# Indian Institute of Science Education and Research Bhopal

# Project Report

# Time Series Analysis of Nvidia Stock


Project Guide – Neha B Upadhayay

Submitted by:

Rohit Rawat

Aditya Sinha

# 1. Introduction

In recent years, the landscape of technology and finance has undergone transformative changes, with companies like NVIDIA at the forefront of this evolution. As a leader in graphics processing units (GPUs), NVIDIA has capitalized on the surging demand for advanced computing power driven by artificial intelligence (AI) applications and the cryptocurrency mining boom. These developments have not only reshaped market dynamics but also generated significant interest among investors, making it imperative to understand their impact on stock performance.

Despite the growing body of literature examining the relationship between technological advancements and stock market behaviour, there remains a notable gap in focused analyses of NVIDIA's stock in the context of specific events such as AI hype and cryptocurrency trends. Previous studies have largely addressed these themes in isolation, failing to provide a comprehensive view that integrates the effects of both phenomena on NVIDIA's financial performance. This research aims to bridge this gap by investigating how these intertwined events have influenced NVIDIA's stock over a five-year period.

To address this research problem, we pose several key questions: How have the surges in AI interest and cryptocurrency mining activity correlated with NVIDIA's stock price fluctuations? What statistical relationships can be identified between these events and market behaviour? By answering these questions, this study seeks to illuminate the complexities of NVIDIA's stock dynamics in relation to broader technological trends. The significance of this research lies in its potential contributions to both academic and practical realms. By providing a nuanced understanding of how external technological trends impact stock performance, we can offer valuable insights for investors, analysts, and policymakers. Moreover, our findings may inform future investment strategies in an increasingly tech-driven market.

Our methodology employs a range of time series analysis techniques, including moving averages, seasonal decomposition. For forecasting, we utilize advanced models such as ARIMA, Facebook Prophet, XGBoost, Random Forest, GRU, and LSTM, all optimized through walk-forward optimization. This comprehensive approach enables us to capture intricate patterns in stock data while providing robust predictions.

The scope of this study is defined by its focus on NVIDIA's stock performance over the last five years, specifically analysing the influence of AI and cryptocurrency mining events. Limitations include potential external factors not accounted for in the analysis, such as macroeconomic conditions and competitor activities. Our main hypothesis posits that significant correlations exist between NVIDIA's stock price and the identified technological events.

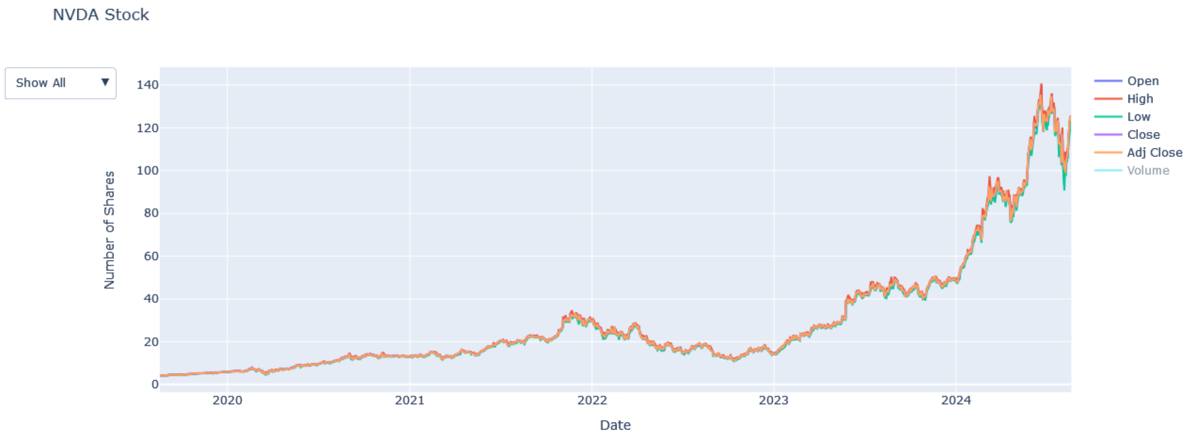
In the following sections, we will outline our methodology, findings, and discussions, culminating in conclusions that highlight the implications of our research and suggest avenues for future study. Through this structured approach, we aim to provide a comprehensive exploration of NVIDIA's stock performance within the context of rapidly evolving technological landscapes.

# 2. Data and Methodology:

The present study uses the NVIDIA STOCK MARKET dataset for the time span of August 2019, to August 2024 from the website (https://finance.yahoo.com/). The collected data are seasonally adjusted to correct for all types of missing values like those due to weekends or holidays.

## 2.1. Data Description

| | Open | High | Low | Close | Adj Close | Volume |
|---|---|---|---|---|---|---|
| **count** | 1828 | 1828 | 1828 | 1828 | 1828 | 1828 |
| **mean** | 29.139342 | 29.701165 | 28.560986 | 29.163804 | 29.141780 | 4.526540e+08 |
| **std** | 28.448584 | 29.006821 | 27.827771 | 28.444060 | 28.450982 | 1.819792e+08 |
| **min** | 4.014500 | 4.083500 | 3.975000 | 4.030500 | 4.011333 | 8.919675e+07 |
| **25%** | 12.883437 | 13.112500 | 12.707250 | 12.928625 | 12.898245 | 3.206995e+08 |
| **50%** | 18.420624 | 18.949376 | 18.084249 | 18.581917 | 18.554703 | 4.273730e+08 |
| **75%** | 32.773833 | 33.367750 | 31.900417 | 32.674500 | 32.621290 | 5.511105e+08 |
| **max** | 139.800003 | 140.759995 | 132.419998 | 135.580002 | 135.58002 | 1.543911e+09 |



## 2.2. Inference

Based on the dataset and the description of its features, we obtained results that serve as the foundation for further analysis.

- High Volatility: The large difference between the minimum and maximum values for Open, High, Low, and Close prices indicate significant volatility in the stock's price.
- Standard Deviation: The standard deviation values (28.45 for Open, 29.01 for High, etc.) are quite large relative to their means, further suggesting that there is substantial variability in the stock prices.
- The average (mean) prices for Open, High, Low, and Close are close to each other, indicating a relatively balanced price movement over the period.
- The median prices are lower than the mean prices particularly hinting towards the presence of outlier in the price dataset that is causing the skewed distribution towards higher values.

## 2.3. Plots

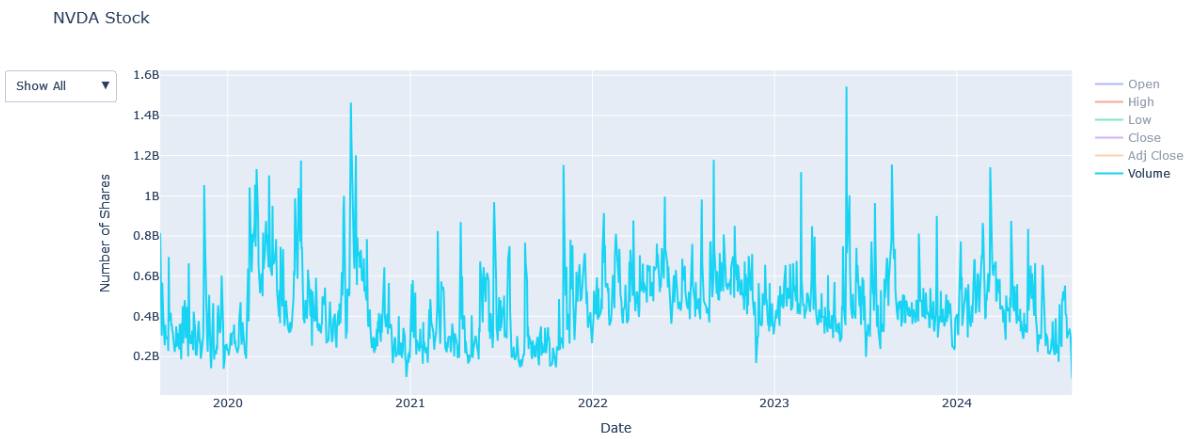



Fig. 1 Volume

- Fig-1 shows the plot of Volume over time. This plot provides a comprehensive view of how the volume dynamically shifts across different time intervals, revealing underlying trends, periodicities, and potential anomalies in the data.
- Volume Variability: The trading volume also shows a high range from about 89 million to over 1.5 billion. The standard deviation (1.82e+08) compared to the mean volume (4.53e+08) suggests that trading volume varies significantly

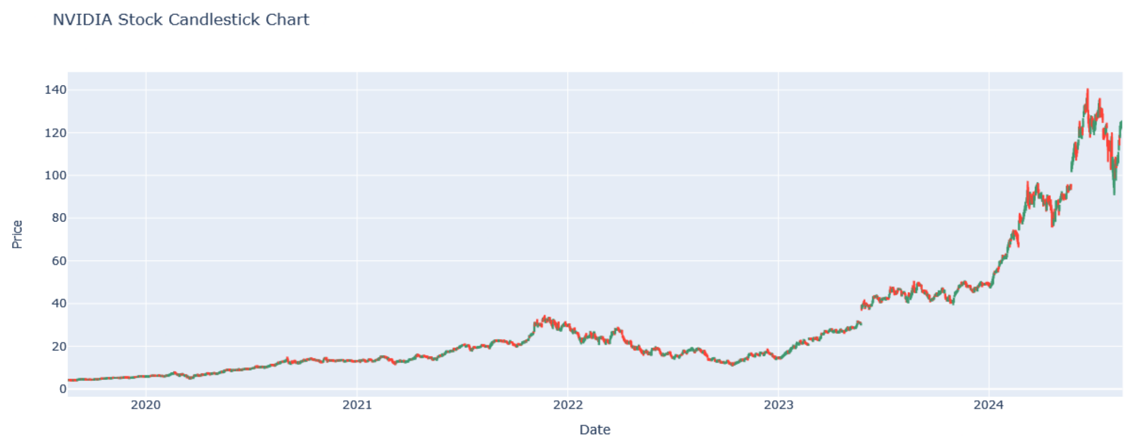



Fig. 2 Candlestick Chart

- Fig-2 represents the candle stick chart of the stock market prices of NVIDIA which helps in analyzing the potential trends in the price and also helps in analyzing the dynamic price movements.
- Candlestick charts helps in understanding market sentiment and identify patterns that may predict future price movements. The gap in the candlestick chart particularly during the mid 2023 and mid 2024 represents the strong price movements overnight and the direction of price movement is the highlighter of the market sentiment which will be covered in the later part of the paper.

# 3. Methodology

In this study, a comprehensive approach has been used to analyze the factors affecting the stock prices of NVIDIA. The approach has been further divided into three categories that use different means for impact identification but reaches out to the same conclusion in the end. The structure of the methodology starts with the statistical analysis followed by a more in-depth trend analysis and concluded by developing forecasting models that predicts the future stock prices movement.

## 3.1. Descriptive Statistical Analysis

As an extension of the data pre-processing phase, a descriptive statistical analysis was conducted to identify the underlying characteristics of the data.

### 3.1.1. Volatility Curve

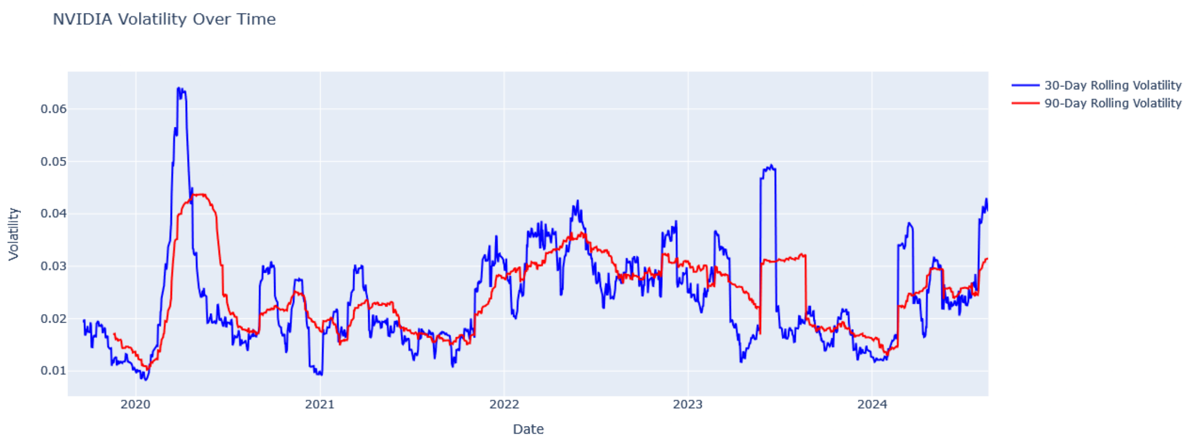



Fig. 3 Volatility Curve

The volatility curve is the graphical representation of the stock's volatility over time. The graph shows both 30 days and 90 days volatility. The 30 days volatility reflects onto short term fluctuations and trends whereas 90 days volatility is plotted to visualize the long-term trends. Comparing these two gives an idea of both macro level trends and the short-term fluctuations.

- First half of 2020
  - i) During the first half of 2020, the short-term fluctuations graph shows a spike of 6.4 percent. This sudden increase in the short-term volatility suggests that there was substantial market turbulence which can be associated to the COVID-19 global crisis that started in 2020 leading to major supply chain disruptions of semiconductors across the world.
  - ii) The longer-term volatility also showed a spike but to a lesser extent as it covers a longer time horizon, and it shows how the market began to digest and adapt the unfolding situation.
  - iii) The near about convergence of both short term and long-term volatility represents that the market has finally adapted the new normal and has adjusted to this new norm.
- Between 2021 and 2023
  - i) The convergence of both the volatility curve is an indicative of the market stability and zero abrupt news regarding the company.
  - ii) There are several instances when the short-term volatility has again crossed the short-term volatility. With events like unsuccessful acquisition of ARM and Increase in the Federal Reserve interest rates have contributed significantly to shape the curve as it looks.
- First half of 2024
  - i) The first half of 2024 saw a substantial peaks in both short term and long-term volatility and this reflects the fact that the investors are responding strongly to the recent developments. With AI becoming more powerful each passing day the demand for NVIDIA's GPU is also increasing.

### 3.1.2. Daily returns Curve

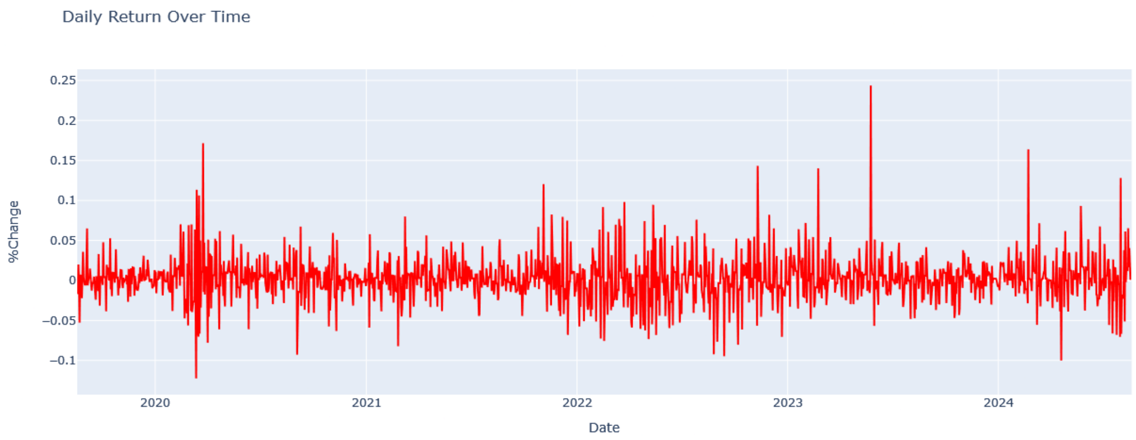



Fig. 4 Daily returns Curve

- Early 2020 saw the impact of COVID-19 and it is clearly visible with the strong density of both positive and negative daily returns.
- Post 2020 - Early 2022 the curve gets sparser showing the period of less frequent larger change of prices and highlight towards stability.
- Around 2022 to First Half of 2023 shows an increase in the magnitude of positive and negative daily returns increases, indicates a period of heightened volatility and larger daily price changes.
- Post-Mid 2023 shows a decrease in the density of positive and negative daily returns.

### 3.1.3. Seasonal Decomposition Curve

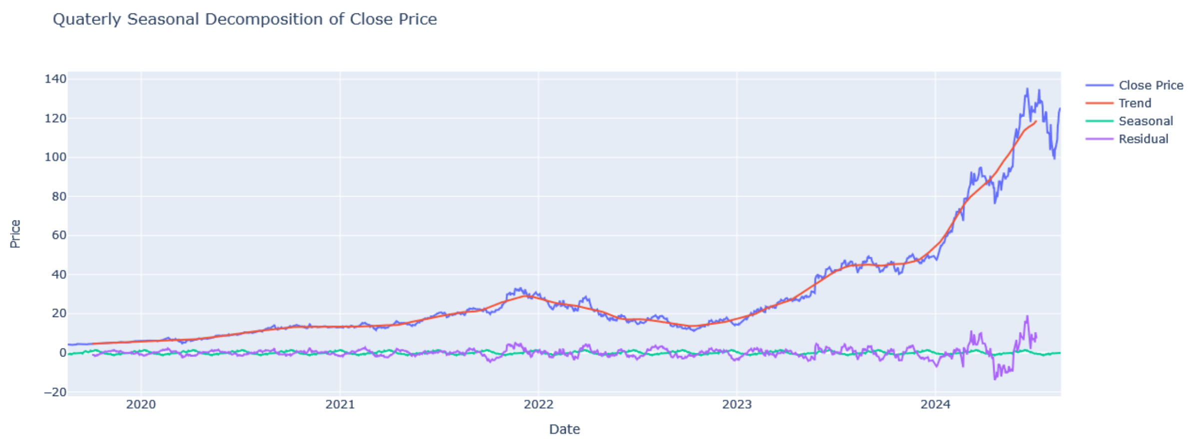



Fig. 5 Seasonal Decomposition Curve

The seasonal decomposition curve is an essential approach when dealing with the time series data which breaks the time series data into 3 components: Trend, which highlights the underlying direction and patterns in the time series data. Seasonality captures the patterns that are regular in that span of time. Residual, this highlights the unexplained fluctuations in the time series data.

- Trend Component Analysis: The trend line suffers a sharp shift after late 2023 shows a possible shift in the stock's long-term behaviour.
- Seasonality Analysis: Consistent fluctuations around zero indicates the presence of seasonal effects that are not very substantial.
- Residual Graph Analysis: Till late 2021, both residuals and seasonality are almost overlapping shows regular trends. Post that the magnitude of Residual components increases shows the addition of new influencing factor in the market

### 3.1.3. Moving Averages Curve


Fig 6. Moving Averages Curve

The moving averages plot in time series analysis is plotted to smooth out short term fluctuations and highlight the longer term trends.Let us analyze the different moving averages.

- Quarterly vs Half-Yearly Moving Average
  i) The quarterly moving average and the yearly moving average generally overlap highlighting the consistency of medium term with longer term trend on a half-yearly basis.
  ii) In 2023 the quarterly MA begin to exceed the half yearly MA reflects the stronger short term performance of the stock than the long term performance.
  iii) Later both these curves again super imposed onto each other shows the agreement of both these curves again.
- Yearly Moving Average
  i) The yearly MA stayed below the half yearly MA and the quarterly MA mostly indicates strong short term trends than the longer term trends.
- Monthly Moving Average vs Quarterly Moving Average

ii) The monthly MA was generally below or at the same level with the quarterly MA till late 2021. After this the monthly MA fluctuated around the quarterly MA this shows the period of increased volatility in the monthly data compared to the more stable quarterly trend.

iii) During late 2022, the monthly MA started to fluctuate around and sometimes above the quarterly MA. This suggests a shift where the short-term trends became stronger or more variable compared to the quarterly trend.

iv) By late 2023, the monthly MA became equal to the quarterly MA, and into 2024, it started dominating the quarterly MA. This indicates a significant shift where short-term performance became more prominent than the quarterly trend, possibly reflecting a period of rapid or sustained short-term gains.

The descriptive statistical analysis has given us some key observations, notably the arrival of a new influencing factor in the market in the late 2021. This shift appears to correlate with positive trajectory in the market growth.

In the light of these observations this paper proposes that the substantial growth in AI field and significant increase in the adoption of AI during these periods has led to these market observations. The growth of AI certainly increases the necessity of strong computational capabilities, particularly using GPU that are useful in training AI models and dealing the large datasets that are used to train the models. Being the leading provider of Graphics processing unit or GPUs NVIDIA stands to get benefit from the growing demand.

## 3.2. Forecasting

### 3.2.1. Stationarity of Data

Financial time series data often exhibit trend and volatility that can change overtime, leading to time series data being non-stationary. For accurate analysis, it becomes a priority to check whether the time series data is stationary or not. Stationary data gives more accurate results whereas non-stationary data leads to misleading inferences, which further can bias the model. A stationary data has properties like mean, variance, standard deviation that do not change over time. A unit root means that the data is non-stationary, and it can drift and don’t revert to long term mean. To check the stationarity of the data we perform the following tests:

- **Augmented Dickey Fuller Test:**

$$\Delta Y_t = \alpha + \beta t + \gamma Y_{t-1} + \sum_{i=1}^{p} \delta_i \Delta Y_{t-i} + \epsilon_t \tag{1}$$

Equation-1 represents the general equation of Dickey Fuller Test where $Y_t$ represents the value of the time series at time t, $\alpha$ is a constant, $\Delta Y_t$ is the first difference of $Y_t$, $\beta$ is the coefficient on the time trend, $\gamma$ is the coefficient to be tested, p represents the number of lagged difference terms and $\epsilon t$ represents the white noise error term.

**Null Hypothesis:** $\gamma = 0$, the dataset has a unit root and is non-stationary.
**Alternate Hypothesis:** $\gamma < 0$, the dataset does not have a unit root and is non-stationary.

**ADF Statistics**: 1.991043

**p value**: 0.998661z

The ADF statistics being high positive shows that it is tough to reject the Null Hypothesis and the p value is very close to 1 shows very high probability of the occurrence of the event in the Null Hypothesis. This test concludes that the time series data is non-stationary.

- Transforming the non-stationary data to stationary data.
  - First Differencing method:

$$Y_t' = Y_t - Y_{t-1} \tag{2}$$

Equation-2 is the general equation of the First Differencing method. Suppose a dataset with n data points y1, y2, y3, .... yn, first differencing method generates n − 1 data points y2 − y1, y3 − y2, ...., yn − (yn − 1). These data points that have been generated are now converted to stationary data points.

- Applying the Augmented Dickey Fuller Test again using Equation-1.

ADF Statistics -8.36726

p value 2.72171e-13

Null Hypothesis is rejected: Data is Stationary.

The data point has been transformed from non-stationary to stationary. Now will perform some more tests that will highlight the correlation of the data points with its past values.

- **Auto Correlation Test:** Measures the correlation between the time series and its lagged versions using the auto correlation function

$$\rho_k = \frac{\sum_{t=k+1}^{n}(Y_t - \bar{Y})(Y_{t-k} - \bar{Y})}{\sum_{t=1}^{n}(Y_t - \bar{Y})^2} \tag{3}$$

Equation-3 is the Auto correlation function where $\rho_k$ is the autocorrelation at lag k, $Y_t$ is the value of the time series at time t, Y is the mean of the time series, n is the total number of observations in the time series, k is the lag, representing the number of time steps between the values being compared.

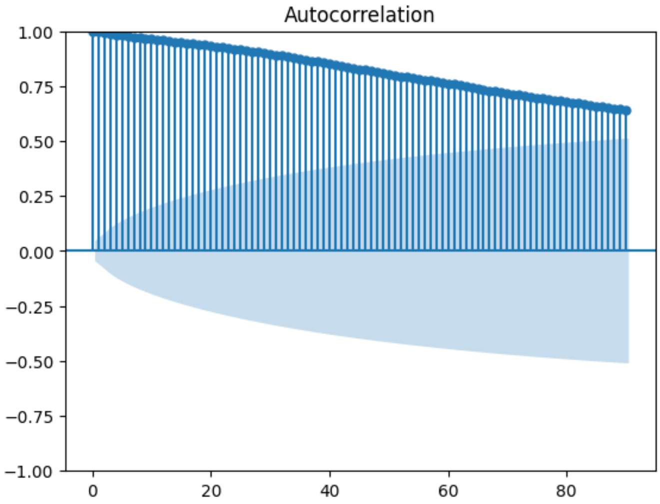



Fig 7. Auto-correlation plot before differencing

This graph shows how the current values of closing price are correlated with the past values. A slow decay series is the prime indicator of a non stationary time series data and the Augmented Dickey Fuller test has also suggested that.

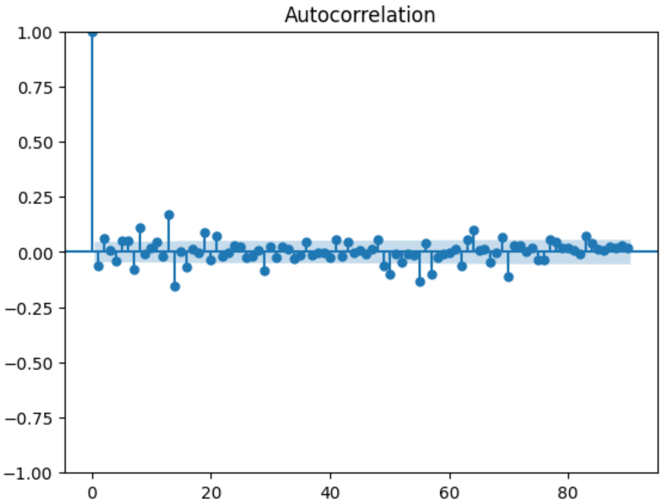



Fig 7. Auto-correlation plot after differencing

First differencing converts the data-points from non stationary to stationary as seen in the Auto-correlation plot also as the auto correlation values are now close to zero. The near close to zero values suggest that the current time series has no such relation with the previous data point and no such underlying hidden patterns are there so tha different forecasting models can be applied to them.

With all these data pre-processing and the transformation, dataset is ready to be used for forecasting purposes. But before we start discussing about the forecasting models, lets understand the basic structure of how the models work.

### 3.2.2. Optimisation

**Walk Forward method** is used in the forecasting of the time series data to consider for the real-life predictions where new data is added to the model continuously after each iteration.

**Algorithm:**

- Split the data points in the training and test dataset.
- Training Dataset is used to train our forecasting models and test is used to evaluate their performance. Train the forecasting model using the training dataset.
- Suppose there are n points in the test dataset.
  i) for i in n:
  ii) Predict for $Y_i$ and store the prediction for future evaluation.
  iii) Update the Training set with new data point $Y_i$.
  iv) Retrain the model using this updated training set.
- Random forward method is used with other models statistical and machine learning models to do the forecasting.

### 3.2.3. Forecasting Models

Conventionally this paper proposes seven diverse set of statistical and Machine learning models for forecasting purposes. Each of these models possess different characteristics analysing the dynamics and characteristics of the dataset.

**Model-1: Auto-regressive Linear Regression**

This statistical regression technique used for forecasting is based on the consideration that the current closing price can be analysed through the previous values. The main assumption of this model is the linear dependency between the current and the previous value of the time series dataset. The general equation of the Auto-Regressive Linear Model of order p is:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \epsilon_t$$

where $Y_t$ is the value of the time series at time t, c is a constant (intercept), $\phi_1, \phi_2, \dots, \phi_3$ are the parameters (coefficients) of the model, $\epsilon t$ is the error term, assumed to be white noise. The identification of the order of the model is important as it measures how many previous values we are considering while considering the value of the current time series. For the identification of the order of the model, we use the partial auto- correlation test. The plot of partial auto correlation test gives us the inference about the lags with which we can identify the lags in the dataset.

From the partial Auto-correlation test, we selected the value of lag to be 7. A seven lagged Auto-regressive Linear Regression along the walk forward method is applied. The above written equation shifts to:

$$Yt = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \phi_3 Y_{t-3} + \phi_4 Y_{t-4} + \phi_5 Y_{t-5} + \phi_6 Y_{t-6} + \phi_7 Y_{t-7} + \epsilon_t$$

where Yt is the closing price at time t. $Y_{t-i}$ is the closing price lagged by i units of time, $\phi_n$ is the regression coefficient of the nth lag, $\epsilon t$ is the white noise.

## Model-2: Auto-Regressive Integrated Moving Average

This statistical method used for forecasting is a combination of three components: Auto-regressive, differencing, Moving Average.

- Auto-Regressive Part: As discussed earlier this part helps in capturing the dependence of the current time series and the lagged value.
- Integrated Part: This term captures the differencing of the time series data to make it stationary. The number of times the data must be differenced to achieve stationarity is the order of integrated part represented by *d*. The time series data here is differenced one time to achieve the stationarity of the data. The value of d here is 1.
- The moving average component models the relationship between the current observation and the number of past error terms. The value of q here satisfies number of lagged forecast error is considered in the model.

The general ARIMA model is denoted by ARIMA (*p, d, q*) where *p* is number of Auto-Regressive terms, *d* is the number of times the differencing has to be performed to achieve stationarity and *q* is the number of lagged error terms or say Moving Average terms considered. The mathematical formulation of ARIMA (*p, d, q*) is given as:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + . . . + \phi_p Y_{t-p} + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + . . . + \theta_q \epsilon_{t-q} + \epsilon_t$$

$Y_t$: The value of the time series at time t

c: A constant term

$\phi_i$: The coefficients for the lagged values of the time series, where i = 1, 2, . . . , *p*

$Y_{t-i}$: The value of the time series at time t – i

$\theta_j$ : The coefficients for the lagged error terms, where j = 1, 2, . . . , *q*

$\epsilon_t$: The error term at time t.

Choosing the parameters (*p, d, q*): The choice of right parameters is important as it defines the performance of the selected model. To select the right parameters, we use the plots of Auto-correlation function and the partial Auto-Correlation function. The section 3.3.1 suggests that the stationary behaviour of the time series dataset was attainted by a single differencing, this gives the order of

Integrated part, or the value of $d$ is 1. For the values $p$ and $q$, the model was run on different combinations of $p$ and $q$.

This gives us the value of parameters as $p$=5, $d$=1, $q$=0.The equation of our model ARIMA (5,1,0)

is:

$$Y_t - Y_{t-1} = \phi_1(Y_{t-1} - Y_{t-2}) + \phi_2(Y_{t-2} - Y_{t-3}) + \phi_3(Y_{t-3} - Y_{t-4}) + \phi_4(Y_{t-4} - Y_{t-5}) + \phi_5(Y_{t-5} - Y_{t-6}) + \epsilon_t$$

## Model-3: Facebook Prophet:

This forecasting tool is developed by Meta Researchers for time series data having strong seasonality component. This forecasting tool is well suited for time series analysis having clear seasonal patterns and clear trend changes. The general equation of Prophet is:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

Where y(t) is the observed value at time t, g(t) represents the trend component, s(t) captures the seasonal effects, h(t) accounts for holiday influences, and $\epsilon_t$ is the error term.

**Model-4: Random Forest:**

While the previous forecasting models were mostly focussed on handling linear patterns and seasonality trends, Random Forest is an ensemble learning technique. The method develops B (say) number of decision trees and combine their results. In the single tree out of B trees the method considers a bootstrap sample of the training set instead of the entire training set, i.e. for each iteration the method randomly selects Z samples out of the set of training set X and build the tree.

Eventually, the results of the B trees are combined to take a decision i.e. a particular test samples will be classified following most of the decision of the B decision trees. During the training phase, for each decision tree, each time a node needs to be split, the search for the best feature to split on is limited to a subset of all features and the subset is randomly drawn from the set of all features. This random drawn is made separately for all nodes in each tree in the set of B trees, and is of fixed size, say m, a prefixed threshold.

Algorithm:

- for b=1 to B do:
- Randomly select Z samples (with replacement) from the set of training samples X.
- Build a decision tree using Z samples recursively repeating the following steps.
- Randomly select V features from the set of total features F.
- Find the best feature out of these V features following a splitting measure.
- Split the node on the best feature for all possible values of this feature.

- end for.
- Predict the value of a particular test sample by taking the average of the predictions from the B decision trees.

### Model-5 XGBoost:

It is an advanced implementation of gradient boosting, a machine learning approach to build model sequentially to improve accuracy. It constructs a series of decision trees where each tree is trained to correct the errors made by the previous trees. The predictions are made by aggregating the predictions from all trees.

**Model-6 LSTM:**

Before understanding LSTM, let's get an idea about the traditional Recurrent Neural Networks. RNNs are designed to process sequences of data by maintaining hidden state that carries information from previous time step. As the name implies, recurrent neural networks have a recurrent connection in which the output is transmitted back to the RNN neuron rather than only passing it to the next node. Each node in the RNN model functions as a memory cell, continuing calculation and operation implementation. If the network's forecast is inaccurate, the system self-learns and performs back propagation toward the correct prediction. This makes them suitable for tasks involving sequential data, like time series forecasting and natural language processing. However standard RNNs struggle with learning long term dependencies due to the vanishing and exploding gradient problems. This takes us to Long Short-Term Memory (LSTM) which is an improved version of recurrent neural network. A traditional RNN has a single hidden state that is passed through time, which can make it difficult for the network to learn long-term dependencies. LSTM models address this problem by introducing a memory cell, which is a container that can hold information for an extended period.

- LSTM Architecture- LSTMs improve on traditional RNNs by incorporating mechanisms to better capture long term dependencies. This is achieved through a more complex architecture involving gates that control the flow of information.
- Components: An LSTM consists of several components:
    i) Cell State ($C_t$): It acts as a memory that carries information across time steps. It is updates through the network and helps in preserving long term dependencies.
    ii) Hidden State ($h_t$): The hidden state represents the output of the LSTM at each time step and is used as input for the next time step.
    iii) Input Gate ($i_t$): Determines how much of the new information should be added to the cell state. It uses the sigmoid activation function to control the update.

**Model-7 Gated Recurrent Units (GRUs):**

They are also an improvement on RNNs like LSTM. They improve upon the standard RNNs by using gating mechanisms to manage the flow of information. GRU uses a combination of fewer gates, making them less computationally intensive while still effective in capturing temporal dependencies.

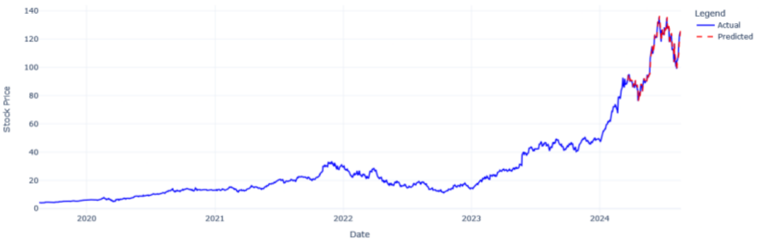
- GRU Architecture: GRU simplifies the LSTM architecture by combining the forget and input gates into a single update gate, reducing the number of parameters while still achieving similar performance in many tasks.
- Components: A GRU consists of following key components:
  i) Update Gate (z): The update gate controls how much of the previous hidden state should be carried forward to the next time step and how much of the new information should be incorporated. It uses the sigmoid activation function to decide this mixture.
  ii) Reset Gate (r): The reset gate determines how much of the previous hidden state should be forgotten when generating the new candidate hidden state. It also uses a sigmoid activation function to control this behaviour.
  iii) Candidate Hidden State ($\hat{h}$): This is the new potential hidden state, which is calculated using the reset gate to modify the previous hidden state. The candidate hidden state is then blended with the previous hidden state by the update gate to form the final hidden state at each time step.

**Code:** https://github.com/rrawatt/nvda_analysis

# 4. Results

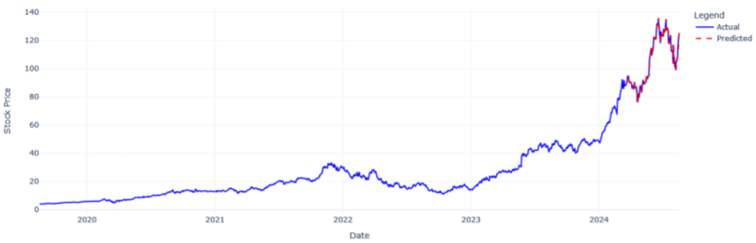
## 4.1. Auto-Regressive Linear Regression

| Metric | Value |
|---|---|
| Mean Absolute Percentage Error (MAPE) | 2.0056 |
| Mean Absolute Error (MAE) | 2.1350 |
| $R^2$ Score | 0.9663 |
| Root Mean Square Error (RMSE | 3.0853 |



## 4.2. Auto-Regressive Integrated Moving Average

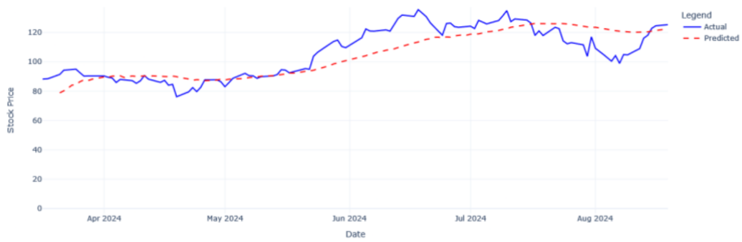
| Metric | Value |
| --- | --- |
| Mean Absolute Percentage Error (MAPE) | 1.9662 |
| Mean Absolute Error (MAE) | 2.0920 |
| $R^2$ Score | 0.9668 |
| Root Mean Square Error (RMSE | 3.0634 |

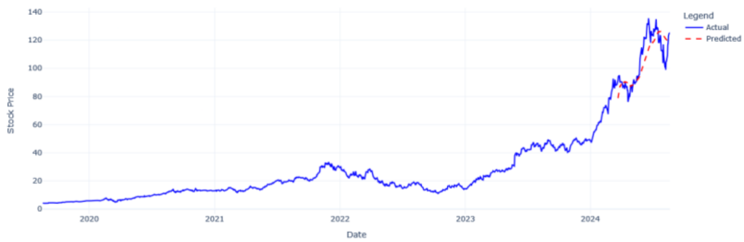




## 4.3. Facebook Prophet

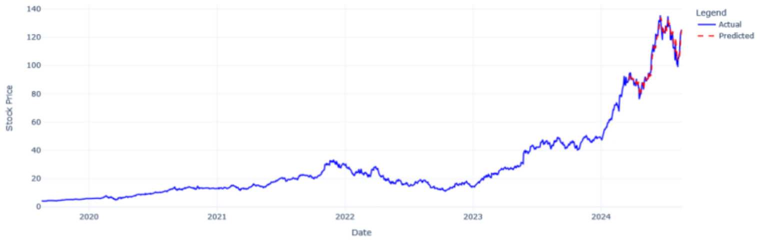
| Metric | Value |
| --- | --- |
| Mean Absolute Percentage Error (MAPE) | 7.0632 |
| Mean Absolute Error (MAE) | 7.7393 |
| $R^2$ Score | 0.6567 |
| Root Mean Square Error (RMSE | 9.8534 |



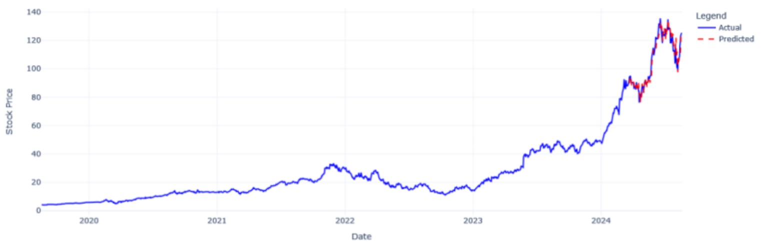
## 4.4. Random Forest

| Metric | Value |
| --- | --- |
| Mean Absolute Percentage Error (MAPE) | 2.5511 |
| Mean Absolute Error (MAE) | 2.7187 |
| $R^2$ Score | 0.9503 |
| Root Mean Square Error (RMSE | 3.7488 |




## 4.5. XGBoost

| Metric | Value |
| --- | --- |
| Mean Absolute Percentage Error (MAPE) | 2.8168 |
| Mean Absolute Error (MAE) | 3.0070 |
| $R^2$ Score | 0.9385 |
| Root Mean Square Error (RMSE | 4.1716 |



## 4.6. LSTM

| Metric | Value |
|---|---|
| Mean Absolute Percentage Error (MAPE) | 0.0361 |
| Mean Absolute Error (MAE) | 3.7025 |
| $R^2$ Score | 0.9385 |
| Root Mean Square Error (RMSE | 0.9263 |



## 4.7. GRU

| Metric | Value |
|---|---|
| Mean Absolute Percentage Error (MAPE) | 0.0298 |
| Mean Absolute Error (MAE) | 3.0181 |
| $R^2$ Score | 0.9531 |
| Root Mean Square Error (RMSE | 3.8307 |


# 5. Conclusion

The time series analysis of NVIDIA's stock from 2020 to 2024 highlights the company's journey through a highly volatile and dynamic market environment. This period has been shaped by both macroeconomic factors (such as the COVID-19 pandemic and Federal Reserve interest rate hikes) and company-specific developments (including NVIDIA's leadership in the booming AI sector and its role in the broader technology market). These external and internal influences have led to substantial fluctuations in NVIDIA's stock prices, as investors have reacted to changing economic conditions and the company's shifting business landscape.

## 5.1. Macroeconomic Influences and Volatility

One of the most significant drivers of volatility during this period was the COVID-19 pandemic in early 2020. The pandemic caused massive uncertainty across global markets, with sharp price swings observed in both directions as the stock market reacted to rapidly changing economic conditions. Early 2020 saw a spike in short-term volatility, reaching as high as 6.4%. This was primarily due to the market's reaction to the sudden economic disruptions caused by the pandemic, including widespread lockdowns, supply chain disruptions, and a global economic slowdown. NVIDIA, being heavily involved in the tech sector, was not immune to these shocks, but it managed to recover quickly as the tech industry benefited from shifts toward remote work and increased demand for digital infrastructure.

Additionally, Federal Reserve interest rate hikes and broader inflationary concerns during the post-pandemic recovery period also created volatility. As the Fed raised rates to combat inflation, markets experienced further fluctuations, particularly in 2022 and 2023, when uncertainty about the broader economic outlook led to increased market turbulence. NVIDIA's stock was not immune to these broader economic pressures, leading to spikes in short-term volatility. For instance, the market saw increased fluctuations in the short term, driven by investor responses to interest rate decisions and shifting expectations about future economic conditions.

## 5.2. The Crypto Boom and Its Impact on NVIDIA

During 2020 and 2021, cryptocurrency prices, especially for Bitcoin and Ethereum, saw explosive growth, leading to a boom in cryptocurrency mining. Cryptocurrencies rely on GPU power for mining operations, particularly Ethereum, which is known for being mined using high-performance GPUs. NVIDIA's graphics cards, especially the GeForce RTX series, became highly sought after by cryptocurrency miners due to their powerful processing capabilities.

The cryptocurrency bubble's rise and fall significantly impacted NVIDIA's stock, contributing to heightened volatility in 2021. As the price of Ethereum and other cryptocurrencies skyrocketed, demand for NVIDIA's GPUs surged, which likely led to a short-term boost in stock prices. However, the eventual collapse of the crypto market bubble, marked by sharp declines in the price of cryptocurrencies in late 2021, led to a sudden decrease in mining demand and affected NVIDIA's sales projections for its GPU products.

This contributed to sharp price swings in NVIDIA's stock—a mixture of optimism driven by strong demand for GPUs during the crypto boom, followed by concerns over a downturn in mining activity as the cryptocurrency market corrected. The short-term volatility observed during this period can be largely attributed to the cryptocurrency sector's boom and bust cycle.

However, the crypto bubble was characterized by extreme price fluctuations, and as cryptocurrency prices surged, they also faced sharp corrections, particularly in the second half of 2021. This bubble burst when the price of major cryptocurrencies started to decline, and crypto mining demand for NVIDIA's products tapered off.

## 5.3. The A.I. Boom

Alongside broader economic factors, company-specific developments have played a critical role in shaping NVIDIA's stock price movements. One of the most notable developments in recent years has been the growth of artificial intelligence (AI) and the increasing demand for NVIDIA's GPUs to power AI applications. NVIDIA has positioned itself as a leader in AI technologies, with its GPUs being widely used in data centres, autonomous vehicles, and other AI-driven applications. As AI technologies have matured, so too has the demand for NVIDIA's hardware, which has directly contributed to a surge in the company's stock price.

The AI boom has acted as a significant tailwind for NVIDIA's stock in 2023 and 2024, particularly as companies across industries ramp up their AI investments. This growth in demand for GPUs, coupled with the company's innovation in AI-specific hardware and software, has led to strong short-term stock performance. The heightened investor focus on the AI sector and NVIDIA's position as a key player in this space has led to increased short-term volatility in the stock price. The fluctuations in short-term moving averages reflect the market's growing attention to the company's role in this rapidly evolving industry.

## 5.4. Forecasting Performance

- **Best Overall Performance:** LSTM shows the best performance overall, with an exceptionally low MAPE (0.0361) and RMSE (0.9263), reflecting its ability to make very precise predictions. However, its $R^2$ and MAE are slightly weaker compared to the simpler models like AR and ARIMA.
- **Strong Linear Models:** AR and ARIMA provide solid performance with very low MAPE and high $R^2$ scores (around 0.96), making them excellent models for forecasting stock prices, especially when computational efficiency is important.

- **Traditional Machine Learning Models:** Random Forest and XGBoost show good performance but are slightly less accurate than the time-series models. The $R^2$ score for both is high, but they experience higher RMSE values, suggesting some sensitivity to volatility.
- **GRU (Gated Recurrent Unit):** The GRU model provides impressive results with a MAPE of 0.0298, reflecting high forecasting accuracy. It achieves an excellent $R^2$ score of 0.9531, indicating a strong fit to the data and explains 95.31% of the variance. However, its RMSE value of 3.8307 and MAE of 3.0181 are slightly higher than LSTM, showing that while GRU is highly effective, LSTM still outperforms it slightly in terms of precision.
- **Facebook Prophet:** Facebook Prophet performed the worst with high error metrics across the board. Despite its popularity in time-series forecasting, it was less effective for stock price prediction, likely due to the complexity and volatility of stock market data.