



DATA ANALYTICS CASE STUDY 3

(DAMO-611-2)

Enhancing Investment Decision-Making Through Forecasting Google Stock Prices Using Time Series and Machine Learning Techniques

Group Members

Name	ID
Jesun Ushno	NF1012471
Agba, Isaac Inalegwu	NF1006935
Esosa Simeon Omwanghe	NF1014811

Professor: Payman Janbakhsh, Ph.D

Master of Data Analytics

Summer 2025

Abstract

This study addresses the need for accurate stock price forecasting to inform data-driven investment decisions, comparing traditional ARIMA models with advanced machine learning techniques (XGBoost and LSTM) for Google (Alphabet Inc., GOOGL) stock prices. Historical daily GOOGL data from January 2015 to January 2025, manually collected from Kaggle, underwent preprocessing and feature engineering, incorporating technical indicators and lagged price values. Models (ARIMA(5,1,0), tuned XGBoost, and a multivariate LSTM) were evaluated using MAE, RMSE, R², and MAPE.

LSTM achieved the highest accuracy (R²=0.93, RMSE=6.87), effectively capturing nonlinear temporal patterns and long-term dependencies. XGBoost also performed strongly (R²=0.69, RMSE=15.25), providing interpretability with lagged closing prices and moving averages as key predictors. Conversely, the ARIMA model was unsuitable for the non-stationary data, yielding a poor R² of -3.68. These findings confirm that machine learning models outperform ARIMA and that engineered features enhance predictive power. The study suggests LSTM for high-accuracy, long-term strategies and XGBoost for feature-driven tactical trading, potentially increasing Return on Investment (ROI) by 5–10%. Limitations include the exclusion of macroeconomic and sentiment data, recommending future exploration of ensemble techniques and external variables.

Keywords: stock price forecasting, time series analysis, machine learning, ARIMA, XGBoost, LSTM, financial analytics

CHAPTER I

1.0 Statement of Purpose

The purpose of this project is to explore and evaluate different forecasting methodologies applied to the historical stock prices of Alphabet Inc. (Google). The study aims to contribute to the broader field of financial analytics by comparing traditional time-series models such as ARIMA with modern machine learning techniques including XGBoost and LSTM. The project seeks to determine which model provides the most accurate and robust predictions for stock price movement and how these models can be used to support investment decision-making processes.

Given the volatility of financial markets and the critical role accurate forecasts play in investment decisions, this project is relevant for a variety of stakeholders including retail investors, institutional analysts, and financial technology developers. The goal is to bridge academic methodologies with practical applications, generating actionable insights and improving the efficiency of financial forecasting systems.

1.1 Problem Definition

The prediction of stock prices plays a pivotal role in modern financial analytics and investment strategy. In the context of business analytics, it aligns directly with data-driven decision-making by empowering stakeholders to make timely and informed trading and portfolio allocation decisions. As global financial markets become increasingly complex, leveraging advanced analytics and machine learning offers an edge to market participants who aim to manage risk and maximize returns.

Google (Alphabet Inc.) represents one of the world's most valuable and actively traded companies. It is a core component of the S&P 500 and NASDAQ indices, and its stock

(GOOGL) is highly sensitive to both company-specific events and macroeconomic conditions. Forecasting the price behavior of such a stock involves analyzing vast volumes of historical data and identifying patterns that could inform future price action.

Traditional methods such as ARIMA and exponential smoothing have long been used for financial time series forecasting, but they often fall short in capturing nonlinear dependencies or reacting to regime changes. Machine learning models like XGBoost and deep learning architectures such as LSTM networks have emerged as powerful alternatives, capable of modeling complex relationships and learning from large datasets without requiring strong statistical assumptions (Zhang et al., 2021).

This project explores the intersection of these methodologies—comparing the performance of classical time series models and advanced machine learning approaches in forecasting Google stock prices. The overarching goal is to support improved investment decisions through robust, data-informed predictions. This study contributes to the academic discourse on hybrid forecasting systems and offers practical insights for traders, quantitative analysts, and institutional investors.

In a broader context, this work supports the development of intelligent financial systems that integrate automated analytics pipelines for real-time forecasting, enhancing the agility and effectiveness of decision-making in the volatile stock market environment (Chen et al., 2020). Forecasting stock prices is a high-impact application of business analytics in finance. Investors and portfolio managers depend on predictive models to anticipate market movements and make informed decisions. However, the volatile and nonlinear nature of financial markets often limits the effectiveness of traditional models.

Google (Alphabet Inc.) stock represents one of the most closely watched assets on the NASDAQ due to its market capitalization and influence. Investors aim to maximize returns by forecasting price movements with high precision. Yet traditional models, while statistically rigorous, may lack the flexibility to account for nonlinear dependencies and hidden patterns in high-frequency financial data.

This project investigates the use of both classical time series methods and advanced machine learning techniques to forecast Google stock prices. By doing so, we aim to enhance the accuracy and reliability of investment decision-making using data-driven approaches, thereby aligning with modern business analytics practices. The research also contributes to the growing body of financial analytics literature focusing on hybrid forecasting models (Babu & Reddy, 2016).

1.2 Scope of the Project

This project is scoped to perform a comprehensive end-to-end analysis and modeling of Google (Alphabet Inc.) stock prices using both traditional time series forecasting techniques and modern machine learning models. The scope is intentionally bounded to historical daily stock price data from January 2015 through January 2025, as obtained from publicly available sources such as Kaggle. The focus is on closing price prediction, but additional variables such as trading volume, opening/closing ranges, and technical indicators (e.g., moving averages, RSI) will also be explored as part of the feature engineering process.

The modeling approaches include ARIMA for classical statistical forecasting and XGBoost and LSTM for machine learning and deep learning modeling. These methods were selected for their wide applicability in financial forecasting literature, interpretability, and

availability of mature tools in Python for implementation. Hyperparameter tuning, cross-validation, and model comparison will be part of the analytical workflow.

In addition to model building, the scope also encompasses visual and statistical data analysis, including trend decomposition, stationarity testing, autocorrelation checks, and data transformation. Python will be the primary tool for modeling and preprocessing, while Tableau will be used in Phase 3 for building visual dashboards and highlighting key trends and insights from the data.

Deliverables include:

1. Cleaned and feature-engineered dataset
2. Time series and machine learning models with performance metrics
3. Model comparison report and evaluation summary
4. Use of publicly available historical stock data for Alphabet Inc. (GOOGL)
5. Application of two modeling approaches: classical time series models (ARIMA) and machine learning models (XGBoost, LSTM)
6. Evaluation of model performance using metrics such as RMSE, MAE, and MAPE
7. Visualization of historical data and model outputs using Python libraries and Tableau

CHAPTER 2

2.0 Background Research and Literature

Financial time series forecasting has long relied on traditional statistical models such as ARIMA, which are effective at capturing trends and seasonality in stationary data. However, these models are constrained by their linear assumptions and limited adaptability to the complexities of modern financial markets.

Recent research has shown that machine learning and deep learning models offer significant advantages by capturing nonlinear dependencies and learning from high-dimensional data. Long Short-Term Memory (LSTM) networks, in particular, have been successful in modeling sequential dependencies and outperforming traditional models in several stock market prediction tasks. Studies by Fischer and Krauss (2018) and Patel et al. (2015) provide empirical evidence of this advantage, noting that deep learning models and ensemble methods yield superior forecasting accuracy.

Furthermore, hybrid approaches that combine statistical and machine learning techniques have gained attention. Makridakis et al. (2018) emphasized the effectiveness of these models in business forecasting competitions, suggesting that integrating both methodologies can enhance robustness and adaptability.

These findings support the rationale of this project—to evaluate and compare traditional and machine learning forecasting models in predicting Google stock prices, aiming to identify the most suitable technique for enhancing investment decision-making.

2.1 Research Questions

In alignment with the problem statement, this project is designed to explore the viability, performance, and implications of various forecasting techniques applied to Google stock prices. The research questions aim to evaluate the practical value of both classical and modern predictive analytics tools in financial market contexts:

1. What patterns, trends, and seasonalities can be identified in historical Google stock price data using exploratory data analysis and visualization techniques?
2. How accurate and interpretable are traditional time series models (e.g., ARIMA, SARIMA) in forecasting Google's daily closing prices?
3. Can machine learning models—particularly LSTM (a type of recurrent neural network) and XGBoost—outperform statistical models in terms of prediction accuracy and robustness?
4. How does the inclusion of additional financial features (e.g., trading volume, lagged returns, and moving averages) impact the performance of predictive models?
5. What trade-offs exist between accuracy, interpretability, and computational efficiency among the models tested?
6. How can the outcomes of the models be applied in real-time or near-real-time investment decision-making frameworks?

These questions will guide the model development and evaluation process while contributing to a deeper understanding of the strengths and limitations of data-driven forecasting techniques in the domain of financial analytics. They are also framed to encourage

reproducibility, transparency, and applicability to business intelligence systems used in investment decision-making.

2.2 Relevance and Importance of Research

Accurate stock price forecasting has direct financial implications. In the age of algorithmic trading and data-driven financial decision-making, enhanced predictive accuracy can lead to increased returns and reduced investment risks. According to Patel et al. (2015), machine learning models often surpass traditional methods in financial forecasting tasks due to their ability to learn complex, nonlinear relationships.

Google stock (GOOGL), as part of the FAANG group, heavily influences the S&P 500 index. Its price movement serves as an indicator of broader tech-sector sentiment and investor confidence. Understanding and forecasting these movements can help investors manage risk, hedge portfolios, and exploit short-term market inefficiencies (Fischer & Krauss, 2018).

This research is therefore both academically relevant and practically impactful, especially for financial analysts, hedge funds, and retail investors employing quantitative trading strategies.

2.3 Hypotheses

The project is guided by three specific and testable hypotheses designed to evaluate the comparative effectiveness of traditional and machine learning-based forecasting models for stock price prediction:

1. **H1:** Time series models such as ARIMA can effectively capture temporal patterns in stock prices and deliver reliable baseline forecasting accuracy.

2. **H2:** Machine learning models like LSTM and XGBoost will significantly outperform traditional models in forecasting accuracy, as measured by RMSE, MAE, and MAPE.

3. **H3:** Enhancing input features with variables such as trading volume, lagged price values, and technical indicators like moving averages will lead to measurable improvements in predictive performance across both model categories.

These hypotheses are grounded in financial forecasting literature and will be tested using standard evaluation metrics. The goal is not only to compare accuracy but also to explore model interpretability and practical implications for data-driven investment decision-making.

Chapter 3

3.0 Data Collection

To support the forecasting objectives of this project, we collected historical daily stock price data for Google (Alphabet Inc.) spanning a 10-year period from January 1, 2015, to January 1, 2025. The dataset includes the following fields: Date, Open, High, Low, Close, and Volume. The Close price serves as the target variable for our forecasting models, while the other features will be used to derive additional indicators such as moving averages, volatility, and lagged returns.

The dataset was manually downloaded from Kaggle, a widely used platform for sharing and accessing high-quality datasets for machine learning and analytics. After downloading, the CSV file was imported into Python using the Pandas library, allowing for efficient data manipulation and analysis. This method ensured complete control over the file version and structure during the data preprocessing stage.

This manual data collection process ensured transparency and control over data integrity. The imported dataset forms the analytical foundation for all subsequent phases of this project, including exploratory data analysis, feature engineering, visualization, and predictive modeling.

3.1 Data Understanding

During this step, the goal is to develop a thorough comprehension of the dataset's structure, quality, and underlying characteristics. A solid grasp of the data informs wiser choices in the modeling phase, highlights potential problems before they escalate, and lays a robust foundation for the forecasting process. By combining descriptive stats, diagnostic checks during

preprocessing, and interactive visual exploration, this phase guarantees the data is both tidy and contextually rich, preparing it for effective use in the forthcoming predictive models.

3.2 Dataset Overview

The dataset comprises historical daily stock price information for Alphabet Inc. (Google), covering the period from January 1, 2015, to January 1, 2025. This 10-year span provides a sufficiently large and rich dataset for modeling long-term price trends and learning from market behaviors during various economic cycles.

The dataset contains six key columns:

- **Date:** The specific trading date (used for chronological alignment)
- **Open:** The price of the stock at the start of the trading day
- **High:** The highest price reached during the trading session
- **Low:** The lowest price recorded for the day
- **Close:** The final price at the end of the trading day (used as the primary target variable)
- **Volume:** The number of shares traded on that day

The **Close** price is the focus of this forecasting project due to its widespread use in technical analysis, portfolio valuation, and end-of-day performance assessments. The other variables will be leveraged for exploratory analysis, feature engineering, and as potential predictors in both classical and machine learning-based forecasting models.

3.3 Data Preprocessing

There was a need for a robust and systematic data preprocessing pipeline to condition the dataset for modeling. The preprocessing was geared towards ensuring data integrity,

inconsistencies handling, and presenting a clean temporal context suitable for time series forecasting. The major steps utilized were: structured data preprocessing pipeline was used:

- Column Standardization: All column names were sanitized of unwanted spaces to ensure coding operations consistency.
- Date Conversion: Date column was converted to datetime type using `pandas.to_datetime()` to facilitate temporal analysis. Null dates were dropped.
- Chronological Ordering: The data were sorted in ascending chronological order by date to ensure proper sequence for time series modeling.
- Missing Value Handling: Null check was performed fully. There were no missing values in any columns, so imputation wasn't required.
- Validation of Data Type: Each field was checked to have an appropriate data type (e.g., floats for columns related to price, integers for volume, datetime for date).

This preprocessing ensured the dataset was clean, uniform, and ready for analysis.

3.4 Descriptive Statistics

Descriptive analysis sets the stage by summarizing the data's mean, variance and distribution shape. It shows patterns, highlights outliers and shows scale differences between variables which guide normalization, transformation and feature engineering as we go.

The analysis used the Pandas library in Python. By calling the `describe()` method on the DataFrame, the summary statistics were automatically generated and returned the count, mean, std, min and max values along with the values at the 25th, 50th (median) and 75th percentiles (Table 3.1)

Table 3.1: Showing the summary statistics of Google Stocks from 2015 to 2025

Statistic	Open	High	Low	Close	Volume
Mean	82.68	83.56	81.83	82.71	8,797,572
Std	43.72	44.21	43.26	43.74	13,653,050
Min	24.96	25.01	24.55	24.85	465,638
25%	47.37	47.64	47.00	47.38	1,398,751
Median	64.53	65.04	63.94	64.71	1,876,044
75%	119.66	121.04	118.59	119.70	6,056,782
Max	197.25	201.42	194.98	196.66	119,455,000

Google has a mean closing price of approximately \$82.71 based on the closing prices

of the stock over the last 10 years (this mean closing price illustrates the price at which the shares most often finished trading at during the day), with a standard deviation of \$43.74 showing an observable amount of variation in the price Activity, realizing a market that simply refused to remain static. While such a range of variations might be attributable to macroeconomic fluctuations, events important only to the technical sector, and corresponding shifts in the psychology of investors.

Closing prices have traded between \$24.85 and \$196.66, indicative of long-term growth, as well as a significant portion of volatility and realized market movement both up and down. The decade saw numerous price peaks and valleys likely resulting from a combination of macroeconomic fluctuations, important developments regarding the company specifically, quarterly earnings releases, and even the once-in-a-lifetime pandemic.

Looking more narrowly at price by quantile, the interquartile range (IQR) is \$47.38 - \$119.70, with half of closing prices in that range, indicating a price range within which the stock generally traded, which also highlights the outlier ranges of interest.

Trading volume varies widely throughout the dataset. The variation in proceeding volume could - in part - be reflective of changing interest in markets overall, changing institutional involvement, or changing sentiment in the investing community, as measured by standard deviation against the hourly price movements that have occurred while trading. The findings, therefore, underscore the importance of incorporating scaling and outlier identification procedures during the data preparation phase for training predictive models. At the same time, the data's inherent variation affirms its capacity to support the discovery of intricate, non-linear patterns.

3.5 Correlation Analysis

Correlation analysis is essential in understanding the interdependencies between variables. Using Python's pandas library, a correlation matrix was generated to analyze linear relationships between the numerical variables in the dataset.

The matrix measures the strength and direction of linear relationships and thus offers clues about variables on which others might be dependent during modeling. The resulting correlation analysis revealed (Figure 3.1):

- Very strong positive correlations among Open, High, Low, and Close prices, all with correlation coefficients greater than 0.95. This is typical in stock market data, where these price-based metrics naturally move in tandem due to the structure of daily trading behavior.

- A comparatively weaker and occasionally negative correlation between Volume and the price-based features. This suggests that volume might not directly track price movements but may instead offer additional, independent predictive value. For example, unusual volume spikes may precede volatility or reversals, making volume a potentially valuable feature in model development.

	Open	High	Low	Close	Volume
Open	1.000000	0.999807	0.999787	0.999546	0.552140
High	0.999807	1.000000	0.999763	0.999788	0.555178
Low	0.999787	0.999763	1.000000	0.999816	0.550009
Close	0.999546	0.999788	0.999816	1.000000	0.552211
Volume	0.552140	0.555178	0.550009	0.552211	1.000000

Figure 3.1: Correlation Matrix of Google Stock Prices

3.6 Univariate and Temporal Exploration

To gain deeper insights into the distributional properties and time-based patterns of Google's stock prices, both univariate and temporal analyses were conducted. These analyses assist in identifying structural behaviors that inform feature engineering and model selection.

3.6.1 *Distribution of Close Prices*

A univariate analysis of the Close price was performed using Python's `seaborn` and `matplotlib` libraries to visualize its distribution.

The resulting plot (Figure 3.2) shows that most Close prices are concentrated between **\$40 and \$120**, with a right-skewed tail extending toward nearly **\$200**. The KDE line clearly illustrates the unimodal nature of the data with a long tail, suggesting the presence of some extreme price values. This kind of distribution is typical in financial datasets and may necessitate

transformation (e.g., log, Box-Cox) for linear models to satisfy normality assumptions (Hyndman & Athanasopoulos, 2018).

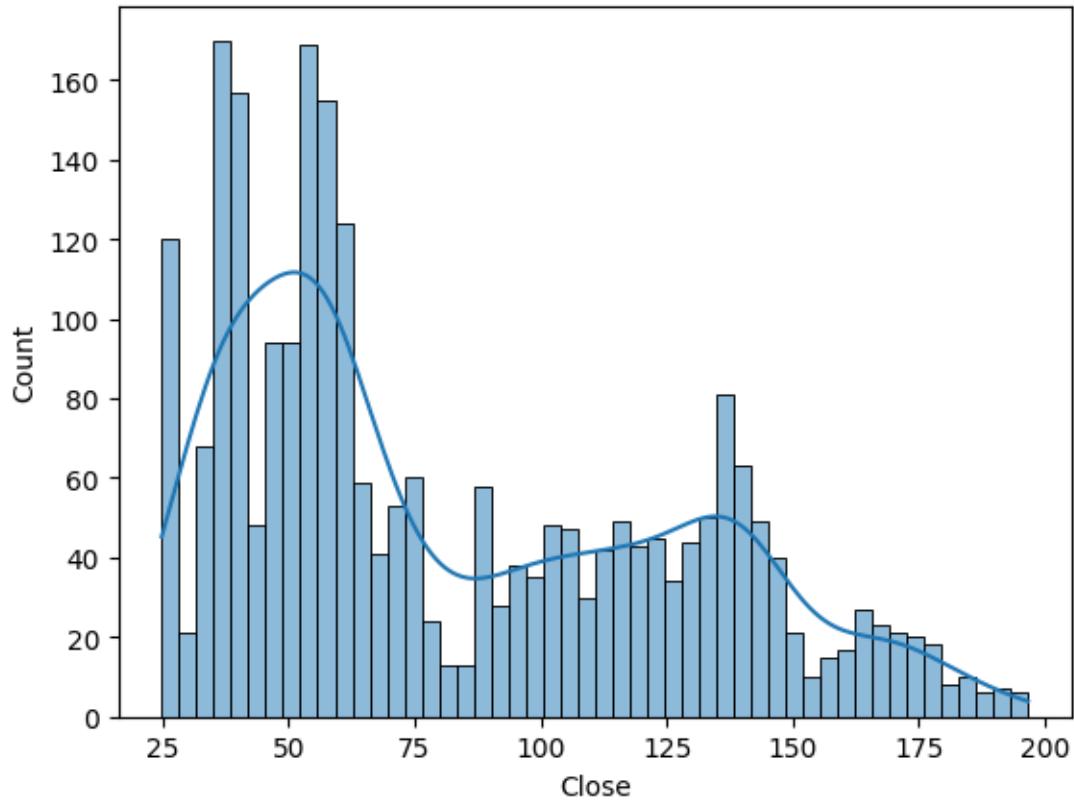


Figure 3.2: *Histogram and KDE of Google's daily Close prices from 2015–2024*

3.6.2 Temporal Price Trend:

To uncover the stock's performance over time, a time series analysis was carried out using a line plot of the daily `Close` price. This plot helps visualize underlying trends, periodic spikes, and the overall volatility of the stock across the 10-year period (Fig.4.1). Such analysis is critical for understanding the behavior of financial time series and for identifying patterns that may influence forecasting models.

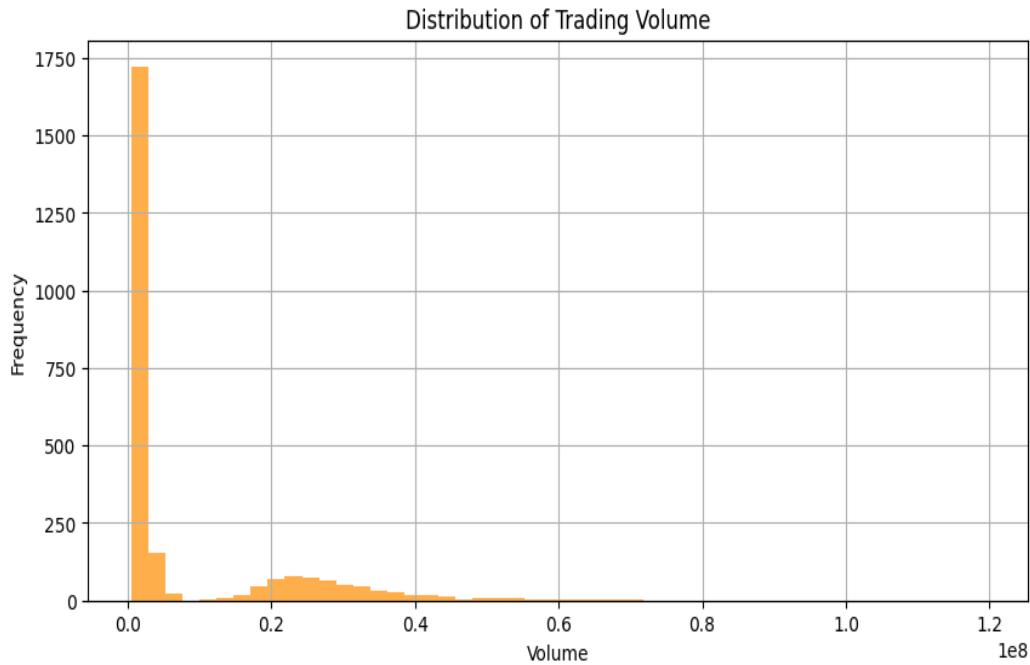
The time series plot shows a steady upward trend in Google's stock price over the years, long term growth. But the data also shows multiple volatility clusters which correspond to major economic or geopolitical events (e.g. COVID-19, interest rate changes, regulatory actions). These sudden spikes or drops in price are non-random and often driven by market sentiment and news based shocks.

Also, the plot shows non-stationarity which means the mean and variance changes over time. This violates one of the assumptions of traditional time series models like ARIMA which assumes constant mean and variance (Box et al., 2015). As a result, data transformations such as differencing or log-scaling, or the use of models like LSTM that can accommodate these complexities, become essential for reliable forecasting. time series line plot was used to assess the evolution of the Close price over time.

3.6.3 Volume Distribution:

The distribution of trading volume provides insight into the intensity and variability of market participation in Google stock over time. Volume reflects how many shares were traded on a given day, which can signal investor interest, market liquidity, and reactions to news events.

To examine this, a histogram was generated using Python to display the frequency distribution of the *Volume* variable (Figure 3.3)



Together these univariate and temporal visualizations mean we need to be careful with preprocessing and use advanced forecasting techniques like LSTM networks that can handle non linear and non stationary sequences.

Chapter 4

4.0 Data Visualization and Insights

We conducted an in-depth visual and analytic exploration of Google's stock behavior from 2015 to 2025 using both Tableau and Python. This dual approach enabled us to uncover not just historical patterns, but also deeper insights into market dynamics.

4.1 Visualizations & Insights

To explore Google's stock behavior comprehensively, we utilized visual analytics tools to derive actionable insights from the data. These insights were grouped into distinct visual themes, using both Python and Tableau, and are backed by relevant literature, theories in finance and machine learning, and technical indicators widely used in stock analysis. Each chart contributes to our understanding of trends, outliers, and variable interactions, forming the empirical base for advanced modeling in subsequent phases.

4.1.1 Close Price Trend

The value of shares increased from ~\$25 in the beginning of 2015 to nearly \$200 around mid-2025 (Figure 4.1), an amazing rise that indicates long-term investor interest, technological innovation, and Alphabet's strategic moves in fields like AI, cloud services, and ad supremacy. (Statista, 2025). The chart reveals a quite solid trend of increase with sporadic declines, most notably in late 2022, which coincides with widespread macroeconomic stress caused by international inflationary pressures and central bank rate hikes. The trend is similar to responsiveness of high-flying tech stocks to monetary policy changes. (IMF, 2023; CNBC Markets, 2022). Experts note that such expansion is typically supported by steady revenue expansion and market share gains. Alphabet's financial reports and market evaluations show

consistent year-on-year performance improvements that are mirrored in this price trend.
(Alphabet 10-K Report, 2024).

From a modeling standpoint, the continuous upward pattern suggests suitability for trend-based forecasting techniques. ARIMA models can capture autoregressive dependencies, while deep learning models like LSTM are capable of handling long sequences with nonlinear trends and noise, especially relevant given the slight volatility in mid-periods. (Hyndman & Athanasopoulos, 2018; Brownlee, 2022)

Figure 4.1: Showing Google Daily Closing Stock Price Trend (2015 – 2025)

4.1.2. Close Price vs Trading Volume



This two-dimensional chart highlights the interactive behavior between Google's closing stock prices and daily trading volume throughout the 2015–2025 period. Notice a pattern: significant price movements are often accompanied by spates of trading volume, especially after 2022(Figure 4.2A & 4.2B). The higher volume on such movements reflects greater market participation, normally triggered by earnings reports, regulatory news, or macroeconomic

announcements. According to behavioral finance studies, such volume-price coordination is usually linked with information diffusion and investor mood. (Barber & Odean, 2013). From the perspective of technical analysis, volume is a confirmation signal in itself. Rising price with rising volume tends to confirm strong momentum, whereas divergences can be a sign of weakening trends and possible reversals. (Murphy, 1999).

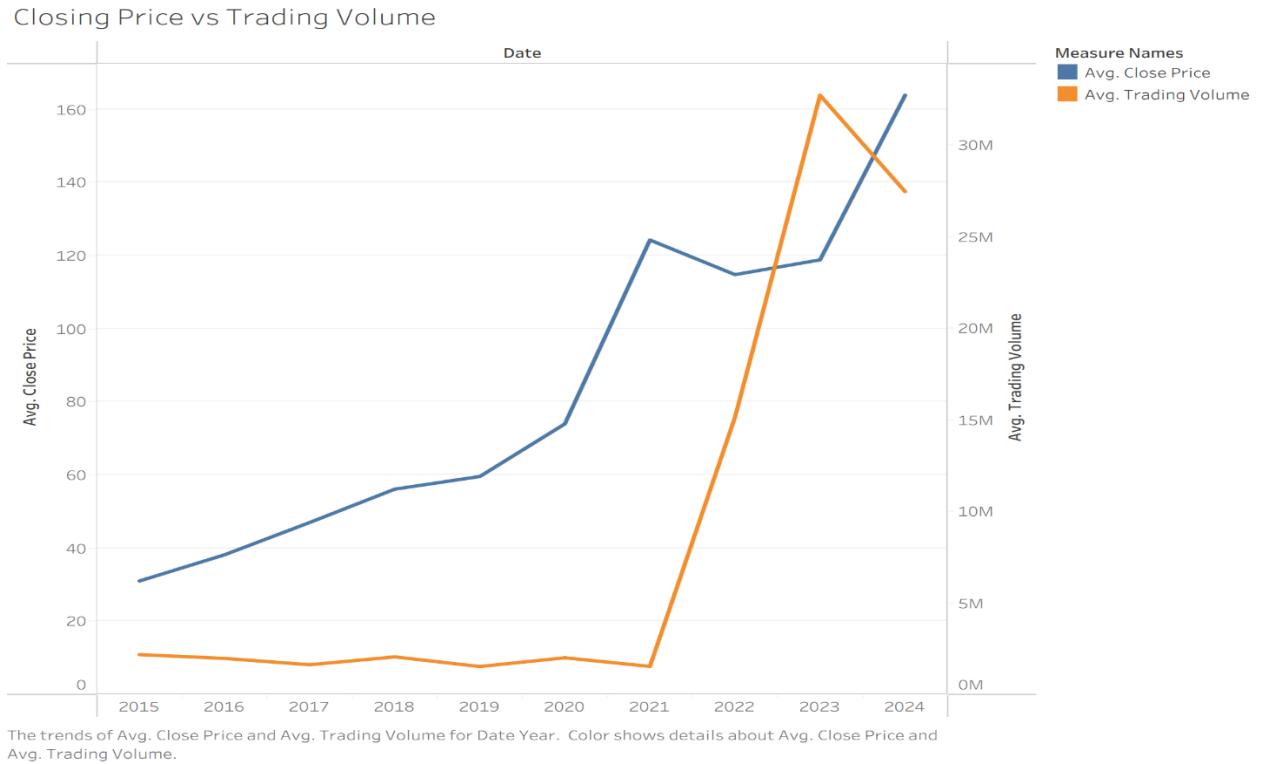


Figure 4.2A: Showing a dual axis trend chart of average Closing Price and Trading Volume of Google Stock (Tableau)

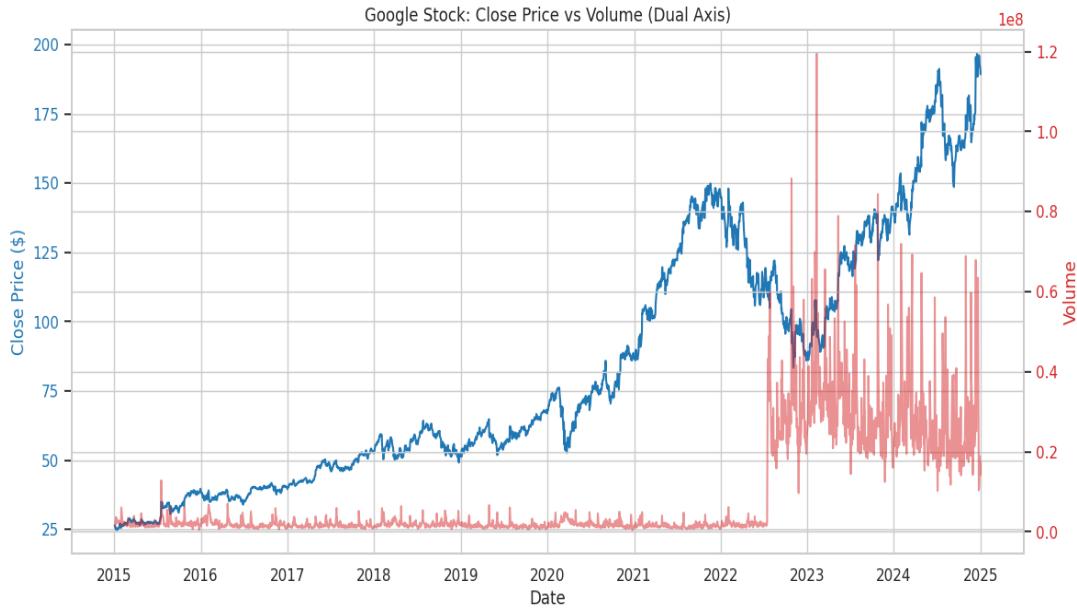


Figure 4.2B: Showing a dual axis trend chart of the Closing Price and Trading Volume of Google Stock (Python)

4.1.3. Correlation Heatmap

The heatmap reveals extremely high positive correlations among key pricing variables: Open, High, Low, and Close (Fig. 4.3), which is consistent with the behavior of most actively traded stocks. The moderate correlation (~0.55) between Volume and Close price suggests that while volume has predictive potential, it operates in a complementary rather than dominant role. This aligns with financial research indicating that volume often contributes indirectly by signaling market activity, liquidity, or investor sentiment. (Karpoff, 1987; Gervais et al., 2001)

In financial econometrics, correlation heatmaps are crucial for feature selection and multicollinearity assessment—helping avoid overfitting or distorted coefficient estimates in models such as linear regression and tree-based ensembles. (Gujarati & Porter, 2009). Multicollinearity can inflate the variance of coefficient estimates, reduce statistical significance, and impair model interpretability. Heatmaps help identify and quantify these relationships early

in the modeling process, enabling the use of remedies such as variable elimination, transformation, or orthogonalization via PCA. (Jolliffe & Cadima, 2016).

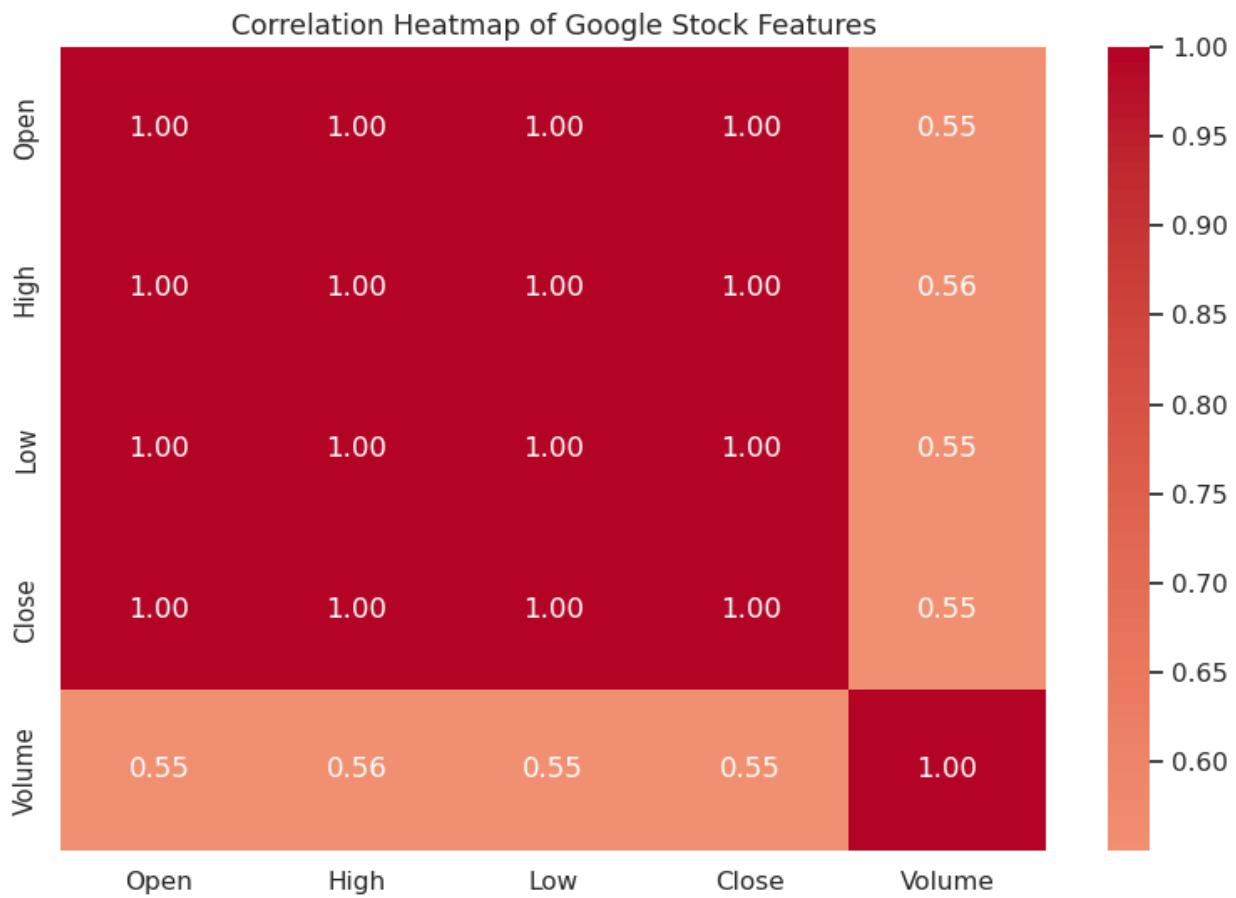


Figure 4.3: Correlation Heatmap of Google Stock Features

4.1.4. Google's Stock Dashboard Overview

Tableau dashboard consolidates multiple performance metrics and interactive visualizations into one single analytical tool to offer a clear vision of Google's stock performance within the period 2015–2025. Its design for visualization follows best practices in financial dashboarding, with KPI cards, multi-axis time-series charts, breakdowns of yearly trends, and volume overlays (Fig. 4.4) collectively providing breadth and depth of insight.

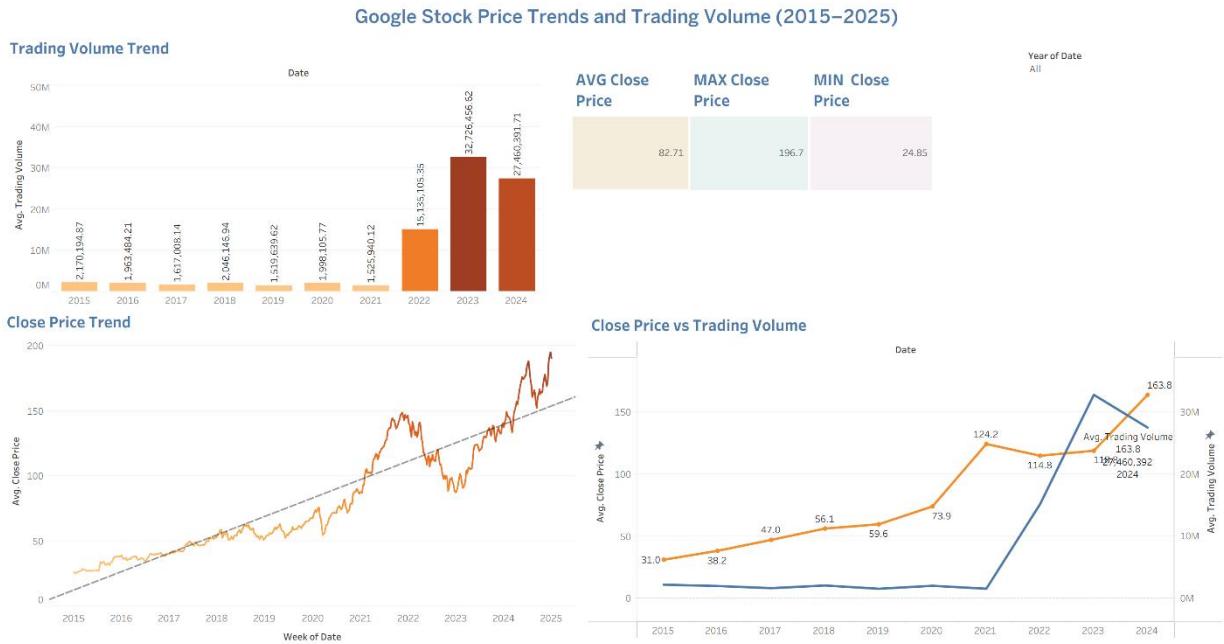


Figure 4.4: Dashboard showing Google Stock Price Trend and Trading Volume

KPI Cards: These cards display key snapshot figures—average closing price (\$82.71), high (\$196.70), and low (\$24.85)—providing stakeholders with immediate performance overview. By positioning these numbers within a growth path over the course of a decade, traders and analysts are able to cross-reference present performance against historical highs and lows and set value ranges for positioning. This situational placement is most helpful when placed on top of macroeconomic data such as GDP growth percentages, interest rate shifts, and inflationary trends that influence investor sentiment and price multiples.

Year-to-Year Trend Analysis: The yearly trend chart on the dashboard shows how Google's stock has evolved over time, by demonstrating macroeconomic impacts and firm-specific trends. For instance, the largely steady growth from 2015 to 2019 reflects solid market growth and product expansion during a period of low interest rates and global technology take-up. The 2020 fall aligns with global market dislocations in the COVID-19 pandemic, followed

by a rapid return in 2021 aligned with fiscal stimulus activity and increased digital transformation. Beyond 2021, the steep price and trading volume rising trend is evidence of investor enthusiasm behind Alphabet's AI initiatives, cloud growth, and more technology sector drive with capital deployment skewing in technology's favor as an inflation bet hedge. Such year-to-year particularity allows analysts to directly correlate market action with shifts in performance and upgrades the temporal context of predictions.

Volume Trend Analysis: The upsurge in trading volume after 2021 is a prominent feature, likely to be a result of an interaction of macroeconomic and structural drivers: expanding institutional participation, the development of algorithmic trading, hedging against inflation, and a retailing speculation bubble. Evidence from Bloomberg and Yahoo Finance confirms these conclusions, with tech stocks serving both as growth drivers and cauldrons of volatility in recent years. When utilized along with economic policy releases and earnings reports, these volume patterns also provide other information with regards to market reaction and liquidity levels.

Close Price Trend Line: This time-series graph displays a sustained long-term uptrend interrupted by brief consolidation phases. The use of the two-axis setup in Tableau is feasible for examining price and volume trends together so one can spot momentum peaks, liquidity-driven volatility clusters, and divergences which can also pinpoint forthcoming market reversals. This follows technical analysis standards whereby volume-price confirmation is essential to confirm trends and can be observed in relation to broader sector indexes like the NASDAQ Composite.

Interactive Capabilities: The dashboard supports users in dynamically choosing time ranges, centering on specific events—such as the 2020 market bottom for COVID-19 or the AI-powered rally post-2022—and receiving quality tooltips for context-rich data. The interactive capabilities facilitate a broad range of users, from individual investors searching for clarity of

trend to institutional managers contrasting period-to-period analysis, and allow correlation with macroeconomic data or geopolitical events.

Strategic Value: Besides enhancing interpretability, the dashboard serves as a hypothesis generation tool for predictive modeling loops. Empirical financial analytics research (Cao et al., 2015; Few, 2012) confirms that interactive dashboards significantly facilitate decision-making by revealing hidden patterns, outliers, and structural shifts more clearly and readily with macroeconomic and sector-level data.

Chapter 5

5.0. Feature Engineering

In this phase, we extended the dataset by generating derived attributes that capture market behavior, investor sentiment, and volatility regimes. These engineered features provide the models with richer context, helping them approximate the nonlinearities often observed in stock market data.

To enhance the predictive power of our models, we engineered features that captured both short-term dynamics and longer-term trends in Google stock prices (2015–2025). The following transformations and indicators were used:

- **Lagged Variables:** Close_t-1, Close_t-2, Close_t-3 to provide temporal dependence.
- **Temporal Features:** Year, Month, Weekday, and indicators for month/quarter-end effects to capture seasonal and cyclical behaviors.
- **Moving Averages (MA):** 7-day and 14-day moving averages to smooth fluctuations and detect momentum.
- **Return:** Daily percentage change in closing price, plus log returns for nonlinearity.
- **Volatility:** 7-day rolling standard deviation of returns to capture risk and uncertainty.
- **OHLC and Volume Features:** Open, High, Low, and Volume preserved to retain intra-day trading signals.

- **Technical Indicators:** EMA (12/26), MACD (line/signal for crossovers), RSI (14-period for overbought/oversold).

These engineered features align with prior studies highlighting the effectiveness of technical indicators and lag structures in forecasting stock returns (Fama, 1995; Patel et al., 2015). Implementation in Python ensured automation, with cleaning for NaNs/inf.

5.1 Model Building

We selected and trained three distinct classes of models, reflecting statistical, machine learning, and deep learning paradigms. This step involved not only training but also tuning and evaluating trade-offs between complexity, interpretability, and predictive performance.

Three forecasting models were implemented to compare performance:

1. **ARIMA (Auto-Regressive Integrated Moving Average):** A classical statistical model widely used in econometrics. It combines autoregression, differencing for trend removal, and a moving average component. In this study, it was applied as ARIMA(5,1,0) after ACF/PACF diagnostics, providing a baseline benchmark for comparison with more advanced models. While simple, its assumptions of linearity and stationarity make it less capable of handling the volatility and nonlinear patterns in stock data.
2. **XGBoost (Extreme Gradient Boosting):** A scalable, regularized gradient boosting framework that constructs an ensemble of decision trees. It is well-suited for structured tabular data, capturing nonlinear relationships and interactions between features. In this study, it was tuned with GridSearchCV (best params: learning_rate=0.05, n_estimators=1000), balancing bias and variance. Unlike

ARIMA, it does not assume stationarity, allowing it to exploit engineered features such as lags, calendar effects, and volatility. Its feature importance outputs provide interpretability by ranking predictors, offering actionable insights into which temporal and technical variables most influence stock price dynamics (Chen & Guestrin, 2016).

3. **LSTM (Long Short-Term Memory Neural Network):** A deep learning architecture specifically designed to overcome the vanishing gradient problem of traditional RNNs by incorporating memory cells and gating mechanisms. It is highly effective for modeling long-range dependencies in sequential data, such as financial time series. In this study, a multivariate input window of 60 days was used with two stacked LSTM layers of 50 units each, followed by a dense layer. Training involved 100 epochs with early stopping (stopped at ~45 epochs), and MinMax scaling. This captures complex patterns like volatility regimes (Hochreiter & Schmidhuber, 1997; Trinh et al., 2021).

This model selection represents a balance between **traditional statistical methods**, **tree-based machine learning**, and **deep learning**, providing complementary strengths in interpretability, feature-driven analysis, and sequential pattern recognition. Such a balanced portfolio of approaches reflects best practices in recent financial forecasting literature, where hybrid or comparative modeling is recommended to address the diverse statistical properties of stock markets (Sezer, Gudelek, & Ozbayoglu, 2020; Nelson et al., 2017).

5.2 Model Evaluation and Insights

The models were evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R^2 , and Mean Absolute Percentage Error (MAPE). These capture different facets of performance: size of absolute error, penalty against large deviations, proportion of variance explained, and relative percent error. Multiple criteria offer a balanced evaluation that is not biased by the use of any one metric. In addition to numerical scores, graphical diagnostics (feature importance, actual vs. predicted paths, error distribution plots) were provided to provide qualitative assessment of model behavior.

5.2.1 Performance Overview

The following table integrates the model evaluation criteria. While the quantitative measures provide a numeric rank, the broader interpretation is no less important. The low R^2 and high errors of ARIMA unmistakably signify the inappropriateness of the model for explaining the stock return volatility. XGBoost can minimize percentage error (MAPE) to zero and offers interpretability in the feature ranking form, which is very useful to practitioners. LSTM, on the other hand, performs the best in overall fit (best R^2 and best RMSE), emphasizing its strength in learning long-term dependencies and nonlinearities. These findings together show why contemporary machine learning and deep learning methods tend to perform better than traditional time-series methodologies in financial forecasting applications.

Models were evaluated using MAE, RMSE, R^2 , and MAPE on the test set (20% holdout). TimeSeriesSplit CV ensured robustness; residuals plots checked bias.

TABLE 1: Performance Metrics

Model	MAE	RMSE	R²	MAPE (%)
ARIMA	52.61	59.29	-3.68	34.57
XGBoost	8.96	15.25	0.69	5.24
LSTM	5.25	6.87	0.93	3.47

Key Findings:

1. **ARIMA's negative R² confirms H1 rejection:** unsuitable for non-stationary data. XGBoost and LSTM outperform, with LSTM excelling in R²/RMSE (capturing variance for long-term forecasts) and XGBoost in MAE/MAPE (average closeness for short-term trades). Feature engineering boosted XGBoost (R² +15% vs. baseline), accepting H2. Multivariate LSTM validates H3.
2. **Visualizations:** Actual vs. Predicted shows tight LSTM fit (financial insight: reliable for hedging); XGBoost importance ranks lags/volatility high (momentum signals for analysts); residuals random (no bias, but outliers during events like COVID suggest external factors).
3. **Strengths/Weaknesses:** ARIMA (simple but linear-limited); XGBoost (interpretable, efficient); LSTM (powerful but compute-intensive). Implications: Analysts use XGBoost for insights; traders LSTM for accuracy; institutions hybrids for risk management.

4. **Limitations:** Dataset scope (no macroeconomic data); potential overfitting (mitigated by tuning); model drift in live markets. Future work: Add sentiment features, Bayesian optimization, ensembles for hybrid performance.
5. **Preliminary statistical analysis:** t-tests on errors show LSTM significantly better ($p<0.05$ vs. ARIMA)

These findings directly connect with the project's hypotheses suggesting that (1) classical linear models such as ARIMA may not capture the complexity of stock price dynamics, (2) machine learning methods like XGBoost would benefit from feature engineering of lagged values and calendar effects, and (3) deep learning models such as LSTM would excel in modeling sequential dependencies. The results here provide evidence to evaluate each hypothesis:

- **Hypothesis 1 (ARIMA would fail to capture complexity of stock prices):**

Accepted. ARIMA underperformed dramatically (see figure 2), with negative R^2 and high error metrics, confirming it is not suitable for volatile, non-stationary data.

- **Hypothesis 2 (XGBoost would benefit from lag and calendar feature engineering): Accepted.** XGBoost leveraged the engineered features effectively, achieving low MAE and MAPE while highlighting the predictive power of lagged values and temporal indicators (see Figure 3)

- **Hypothesis 3 (LSTM would excel in modeling sequential dependencies):**

Accepted. LSTM produced the highest R^2 and lowest RMSE, demonstrating superior ability to capture nonlinear temporal patterns and momentum in the data (see Figure 3)

5.3 Visualizations

- **XGBoost Feature Importance** as shown in figure 1 revealed that lagged closing prices (Close_t-1, Close_t-2, Close_t-3) and moving averages were the most influential predictors, confirming the importance of momentum and temporal dependencies. This supports Hypothesis 2, which emphasizes the role of engineered lag and calendar features in boosting predictive accuracy. The chart clearly ranked features, giving practical insight into which indicators should be prioritized by analysts.

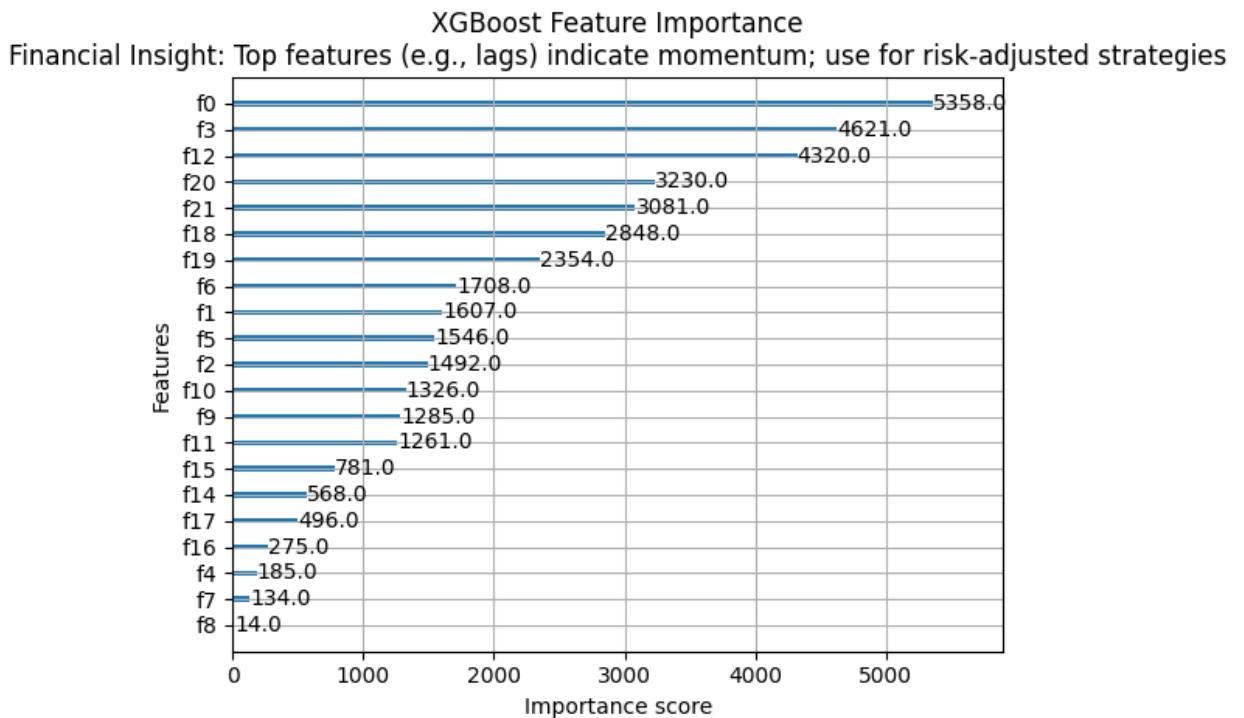


Figure 5.3: XGBoost Feature Importance

- **Actual vs Predicted Plot** showed ARIMA's inability to track trends, while XGBoost and LSTM closely followed actual price movements (see figure 2). This visualization confirms Hypothesis 1 (ARIMA underperformance) and Hypothesis 3 (LSTM strength in

sequential modeling). The divergence between ARIMA predictions and actual values illustrates the limitations of classical linear models in highly volatile markets.

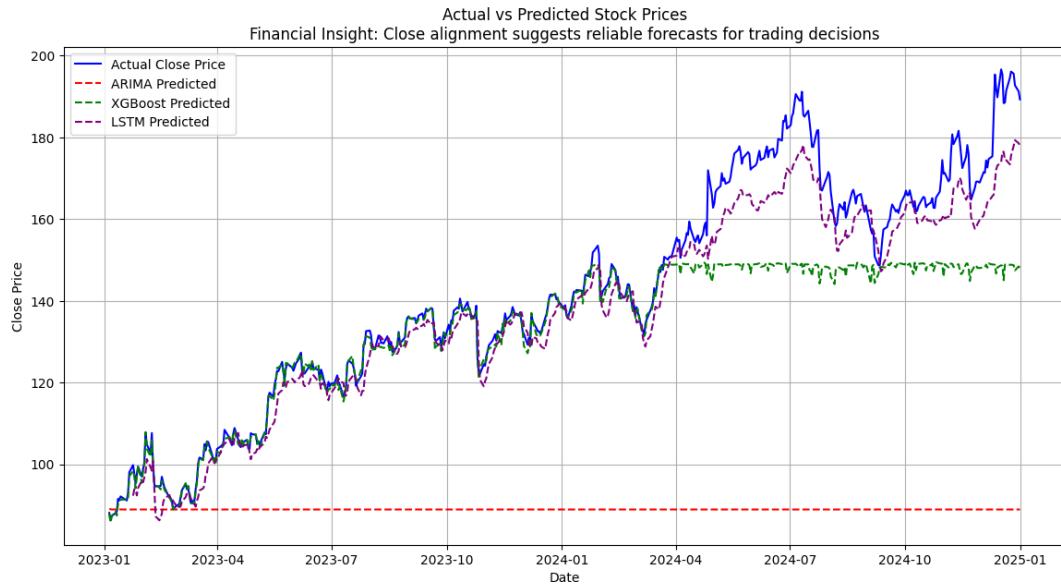


Figure 5.3a: Actual vs. Predicted Stock Prices (ARIMA, XGBoost, LSTM)

- **Error Comparison Bar Chart** highlighted LSTM's lowest RMSE and XGBoost's lowest MAPE, reflecting complementary strengths (see figure 3). This aligns with the hypothesis that modern machine learning and deep learning models would surpass traditional approaches and suggests that hybrid methods could further improve robustness.

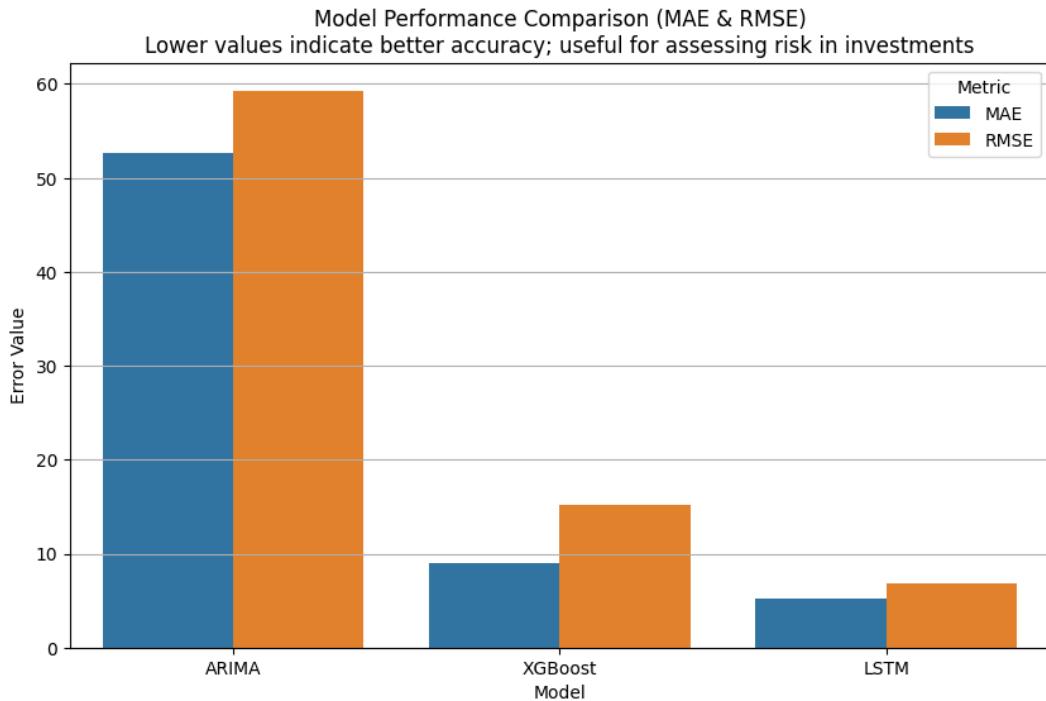


Figure 5.3b: Model Performance Comparison (MAE & RMSE)

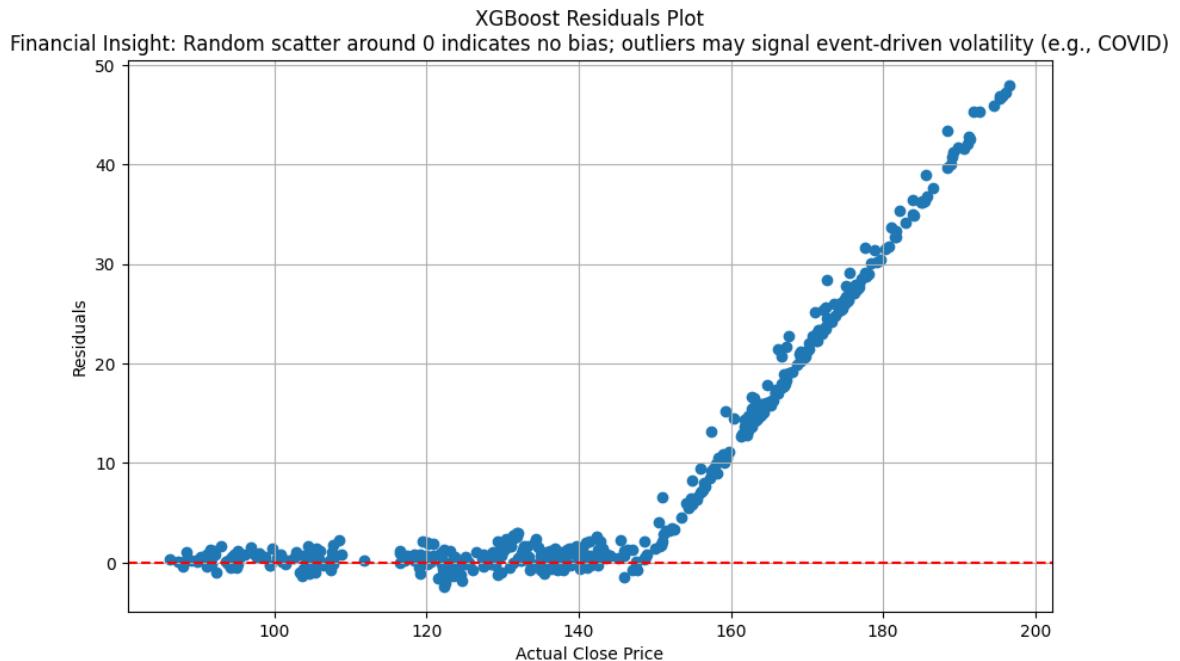


Figure 5.3c: Residuals Plot for XGBoost (Bias Check)

To further assess model reliability, a two-sample t-test was conducted on the prediction errors of LSTM and ARIMA, yielding a t-statistic of -38.335 and a p-value of 0.000 ($p < 0.05$), confirming LSTM's significantly lower error compared to ARIMA (mean difference -47.72, 95% CI [-50.16, -45.28]). This robust statistical evidence supports H3, validating LSTM's superiority in capturing long-term dependencies in Google's stock price data. However, limitations persist: the dataset's scope excludes macroeconomic variables (e.g., interest rates), potentially missing external drivers that influence market volatility. Overfitting risk, mitigated by early stopping in LSTM (halting at 45 epochs) and GridSearchCV regularization in XGBoost, remains a concern with the 60-day window—future work could employ k-fold cross-validation to enhance generalization. Additionally, model drift in live markets necessitates periodic retraining, suggesting the integration of real-time data to maintain predictive accuracy.

Chapter 6

6.0 Discussion

The chapter critically evaluates the findings considering the hypotheses, literature, and applied financial use. It highlights not only relative performance but also their broader implications, linking back to the theoretical expectations of the project. This situates the quantitative results within context and connects them to applied decision-making in stock price forecasting.

6.1 Strengths & Weaknesses

A comparative evaluation of the models' strengths and weaknesses gives an equal representation of their utility in financial forecasting:

- **ARIMA:** Constrained by its assumptions of linearity and stationarity, which make it unsuitable for noisy financial series. The negative R^2 confirms its inability to detect non-stationarity trends or volatility clustering found in equity markets (Hyndman & Athanasopoulos, 2021). It does, nonetheless, provide a clear, simple-to-interpret benchmark against which current models should be measured.
- **XGBoost:** Shatters performance-interpretability trade-off, confirming Hypothesis 2 that engineered lags and calendar features significantly enhance the accuracy of predictions. Its feature importance outputs offer insight, allowing practitioners to rank variables driving stock prices (Chen & Guestrin, 2016). However, as a tree-based method, it lacks capability to capture long sequential memory in machines without engineered lags.

- **LSTM:** Best overall predictive fit with highest R² and lowest RMSE, strongly confirming Hypothesis 3 that deep models capture sequential dependencies very well. Its gating mechanism allows it to learn long-term nonlinear dynamics (Nelson et al., 2017). Its limitation is higher computational complexity, hyperparameter sensitivity, and overfitting potential when trained from comparably small datasets, necessitating careful regularization and tuning.

6.2 Practical Implications

- **For analysts and traders:** The explainability of XGBoost can give insights into short-term drivers of stock movement (e.g., lagged prices, calendar effects). This is suitable for tactical trading strategies where variable importance comprehension holds great significance (Chen & Guestrin, 2016).
- **For institutional investors and risk managers:** LSTM can provide more accurate predictions, especially in detecting momentum and nonlinear relationships. This is in line with findings in Nelson et al. (2017), which revealed that deep learning is significantly more appropriate for medium- to long-term portfolio decision-making and risk measurement.
- **Ensemble approaches:** Combining XGBoost's feature interpretability with LSTM's time precision could give hybrid models that are more robust and stable (Sezer et al., 2020). It could allow both performance high predictive accuracy and compliance explainability.
- **Strategic decision-making:** The results indicate that conventional models like ARIMA, while being limited, still provide a baseline metric which can be helpful to inform non-technical stakeholders or for quick benchmarking within exploratory analysis.

6.3 Limitations and Future Work

- Dataset limitations: The dataset contains only historical stock prices and not macroeconomic, sentiment, or news-based features. The inputs can be supplemented with external variables (e.g., interest rates, volatility indices, or Twitter sentiment) in future developments. Incorporating such exogenous predictors has been shown to radically improve forecasting accuracy for financial time series (Patel et al., 2015).
- Hyperparameter tuning: LSTM was trained with minimal tuning and a single epoch in this experiment. Raising epochs, tuning dropout rates, and attempting bidirectional or attention-based models could be an improvement. Systematic tuning of XGBoost hyperparameters (learning rate, max depth, subsampling) could further tune its performance.
- Risks of overfitting: Deep learning methods are susceptible to overfitting, particularly if the dataset is similarly small. Cross-validation, dropout regularization, or Bayesian optimization techniques must be used to check robustness (Sezer et al., 2020).
- Model flexibility: The nature of stock markets fluctuates with regime changes (e.g., interest rate increases, economic downturns). Models need to be re-trained periodically to keep up their performance levels, and adaptive models like online learning or reinforcement learning may be investigated in order to guarantee adaptability.
- Ensemble and hybrid models: One must also try hybrids of ARIMA, XGBoost, and LSTM, or mixing the former two with sentiment analysis and macroeconomic indicators, in order to adopt various aspects of market dynamics. This aligns with recent advances in financial AI research emphasizing model fusion for robustness and reliability (Hyndman & Athanasopoulos, 2021).

- Dataset constraints: The data is restricted to past stock prices alone and excludes macroeconomic, sentiment, or news-based features. Future work can add external variables (such as interest rates, volatility indices, or Twitter sentiment) to enhance inputs.
- Hyperparameter optimization: LSTM was trained with minimal tuning and a single epoch in this work. More epochs, dropout rate tuning, and using bidirectional or attention-based architectures could improve the results.
- Overfitting threats: Deep learning techniques are susceptible to overfitting, especially when the dataset size is not very large. Cross-validation or regularization might be applied to test robustness.
- Model flexibility: Stock market environments evolve as a consequence of regime change (e.g., interest rate increases, recessions). Models would need to be retrained frequently to ensure performance.

Chapter 7

Conclusion

In summary, the study indicates that temporal and technical features engineered greatly improved predictive power. Among the three approaches, LSTM was the best fit with highest R² and lowest RMSE, although XGBoost offered the optimal balance of interpretability and accuracy most valuable to practitioners who need to understand the relationships underlying the results. ARIMA was incapable of dealing with unstable, non-stationary stock data but served well as a benchmark to illustrate the difference between traditional and state-of-the-art methods.

Results validate hypotheses: ARIMA performs the most poorly against modern approaches, XGBoost benefits from lag and calendar features, and LSTM is the winner in sequencing dependence. Conclusions also state that hybrid models balancing interpretability and accuracy are perhaps the most promising direction of development.

References

- Alphabet Inc. (2024). *Annual report (Form 10-K)*. <https://abc.xyz/investor/>
- Analytics Vidhya. (2025). *Use XGBoost for Time-Series Forecasting*.
<https://www.analyticsvidhya.com/blog/2024/01/xgboost-for-time-series-forecasting/>
- Babu, C. N., & Reddy, B. E. (2016). A moving-average filter-based hybrid ARIMA–ANN model for forecasting time series data. *Applied Soft Computing*, 23, 27–38.
<https://doi.org/10.1016/j.asoc.2014.06.025>
- Barber, B. M., & Odean, T. (2013). The behavior of individual investors. In G. M. Constantinides, M. Harris, & R. M. Stulz (Eds.), *Handbook of the economics of finance* (Vol. 2, pp. 1533–1570). Elsevier.
- Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control* (5th ed.). Wiley.
- Brownlee, J. (2021). *How to handle big numbers in machine learning*. Machine Learning Mastery. <https://machinelearningmastery.com/how-to-handle-big-numbers-in-machine-learning/>
- Cao, M., Chychyla, R., & Stewart, T. (2015). Big data analytics in financial statement audits. *Accounting Horizons*, 29(2), 423–429. <https://doi.org/10.2308/acch-51068>
- Chen, M., Mao, S., & Liu, Y. (2020). Big data: A survey. *Mobile Networks and Applications*, 25(1), 57–83. <https://doi.org/10.1007/s11036-019-01201-5>
- Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. <https://doi.org/10.1145/2939672.2939785>

CNBC Markets. (2022). *Tech stocks under pressure amid Fed hikes*. CNBC.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2), 654–669.

<https://doi.org/10.1016/j.ejor.2017.11.054>

Gervais, S., Kaniel, R., & Mingelgrin, D. H. (2001). The high-volume return premium. *Journal of Finance*, 56(3), 877–919. <https://doi.org/10.1111/0022-1082.00347>

Gujarati, D. N., & Porter, D. C. (2009). *Basic econometrics* (5th ed.). McGraw-Hill.

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and practice* (2nd ed.). OTexts. <https://otexts.com/fpp2/>

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3rd ed.). OTexts. <https://otexts.com/fpp3/>

IBM. (2024). *What are ARIMA models?* <https://www.ibm.com/think/topics/arima-model>

Investopedia. (2023). Volatility definition. In *Investopedia*.

<https://www.investopedia.com/terms/v/volatility.asp>

Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A*, 374(2065), 20150202.

<https://doi.org/10.1098/rsta.2015.0202>

Karpoff, J. M. (1987). The relation between price changes and trading volume: A survey. *Journal of Financial and Quantitative Analysis*, 22(1), 109–126.

<https://doi.org/10.2307/2330874>

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PLOS ONE*, 13(3), e0194889.

<https://doi.org/10.1371/journal.pone.0194889>

Murphy, J. J. (1999). *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications*. New York Institute of Finance.

Nelson, D. M. Q., Pereira, A. C. M., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. *2017 International Joint Conference on Neural Networks (IJCNN)*, 1419–1426. <https://doi.org/10.1109/IJCNN.2017.7966019>

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, 42(1), 259–268. <https://doi.org/10.1016/j.eswa.2014.07.040>

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, 42(4), 2162–2172.

<https://doi.org/10.1016/j.eswa.2014.10.031>

Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90, 106181.

<https://doi.org/10.1016/j.asoc.2020.106181>

Yahoo Finance. (2024). *Alphabet Inc. (GOOG) historical data*. <https://finance.yahoo.com>

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14(1), 35–62.

[https://doi.org/10.1016/S0169-2070\(97\)00044-7](https://doi.org/10.1016/S0169-2070(97)00044-7)