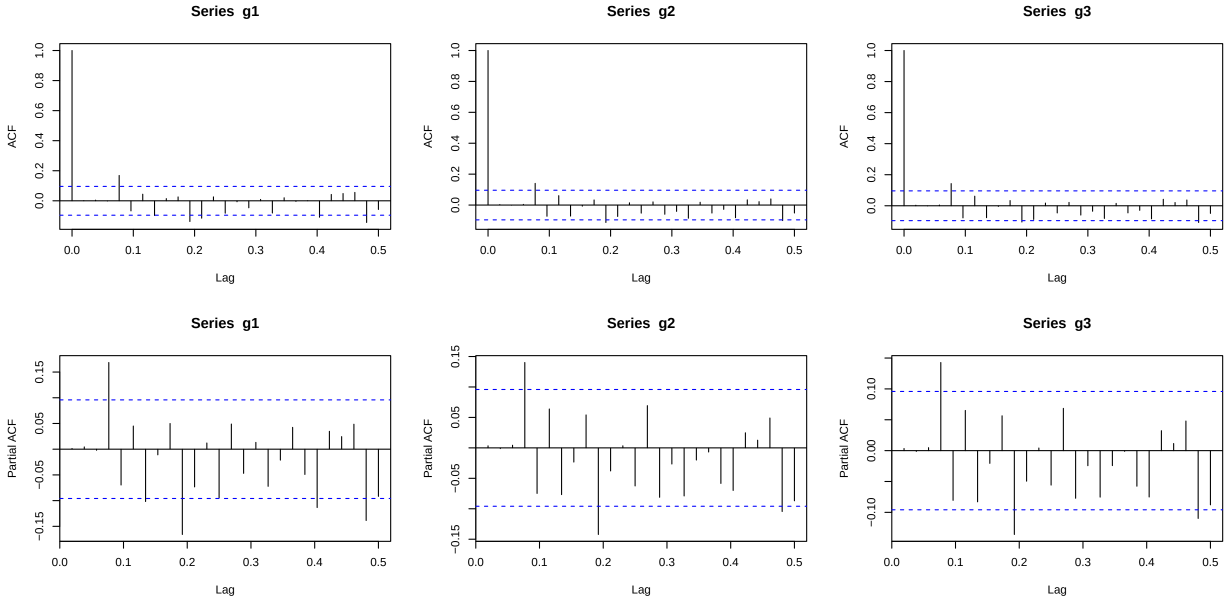




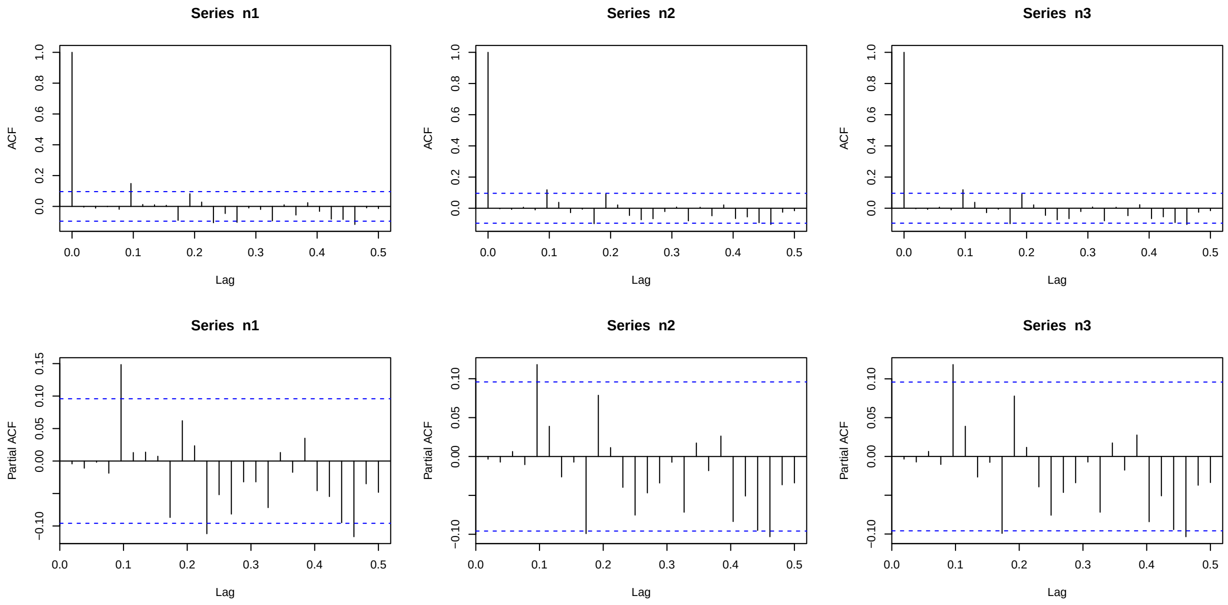


Our histograms look normal and resembles a gaussian white noise, which is a clear indicator that our models performed well.

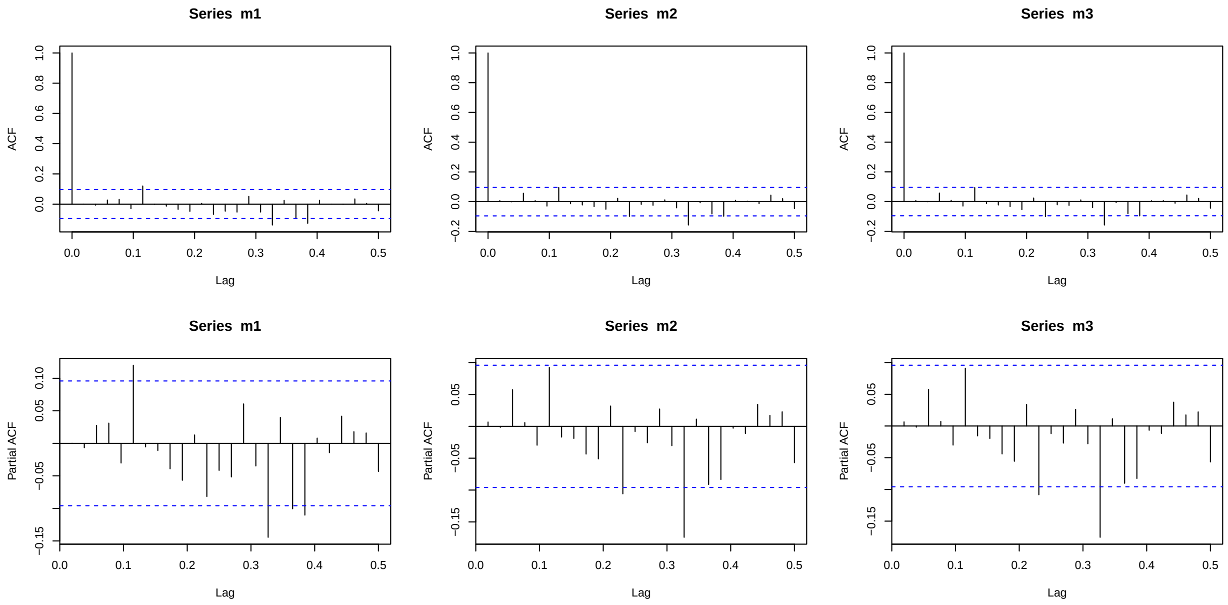
PACFS and ACF of Google model residuals



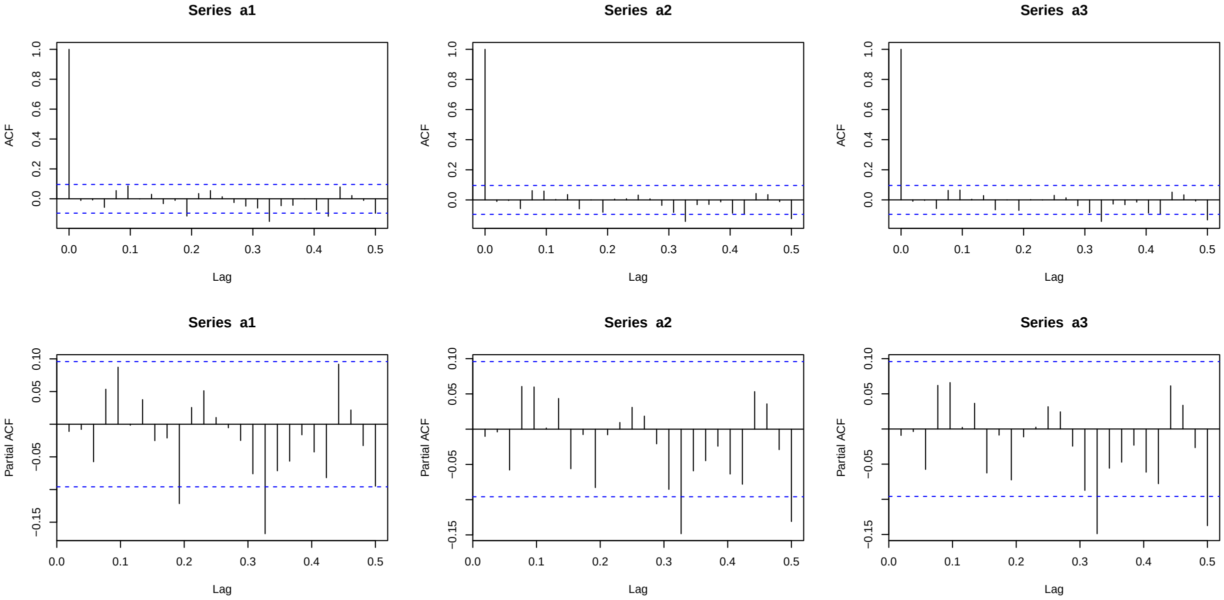
PACFS and ACF of Nvidia model residuals



PACFS and ACF of Microsoft model residuals



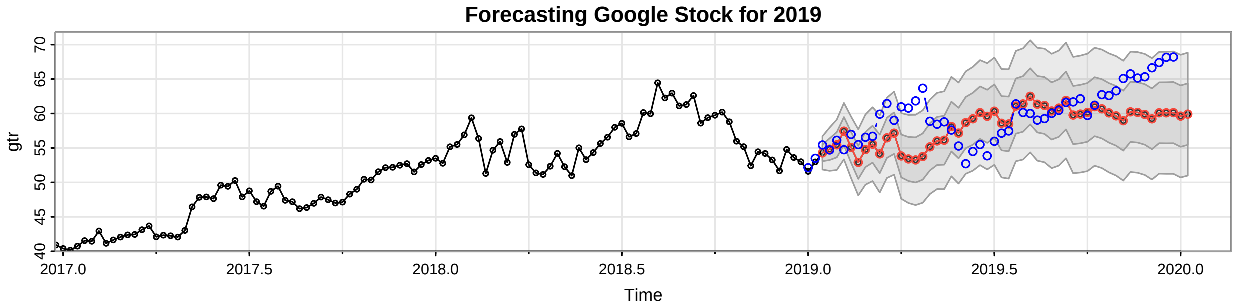
PACFS and ACF of AMAZON model residuals

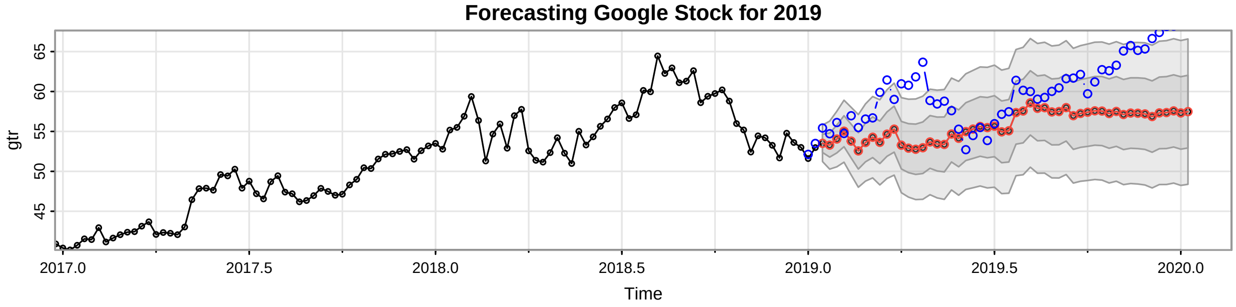


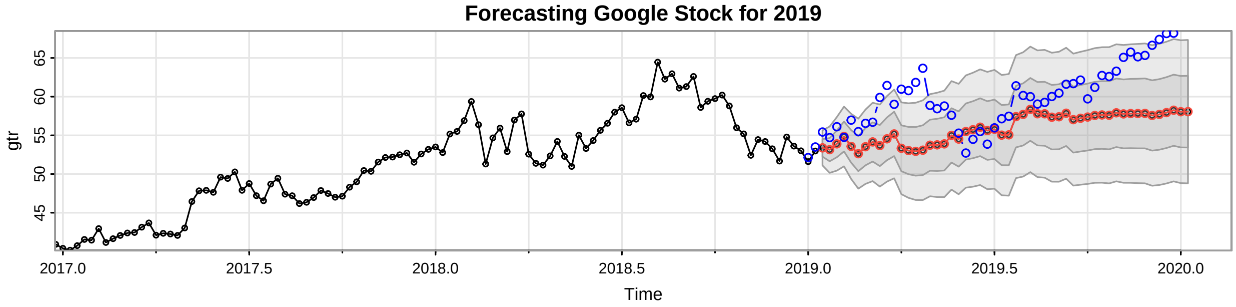
All in all, our pacf and acf charts for all of our stocks do resemble a white noise distribution, and we can continue forecasting our datasets.

VII. Forecasting with Test Data

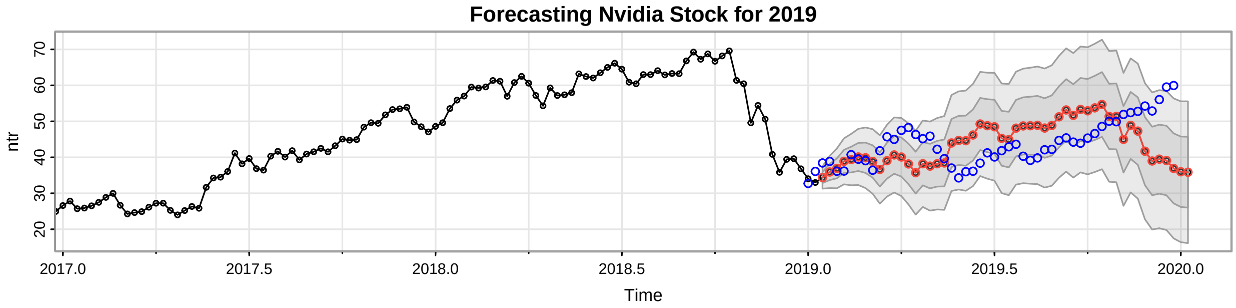
Google Forecast

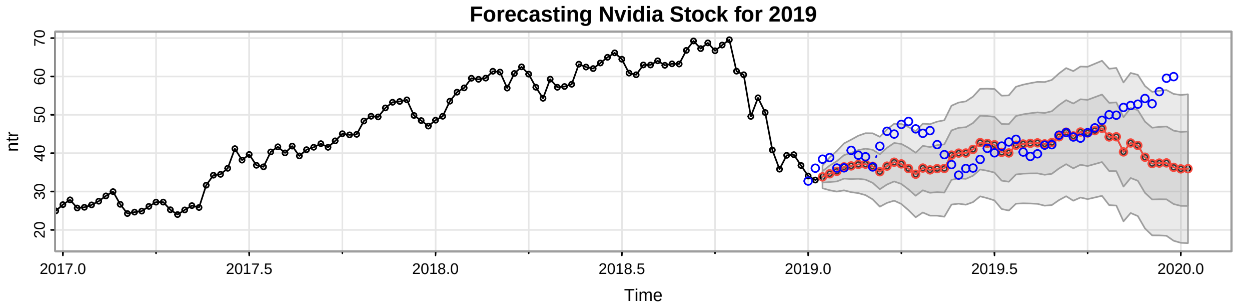

Forecasting Google Stock for 2019

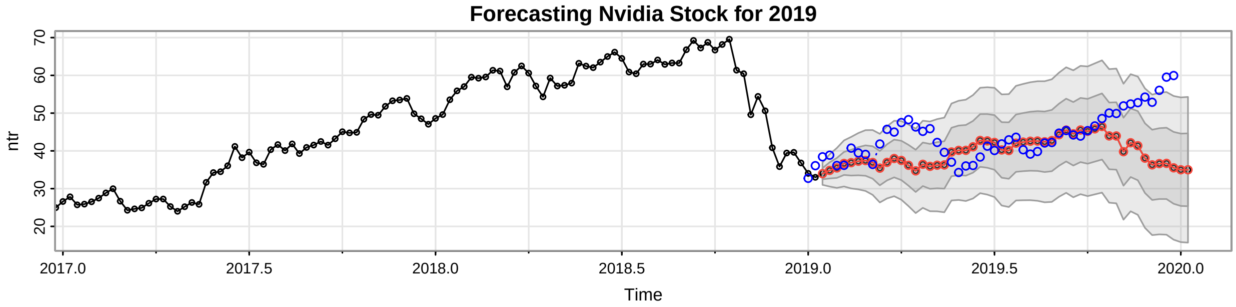

Forecasting Google Stock for 2019


Forecasting Google Stock for 2019

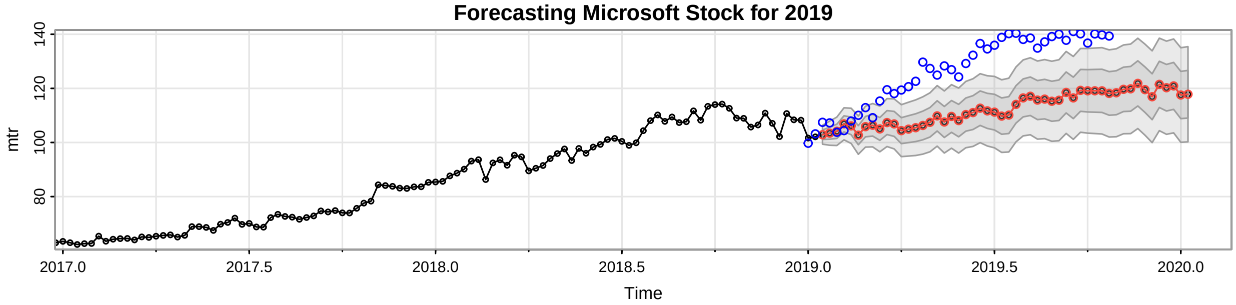
Nvidia Forecast

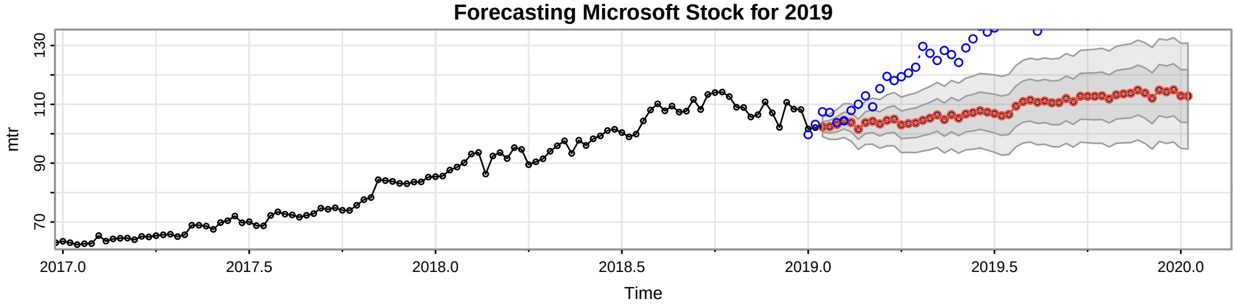


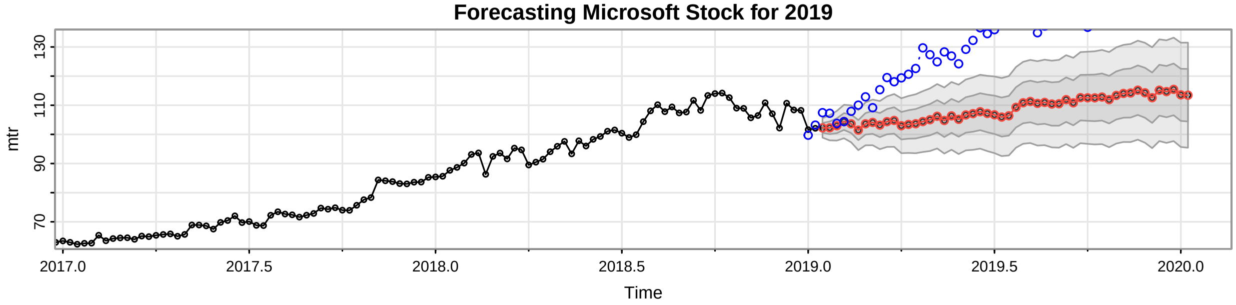




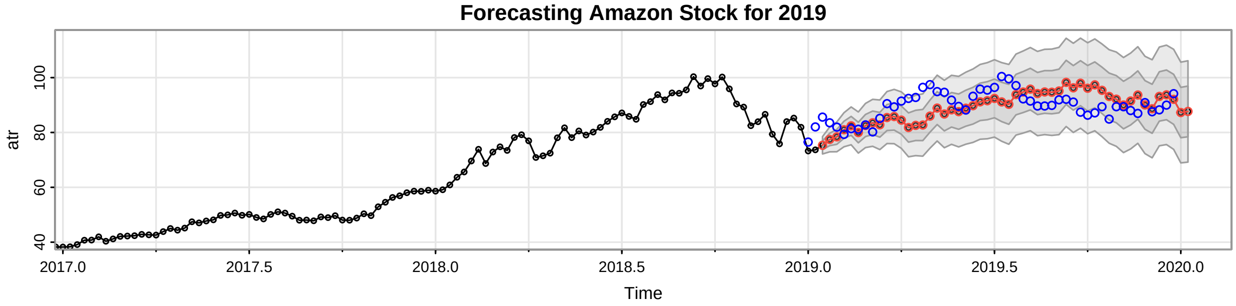
Microsoft Forecast

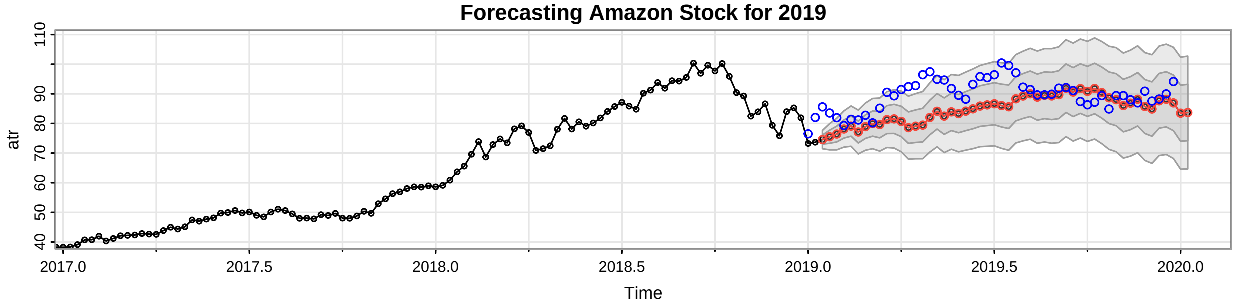


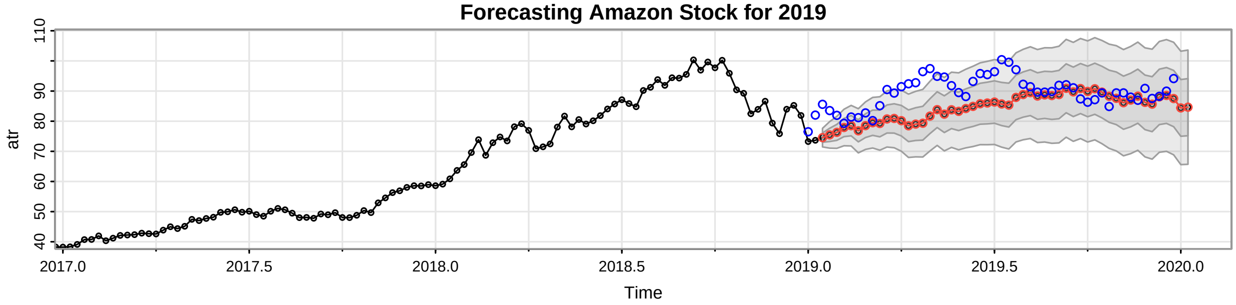




Amazon Forecast







VIII. Results

Google: The SARIMA model applied to Google's stock data projected a consistent, gradual upward trend, without significant seasonal fluctuations. This model closely tracked the actual stock prices until the third quarter of 2019, at which point the actual stock prices experienced a sharper increase than anticipated by the model. This deviation suggests that while the model captured the general trend effectively, it was less responsive to more abrupt market shifts.

Nvidia: The forecast for Nvidia's stock was notably accurate in its early stages, closely mirroring actual stock values. However, a deviation from the predicted trajectory was observed in the third quarter of 2019. Despite this divergence, the model's performance was commendable, ranking as the second most accurate among the analyzed stocks.

Microsoft: The predictions for Microsoft's stock presented a significant deviation from actual stock performance. The model appeared to rely on the stock's previous steady growth trend, failing to anticipate the substantial price increase that commenced at the outset of 2019. This discrepancy highlights the model's limitations in adapting to sudden market changes.

Amazon: The forecast for Amazon's stock was the most accurate, with the model successfully capturing its stable growth trend throughout most of 2019. The true stock values largely fell within the predicted confidence intervals, underscoring the model's effectiveness. This accuracy may be partly attributed to the inherent stability of Amazon's stock compared to more volatile market entities.

It is important to note external factors that potentially influenced these forecasts. One significant event was the escalation of the US-China trade war in mid-July 2019, coinciding with the latter part of our training dataset. This geopolitical development had a pronounced impact on the technology sector, which is heavily reliant on manufacturing and labor resources from China. Such external factors, although not directly accounted for in our model, likely played a role in the deviations observed in our forecasts.

In summary, while our models demonstrated a capacity to track general stock trends effectively, their performance varied across different companies and they showed limitations in responding to rapid market changes and external geopolitical events.

IX. Discussion

This study embarked on an in-depth examination of weekly stock prices from four prominent technology corporations over a decade (2010-2020). The primary aim was to evaluate predictive capabilities for the year 2019, utilizing ARIMA models as the central analytical tool.

Our comprehensive analysis model forecasts against actual stock performances, revealing varying degrees of prediction accuracy. Notably, Google's stock predictions aligned closely with real-world data until a significant rise in the third quarter of 2019, suggesting an underestimation of market dynamics. Nvidia's forecast mirrored actual trends up to a similar time frame, after which the prediction diverged. Microsoft's stock, however, presented the most substantial challenge, with the model failing to capture the increase observed in 2019. In contrast, Amazon's stock forecast emerged as the most accurate, with the predictive model effectively tracking its price trajectory throughout the year.

These outcomes suggest that while ARIMA models offer valuable insights, their precision varied considerably among the different stocks. Amazon's predictability positions it as a potentially reliable option for investors focusing on the technology sector. Nonetheless, the discrepancy observed, especially with Google and Microsoft, underscores the complexities inherent in stock market forecasting, especially in stocks that are much more volatile and susceptible to external factors.

This analysis not only underscores the utility of statistical models in financial forecasting but also highlights their limitations within highly dynamic sectors like technology. The deviations observed reiterate the necessity of considering a broader range of factors, including economic, political, and global health events, in future forecasting endeavors.

Conclusively, while the precision of our ARIMA models did not entirely meet expectations, they provided significant insights into stock market behaviors, underscoring the importance of nuanced analysis in making informed investment decisions. More sophisticated research could expand the model's parameters to encompass a wider array of influencing factors, such as the external factors which were not included in this study, potentially enhancing the accuracy and reliability of stock market forecasts.

References:

[1] Yahoo Finance. (2023). Amazon Stock History. Retrieved February 20, 2023, from https://finance.yahoo.com/quote/AMZN/history?p=AMZN

[2] Yahoo Finance. (2023). Microsoft Stock History. Retrieved February 20, 2023, from https://finance.yahoo.com/quote/MSFT/history?p=MSFT

[3] Yahoo Finance. (2023). Nvidia Stock History. Retrieved February 20, 2023, from https://finance.yahoo.com/quote/NVDA/history?p=NVDA

[4] Yahoo Finance. (2023). Google Stock History. Retrieved February 20, 2023, from https://finance.yahoo.com/quote/GOOGL/history?p=GOOGL

[5] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples* (4th ed.). Springer.