A Composite Valuation-Performance Framework for NIFTY 50 Stocks

Table of Contents

Contents

Summary For Executives

The Price-to-Earnings (PE) ratio and the Capital Asset Pricing Model (CAPM), two important instruments that investors commonly use independently, are combined in this study to present a useful framework for assessing NIFTY 50 stocks. Even though both are widely used in financial analysis, depending only on one of them can sometimes give a misleading impression of the actual performance or value of a stock.

The main concept—which is straightforward but effective—is to integrate performance and valuation to obtain a more comprehensive picture. By comparing PE ratios to sector and index averages, the study assesses whether a stock may be overpriced or undervalued. Furthermore, CAPM helps determine an investor's expected return based on the risk level of the stock. These metrics aid in the classification of stocks into four groups: Overvalued & Underperforming, Undervalued & Underperforming, and Undervalued & Underperforming.

ARIMA forecasting adds a forward-looking layer to the analysis by predicting short-term price movements on a limited number of stocks.

The findings are instructive: while high PE ratios can be associated with underperformance and often indicate overvaluation, this is not always the case. Because market behavior is influenced by a variety of factors, including sentiment, industry trends, and general economic conditions, a one-size-fits-all approach is ineffective. A combined perspective based on valuation and performance is more trustworthy for investors to make well-informed decisions.

This framework provides a more realistic, balanced approach to stock evaluation for experts in finance, investment, and policy. It also suggests a future in which predictive tools and more intelligent analytics will be used to supplement traditional metrics like PE, resulting in a more comprehensive approach to stock evaluation.

Chapter 1

Introduction

1.1 Introduction

Investors frequently use well-known instruments, such as the Price-to-Earnings (PE) ratio or models, such as the Capital Asset Pricing Model (CAPM), to help them choose the best stocks. Although these approaches have been tried and tested, they only provide a portion of the picture. Based on its PE ratio, a stock may appear pricey, but it may still be producing solid returns. Conversely, a stock may appear inexpensive but still underperform. The issue is that relying solely on one metric can occasionally be deceptive.

Using the NIFTY 50 as a case study, this study examines that problem in greater detail. The NIFTY 50 offers a good mix of sectors and industries and contains some of India's largest and most well-known companies. This makes it the ideal environment for testing a more comprehensive approach to stock evaluation, one that integrates performance and valuation into a single framework.

The concept is straightforward: to gain a better understanding of whether a stock is truly worth investing in, combine PE-based valuation with CAPM-based expected returns. The stocks can then be divided into four categories: undervalued and underperforming, overvalued and outperforming, undervalued and underperforming, and overvalued and underperforming. Compared to merely examining one side of the issue, this provides a more comprehensive viewpoint.

To take it a step further, this study also employs ARIMA forecasting to examine the potential near-term behavior of specific stocks, particularly those that are both overpriced and underperforming. Making short-term investment decisions is aided by the forward-looking perspective this provides.

The main goal of this study is to demonstrate how combining PE multiples and CAPM can help prevent some of the blind spots that arise when using them separately. A combined strategy simply makes more sense in a market that is changing quickly, like India's, where investor sentiment and worldwide trends both affect stock prices. It's about using all available tools to make more informed, well-rounded decisions.

1.2 Theoretical Background of the Study

The Price-to-Earnings (P/E) ratio is typically one of the first tools mentioned when discussing stock valuation. Because it's easy to use and has been around for ages, it's likely taken for granted at times. Even though tools like the P/E ratio can still tell us a lot when used properly, investors these days frequently chase after complex models and ignore the fundamentals. Refocusing attention on a metric that is both potent and simple to comprehend is the main goal of this section of the study.

Traditional Valuation Methods: Useful, but Not Always Practical

Investors frequently use the following techniques to determine a company's true value:

- **Discounted Cash Flow (DCF)** is exactly what it sounds like. It determines a company's current value by projecting its potential future earnings. The catch? Because of its heavy reliance on assumptions, even minor forecasting errors can cause the entire system to go awry.
- **Comparative Company Analysis (CCA)** is similar to evaluating a business against others in the same industry. Although it's useful for quickly establishing a benchmark, it frequently overlooks the subtleties that set a business apart.
- **Asset-Based Valuation** considers its assets, such as real estate, machinery, or patents. For tech companies or service-based businesses that depend more on ideas than tangible assets, that may not be as effective as it is for other businesses.

Without a doubt, these techniques are effective. However, they also require financial expertise, a lot of data, and time. The P/E ratio is still useful in every investor's toolbox because of this. It's quick, simple, and surprisingly insightful when combined with some background information.

What Makes the P/E Ratio So Handy?

Fundamentally, the P/E ratio aids investors in determining the price they pay for each rupee of a company's profits. People may anticipate rapid future growth from a company with a high P/E ratio. A low one can indicate that something is off or that it's undervalued. It's excellent for comparisons as well. A company may be overpriced if its P/E is significantly higher than the industry average. It could be a hidden gem if it's below. Additionally, because market sentiments fluctuate frequently, shifts in the P/E ratio frequently reflect investor sentiment toward the economy or particular industries.

Even so, the P/E ratio isn't always given the credit it merits, in part because some people think it's "too basic." However, as this study demonstrates, it can be a useful lens for stock analysis if you know how to use it.

The P/E Ratio Isn't Perfect — And That's Okay

The P/E ratio has limitations, just like any other tool. To begin with, it is determined by earnings. Additionally, profits can fluctuate, particularly in erratic markets. The entire ratio can be thrown off by a poor quarter, a one-time expense, or an unexpected profit spike.

Then there is the fact that different industries have different P/E standards. While a manufacturing company with the same P/E might be viewed as excessively expensive, a tech company with a P/E of 50 might still be deemed reasonable. Comparisons can be deceptive if they are made without context.

Of course, more general economic considerations are also important. P/E ratios generally increase during a bull market. Even for essentially sound businesses, they fall during a recession.

For this reason, depending only on the P/E ratio is never a good idea. It works best when combined with other financial indicators to give you a complete picture rather than just one aspect of the situation.

How to Make Better Use of the P/E Ratio

So, how can we maximize the P/E multiple? The first step is to view it as a component of a larger framework rather than a stand-alone fix. For instance:

- **Monitor the ratio's evolution over time:** Is it staying the same, rising, or falling? That frequently conveys more information than the actual number.
- **Make a proper comparison:** Before drawing conclusions, always take into account what is typical for that particular industry.
- **Dig into the quality of earnings.** Is the company's growth a temporary uptick or is it genuine and long-lasting?

The P/E ratio becomes more than just a figure when investors take the time to perform this type of analysis; it becomes a useful tool for making decisions. This study aims to restore that kind of viewpoint to the discussion of stock valuation.

1.3 Importance of the Study

Financial analysts and investors have historically relied on independent measures like the Price-to-Earnings (P/E) ratio or the anticipated return from models like the Capital Asset Pricing Model (CAPM). But depending just on one indicator frequently leads to a distorted or insufficient picture of a stock's actual potential. While a stock with a low P/E ratio may seem appealing but may be underperforming for underlying reasons, a stock with a high P/E ratio may appear overpriced but still offer strong risk-adjusted returns. By combining performance metrics and valuation into a single framework, this study seeks to close this analytical gap, which makes it noteworthy. In doing so, it offers a more data-driven and balanced method of evaluating stocks that are part of the NIFTY 50 index.

- *Enhancing Investment Decision-Making*

  The ability of this research to enhance decision-making among financial institutions, fund managers, and investors is one of its main contributions. The study offers a useful technique for determining whether a stock is reasonably priced and, when adjusted for market risk, justified in terms of its performance by fusing P/E multiples with CAPM-based expected returns. The conventional method of determining whether stocks are overpriced or undervalued based only on valuation multiples frequently ignores whether the stocks are truly producing returns that meet investor expectations. In order to make more logical and knowledgeable investment decisions, the integrated framework created in this study facilitates a more thorough evaluation of a stock's present and prospective position in the market.

- *Addressing Limitations of Traditional Valuation Methods*

  Although they are widely used in the field of financial analysis, traditional valuation methods like asset-based valuation, discounted cash flow (DCF), and comparative company analysis (CCA) have inherent drawbacks. Conversely, the P/E multiple provides a rapid, easily comprehensible, and generally recognized indicator of a company's market value, despite frequently being thought of as being unduly simplistic. The problem, though, is that the P/E multiple by itself is unable to capture performance insights that look ahead. The theoretical robustness of the CAPM and the ease of use and accessibility of the P/E ratio are combined in this study to close that gap and provide a valuation approach that is both methodologically sound and useful in real-world applications.

- *Promoting Awareness and Practical Applications*

  The study also aims to raise awareness of the P/E ratio's ongoing applicability and versatility when used carefully. The study illustrates how P/E multiples can continue to be useful in a range of market conditions and across various industries using a structured quadrant framework that plots valuation against risk-adjusted returns. The framework encourages users to apply context-specific judgment and go beyond preconceived notions. For instance, just as low P/E ratios in cyclical industries might not always indicate value opportunities, high P/E ratios in growth sectors might not always indicate overvaluation. This study provides recommendations for utilizing P/E- based assessments more successfully and avoiding typical misunderstandings in both academic and professional contexts by providing unambiguous, data-supported evidence and clearing up some of the ambiguity surrounding these metrics.

- *Economic and Strategic Benefits*

  By giving market participants a more sophisticated toolkit for stock evaluation, this research helps to improve capital allocation from a wider economic perspective. A strategic advantage in building well-balanced, high-performing portfolios is gained by investors who are better able to discern between stocks that are actually cheap and those that are likely to perform poorly. At the same time, NIFTY 50 companies can gain insight into how their market valuation stacks up against peer companies when risk and performance are taken into account. This knowledge is essential for managing investor sentiment during sectoral rotations, macroeconomic changes, and earnings announcements in addition to corporate strategy and investor relations. The study improves institutional investors', portfolio managers', and corporations' ability to make strategic decisions by providing a dual-metric framework based on empirical analysis.

- *Contribution to Financial Literature and Practice*

  By re-examining a basic but frequently overlooked metric—the P/E multiple—and placing it within a larger analytical framework, this study adds to the corpus of financial literature already in existence. Even though sophisticated valuation models predominate in scholarly discussions, there is a growing need to reconsider the benefits of using easier-to-use tools in an integrated, structured manner. This goal is furthered by the quadrant framework put forth in this study, which provides a reproducible model that can be tested, adjusted, and expanded upon in subsequent studies. Its pragmatic focus also makes it instantly applicable in financial practice, giving analysts and investors a tool that strikes a balance between usefulness and depth. In the end, this study opens the door to a more comprehensive and realistic method of stock evaluation.

1.4 Need of the Study

Investors are inundated with a variety of valuation models and analytical tools in today's dynamic and fast-paced financial markets. Although many of these techniques provide insightful information, their applicability is limited by their steep learning curve and strong presumptions. Despite this complexity, the Price-to-Earnings (P/E) multiple, one of the most fundamental valuation tools, is still surprisingly undervalued in both academic research and actual investment practice. By investigating its application in conjunction with performance models such as the Capital Asset Pricing Model (CAPM), this study seeks to restore the P/E multiple's significance and provide a more comprehensive and useful framework for stock evaluation.

- *Overcoming the Limitations of Traditional Valuation Models*

  Traditional models, such as the Price-to-Book (P/B) ratio, Discounted Cash Flow (DCF), or Residual Income Models (RIM), are highly sensitive to changes in assumptions like growth rates, discount rates, or risk premiums and frequently call for in-depth financial forecasting. These intricacies make them difficult for professionals working in unstable or unpredictable environments as well as for inexperienced investors. In contrast, the P/E multiple provides a simple earnings-linked metric that makes comparisons faster and implementation simpler. This study recognizes these trade-offs and presents the P/E ratio as a workable substitute that, although straightforward, can have significant impact in the appropriate situation.

- *Adapting to the Dynamic Nature of Financial Markets*

  Today's stock markets are impacted by sectoral shifts, investor psychology, geopolitical events, and the swift changes in macroeconomic factors. Traditional models frequently fall short in capturing sentiment in real time in such a dynamic environment. However, because it captures how investors are pricing future earnings in relation to current performance, the P/E multiple acts as a pulse-check for market expectations. By combining the P/E ratio with CAPM-based expected returns, this study takes advantage of that dynamic quality to provide a more precise understanding of which stocks are not only overvalued or undervalued, but also performing well or poorly in relation to their risk profile.

- *Addressing the Complexity of Stock Market Competition*

  Retail investors frequently find themselves at a disadvantage due to the rise of algorithmic trading and institutional dominance in the markets. Stock analysis feels inaccessible to non-professionals due to the complexity of contemporary models. This study fills that gap by providing a framework that combines a commonly used performance measure (CAPM) with a straightforward metric (P/E), making it both practically applicable and intellectually rigorous. This method preserves analytical depth while democratizing access to insightful performance and valuation data.

- *Bridging the Research Gap on the P/E Multiple*

  Even though the P/E ratio is frequently used in investment reports and the media, multi-factor models, machine learning-based forecasts, and behavioural finance have become more popular in academic literature in recent years, frequently displacing traditional metrics. As a result, there is now a clear research gap regarding the continued performance of fundamental tools like the P/E multiple in contemporary market conditions. This study makes a contribution by critically assessing the P/E ratio's continued applicability and incorporating it into a structured quadrant-based framework that links valuation to actual performance results.

1.5 Conceptual Framework of the Study

The conceptual framework of this study, which emphasizes the often-underutilized but powerful Price-to-Earnings (P/E) multiple, is founded on the fusion of fundamental valuation concepts with empirical performance analysis. This approach, which unifies valuation and performance rather than treating them as separate fields, allows investors to evaluate a stock's actual situation through a more comprehensive lens. By combining the simplicity of the P/E ratio with the analytical strength of the Capital Asset Pricing Model (CAPM), the framework aims to guide both academic research and practical investing strategies.

- *Re-evaluating Traditional Valuation Foundations*

  The conceptual framework of this study, which emphasizes the often-underutilized but powerful Price-to-Earnings (P/E) multiple, is founded on the fusion of fundamental valuation concepts with empirical performance analysis. This approach, which unifies valuation and performance rather than treating them as separate fields, allows investors to evaluate a stock's actual situation through a more comprehensive lens. By combining the simplicity of the P/E ratio with the analytical strength of the Capital Asset Pricing Model (CAPM), the framework aims to guide both academic research and practical investing strategies.

- *The P/E Multiple as a Market-Linked Valuation Lens*

  The P/E ratio functions as more than just a price signal; it reflects market expectations, sentiment, and the perceived risk-return trade-off. This layer of the framework explores how the P/E ratio encapsulates both objective financial performance and subjective investor outlook. Whether high or low, a company's P/E ratio provides a snapshot of how the market views its growth potential and profitability. This dual perspective—anchored in earnings yet shaped by perception—makes it a critical metric, especially in times of volatility or uncertainty.

- *Integrating Performance Evaluation: The Role of CAPM*

  Valuation is incomplete without understanding a stock's performance in relation to its expected risk-adjusted return. Here, the Capital Asset Pricing Model is introduced to estimate what a stock *should* return, based on its beta (market risk), the risk-free rate, and expected market return. By comparing this to actual returns, the framework identifies whether a stock is outperforming or underperforming relative to its risk. When paired with the P/E ratio, this allows for a richer analysis: not only whether a stock is expensive or cheap, but whether its price is justified by its performance.

Chapter 2

Review of Literature

2.1 Introduction of Review of Literature

Stock valuation is a fundamental tool in making investment decisions, and plays an important role in identifying whether one of the stocks being traded is reasonably priced given its ability to generate earnings and be an attractive long-term growth potential. To put it in other words, we use a valuation multiple. Price-to-Earnings (P/E) Multiple is one of the oldest valuation methods. While it has been widely utilized, its potential is often not widely appreciated in academic and professional analysis, especially when used in isolation.

The chapter reviews the theoretical and empirical literature on the P/E multiple, focusing on its advantages and limitations, as well as its evolving role in investment evaluation. By combining findings from previous research this review brings to focus some of the critical gaps -- especially the lack of integration between valuation and performance measures (such as CAPM) -- and lays the groundwork for a more complete framework linking the dimensions of valuation and performance to produce better stock evaluation:

Review 1: The Capital Asset Pricing Model: A Critical Literature Review

- *Author: Matteo Rossi*
- *Publication Date: 2016*
- *URL:* https://www.researchgate.net/publication/307180424_The_capital_asset_pricing_model_a_critical_literature_review

This paper starts with a theoretical discussion of the foundational arguments supporting the Capital Asset Pricing Model (CAPM) to which it is presented, as well as some appraisals of the model's empirical validity over time. As regards its arguments about market efficiency and investors' rationality, Rossi makes several arguments concerning criticisms of the model's practical utility, not least of which is that it is simple and frequently used, but noted that it does not account well for real world asset returns. Rossi’s thorough attack on CAPM is not only welcome but in any case, questions the empirical validity of the model. There is a sense, Rossi suggests, that although CAPM is a fundamental construct, it cannot fully account for the complexity of market responses. In general, the results point out that CAPM, although a key model in historical periods, is practical limited and requires alternative or supplementary models for asset pricing for more accurate results.

Review 2: Is CAPM Still Valid in Today's Market Scenario?

- *Author: Rabha*
- *Publication Date: 2021*
- *URL: https://www.indianjournalofcapitalmarkets.com/index.php/IJF/article/view/169518*

This study examines the applicability of CAPM in the Indian capital market by analysing weekly closing prices of 48 NSE Nifty 50 companies over a decade. Using rolling regression methodology, the research compares the traditional CAPM with a constrained model proposed by Bajpai and Sharma (2015). The findings indicate that the constrained model outperforms the conventional CAPM in the Indian context, suggesting that modifications to the traditional model can enhance its predictive power in emerging markets. The research assesses CAPM's relevance in India's stock market, finding that a constrained version of the model provides better explanatory power for stock returns than the traditional CAPM. The constrained CAPM model offers improved performance in the Indian market, highlighting the need for model adaptations to suit specific market conditions.

Review 3: A Better Stock Pricing Model: A Systematic Literature Review

- *Authors: Nsama Musawa, Sumbye Kapena, Chanda Shikaputo*
- *Publication Date: 2020*
- *URL: https://jefjournal.org.za/index.php/jef/article/view/472*

This systematic review surveys the development of stock pricing models, considering the evolution of models concerning stock prices such as the Fama and French five-factor model (FF5M) and their comparative aspects with older models such as the three-factor model and the CAPM. The authors use the empirical studies of different markets to analyse the effects of the model on stock returns and, while they do not conclude that the FF5M is superior to other models, they raise a series of issues that should be addressed in further research. With reference to the present state of development and application of advanced stock pricing models, the paper points out the potential of FF5M but emphasizes the need for further research and refinement. While the FF5M shows promise, inconsistencies in empirical results suggest that no single model universally outperforms others, underscoring the importance of ongoing model development.

Review 4: The Six Decades of the Capital Asset Pricing Model: A Research Agenda

- *Authors: Various*
- *Publication Date: 2023*
- *URL: https://www.mdpi.com/1911-8074/16/8/356*

This comprehensive review traces the evolution of CAPM over sixty years, examining its various adaptations and applications across different asset classes. The authors utilize bibliometric analysis to identify research trends, influential publications, and emerging themes in CAPM literature. The study also outlines a future research agenda, emphasizing the need to address CAPM's limitations and explore its applicability in contemporary financial markets. The article offers an extensive overview of CAPM's development, highlighting its enduring relevance and the necessity for ongoing research to enhance its applicability. CAPM remains a foundational model in finance, but its limitations necessitate continued research and adaptation to address evolving market dynamics

Review 5: The Valuation Mystery: More Clues

- *Author: Financial Times*
- *Publication Date: 2024*
- *URL: https://www.ft.com/content/62c362f2-0c58-42ed-a8ff-19ace3a11821*

This article discusses the perplexing nature of stock valuations, particularly in the context of companies like GameStop experiencing significant stock price increases despite limited profitability. The piece explores factors contributing to high valuations, including low interest rates, capital-light business models, and demographic shifts. It raises questions about the sustainability of such valuations and the potential implications for market stability. The article examines the factors behind unusually high stock valuations, questioning their sustainability and the potential risks they pose to market equilibrium. Multiple factors contribute to elevated stock valuations, but their long-term sustainability remains uncertain, highlighting the need for cautious investment strategies.

Review 6: Predicting Stock Prices Using Future EPS Estimates and Historical PE Ratios: A Dual-Bound Approach

- *Author: Ayan Chaudhuri*
- *Publication Date: June 2024*
- *URL: https://www.researchgate.net/publication/381406895_Predicting_Stock_Prices_Using_Future_EPS_Estimates_and_Historical_PE_Ratios_A_Dual-_Bound_Approach*

This study introduces a valuation method that combines the 52-week average P/E ratio with future Earnings Per Share (EPS) estimates to predict stock price movements over a two-year horizon. By analyzing case studies of NSE: IEX and NSE: CMSINFO, the paper demonstrates the method's effectiveness in assessing stock valuations and predicting future price appreciation. The approach aims to provide a balanced forecast, mitigating market volatility extremes. The dual-bound approach offers a practical method for stock valuation by integrating historical P/E ratios with projected EPS, enhancing the accuracy of price predictions.

- Findings:

1. The method effectively identifies overvalued stocks at all-time highs.
2. It aids in determining optimal stock purchase timing and investment targets.
3. Limitations include dependence on accurate EPS estimates and sensitivity to market sentiment.

Review 7: Analysing and Forecasting P/E Ratios Using Investor Sentiment in Panel Data Regression and LSTM Models

- *Authors: [Not specified]*
- *Publication Date: March 2025*
- *URL: https://www.sciencedirect.com/science/article/pii/S1059056025000036*

This research investigates the influence of investor sentiment on P/E ratios using panel data regression and Long Short-Term Memory (LSTM) models. By extracting sentiment scores from textual data via Natural Language Processing (NLP), the study examines how these sentiments affect P/E ratios and evaluates the predictive capabilities of LSTM models. Investor sentiment significantly impacts P/E ratios, and incorporating sentiment analysis enhances the forecasting accuracy of these ratios.

- Findings:

1. Investor sentiment, derived from textual data, is a significant predictor of P/E ratios.
2. LSTM models outperform traditional regression models in forecasting P/E ratios.
3. The integration of sentiment analysis provides a more nuanced understanding of stock valuations.

Review 8: A Growth Adjusted Price-Earnings Ratio

- *Authors: Graham Baird, James Dodd, Lawrence Middleton*
- *Publication Date: January 2020*
- *URL: https://arxiv.org/abs/2001.08240*

The paper introduces the Growth Adjusted Price-Earnings (GA-P/E) ratio, which accounts for earnings growth in the traditional P/E ratio. By interpreting the P/E ratio as a payback period and adjusting for growth, the GA-P/E provides a more accurate measure of value and predictor of future stock returns. The GA-P/E ratio offers a refined valuation metric by incorporating earnings growth, enhancing its predictive power for stock returns.

Review 9: The Capital Asset Pricing Model: A Critical Literature Review

- **Author**: Matteo Rossi
- **Publication Date**: September 2016
- **URL**: https://www.inderscience.com/info/inarticle.php?artid=78682

This literature review critically examines the CAPM, discussing its foundational concepts, empirical validations, and criticisms. It highlights the model's assumptions and the debates surrounding its applicability in modern financial contexts. While CAPM remains a foundational model in finance, its assumptions and empirical shortcomings necessitate cautious application.

Findings:

1. CAPM's single-factor model oversimplifies the complexities of market returns.
2. Empirical tests show inconsistencies in CAPM's predictive accuracy.
3. Alternative models may offer better explanations for asset pricing.

Review 10: Implementing the Capital Asset Pricing Model in Forecasting Stock Returns: A Literature Review

- **Authors**: [Not specified]
- **Publication Date**: [Not specified]
- **URL**: https://journal.formosapublisher.org/index.php/ijba

This review explores the application of CAPM in forecasting stock returns, analyzing its effectiveness and limitations. It discusses the model's assumptions, its performance in various markets, and comparisons with alternative models. CAPM serves as a useful tool for forecasting stock returns, but its limitations suggest the need for complementary models. The study delves into the model's foundational assumptions and evaluates its effectiveness across various market environments. While CAPM is recognized for offering a baseline estimate of expected returns by quantifying systematic risk, the review highlights that its predictive accuracy is not uniform and can fluctuate depending on market conditions

2.2 Research Gap

In spite of the large amount of research on stock valuation approaches, especially price-to-earnings multiple (P/E) and Capital Asset Pricing Model (CAPM), a series of critical research gaps exist — notably when used in tandem with Indian equity markets like the NIFTY 50. This study was undertaken given the realization that standard valuation frameworks do not provide the adequate depth to judge a stock's fairness and its performance potential in a holistic manner.

Why risk-adjusted performance can make all the difference in one investment strategy: This is a major limitation of the existing literature. The NIFTY 50 index holds its own against overvaluation of companies and over indebtedness. However, from a dual lens perspective (i. e., from P/E multiples to CAPM expected return estimates), few studies consider whether overvalued companies can suffer sustained underperformance compared to CAPM expected return estimates. We therefore have a gap in understanding how valuation of stock and risk-adjusted performance can co-inform more robust investment strategies.

Moreover, the bulk of valuation literature is mostly globally focused or generalized. The scope of such research is not well developed in emerging markets like India. In India, and especially in respect of NIFTY 50 companies, there is obviously a lack of large scale, long term empirical studies on the performance of overvalued stocks over various periods of time. The literature on this topic tends to focus on short term mispricing, or event-driven corrections. Such literature neglects the structural factors that may tell if, in time, high P/E stocks produce consistently returns in line with expectations or fall back to market averages.

Another big difficulty was the lack of financial data. Yahoo Finance serves as a commonly used data source, but the lack of information on some NIFTY 50 stocks makes it difficult to implement fully consistent valuation models for the stock. This in turn raises several challenges regarding the reliability and comparability of results across the entire index, and in part raises another major issue in academic research: the importance of transparent, high-quality data in emerging markets.

Many stocks are priced based on future potential rather than on current fundamentals. In very few studies do they rigorously evaluate the accuracy of analyst projections or take account of adverse consequences of missed expectations. So there is an intriguing question that I have: How often do overvalued stocks meet/completion with their projected earnings, and what are the long-term implications of this?

Chapter 3

Research Methodology

3.1 Introduction to Research Methodology

The core objective of this research is to assess the effectiveness of the Price-to-Earnings (P/E) multiple, when integrated with expected returns derived from the Capital Asset Pricing Model (CAPM), in identifying overvalued and underperforming stocks within the NIFTY 50 index. Rather than relying on standalone valuation metrics or performance indicators, this study proposes a unified, quadrant-based classification framework that brings together valuation and performance insights. This dual-lens approach addresses the limitations of traditional methods and offers a more comprehensive, data-driven strategy for stock assessment and investment decision-making.

The methodology is primarily quantitative, relying on historical and current financial data extracted from reliable sources such as Yahoo Finance. Key variables include stock prices, earnings per share (EPS), beta values, the NIFTY 50 market return, and the risk-free rate—used to calculate CAPM-based expected returns. Due to data limitations, some NIFTY 50 stocks with missing data were excluded from portions of the analysis, and this is acknowledged as a constraint in data completeness.

P/E multiples are evaluated in relation to both sector-specific averages and the overall NIFTY 50 index, allowing the identification of relative overvaluation or undervaluation. Stocks are then cross-referenced with their actual recent performance to determine whether they have outperformed or underperformed against CAPM expectations. This forms the basis for the four-quadrant matrix that categorizes stocks as:

- Undervalued & Outperforming,
- Undervalued & Underperforming,
- Overvalued & Outperforming,
- Overvalued & Underperforming.

To support deeper performance analysis and forecasting, the study also employs ARIMA (Autoregressive Integrated Moving Average) models using extended historical stock price data. This adds a forward-looking dimension to the analysis, particularly for identifying future price trends of overvalued and underperforming stocks.

In addition to numerical evaluation, the study is informed by qualitative insights drawn from a structured review of academic literature and financial market reports. These contextual perspectives help interpret sectoral differences, macroeconomic influences, and behavioural factors that may affect the sustainability of high P/E ratios or CAPM-based return assumptions.

This chapter details the sampling design, data sources, application of the CAPM model, the quadrant classification process, and the statistical and forecasting tools applied. The structured methodology ensures both replicability and analytical rigor, enabling a thorough exploration of how valuation and performance metrics can be meaningfully combined to improve investment decisions in the Indian equity market—specifically within the NIFTY 50 universe.

3.2 Statement of the Problem

In equity valuation, the Price-to-Earnings (P/E) multiple is widely recognized for its simplicity and accessibility. However, its standalone use in identifying truly overvalued or underperforming stocks—particularly within a broad and sectorally diverse index like the NIFTY 50—has proven to be insufficient. Likewise, performance estimation models such as the Capital Asset Pricing Model (CAPM), while useful in calculating expected returns based on market risk, do not incorporate valuation mismatches or sectoral characteristics. As a result, relying on either P/E or CAPM in isolation often leads to conflicting or incomplete assessments of a stock's attractiveness.

For instance, a stock trading at a high P/E multiple might appear overvalued based on valuation norms, yet deliver strong returns that exceed risk-adjusted expectations. Conversely, a stock with a low P/E ratio might signal undervaluation, but continue to underperform due to weak earnings growth, industry pressures, or negative market sentiment. These inconsistencies highlight a key problem: conventional approaches fail to capture the dynamic relationship between valuation and actual performance, leading to frequent misclassifications and suboptimal investment decisions.

Existing literature and market practice have typically treated valuation metrics and performance models as distinct tools, with limited integration between the two. Moreover, sectoral and macroeconomic variables that influence stock behavior are often generalized or overlooked. This results in valuation models that may be academically sound but practically detached from the nuances of real-world market behavior—particularly in the Indian context, where certain sectors command persistently high or low P/E ratios due to structural reasons.

Another practical challenge is data availability. While data for this study is primarily sourced from Yahoo Finance, certain stocks within the NIFTY 50 lack complete historical or earnings data, limiting the scope of continuous analysis. Furthermore, while this study avoids using complex multi-factor models like the Fama-French Three-Factor Model, it incorporates time-series forecasting (via ARIMA) to extend the understanding of historical stock performance—particularly for those identified as overvalued and underperforming.

By unifying valuation and performance dimensions into a single model, this research offers a more structured, data-driven alternative to traditional heuristics. The framework not only enhances the reliability of stock selection but also provides investors and analysts with a holistic tool for navigating complex equity markets—especially in emerging economies like India, where valuation signals often diverge from actual performance.

3.3 Objective of the Study

1. To analyse the valuation levels of NIFTY 50 stocks using P/E multiples relative to index and sector benchmarks.
2. To calculate expected returns for NIFTY 50 stocks using the CAPM model, incorporating market risk, beta, and risk-free rate.
3. To construct a four-quadrant classification matrix that combines P/E valuation and CAPM performance to categorize stocks as:
   - Undervalued & Outperforming
   - Overvalued & Outperforming
   - Undervalued & Underperforming
   - Overvalued & Underperforming

3.4 Scope of the Study

This study aims to investigate the effectiveness of combining the Price-to-Earnings (P/E) ratio with CAPM-based expected returns for identifying overvalued and underperforming stocks in the NIFTY 50 index. By developing a composite quadrant-based classification model, the research explores a more integrated approach to stock analysis. The scope of the study is defined by several key dimensions, which are outlined below:

Focus on the NIFTY 50 Index:

The study is limited to stocks listed in the NIFTY 50 index, a broad representation of the Indian equity market. This index includes major large-cap stocks from diverse sectors, allowing for an extensive examination of how valuation and performance metrics interact within different market conditions.

Use of P/E Ratios and CAPM:

The primary variables in the study are the P/E multiple and the CAPM-based expected returns. The research uses these two models to assess stock valuation and performance, while acknowledging their respective limitations when used in isolation.

Stock Classification Framework:

The study introduces a quadrant-based classification model that categorizes stocks into four types based on both valuation (P/E ratio) and performance (CAPM expected returns). The categories are:

- Overvalued & Outperforming
- Overvalued & Underperforming
- Undervalued & Outperforming
- Undervalued & Underperforming

Forecasting Stock Performance via ARIMA:

For stocks classified as overvalued or underperforming, the study extends the analysis by forecasting their future performance using ARIMA (Auto Regressive Integrated Moving Average) models. This adds a predictive dimension to the analysis, helping to assess whether high P/E stocks will continue to outperform or face a correction.

Data Sources:

Data for this study is primarily sourced from Yahoo Finance, which provides historical stock prices, earnings per share (EPS), and other necessary financial variables. The study acknowledges the limitations of data completeness, as some stocks in the NIFTY 50 may have missing or incomplete historical data, which affects the scope of the analysis for those stocks.

Exclusion of Multi-Factor Models:

This study specifically excludes the use of multi-factor models like the Fama-French Three-Factor Model, focusing instead on a combination of the P/E multiple and CAPM. The exclusion is intentional, as the research aims to simplify the framework for practical use in investment decisions, without introducing additional complexity.

Examination of Sectoral and Macroeconomic Factors:

While the study recognizes sectoral differences in stock behavior and the impact of macroeconomic variables (such as inflation and GDP growth), these are not directly incorporated into the primary analytical model. However, sectoral variations and broader economic trends are discussed qualitatively to help contextualize findings and explore potential biases.

Practical Application for Investment Decision-Making:

The ultimate goal of the study is to provide a reliable and actionable framework for investors. By integrating valuation and performance measures into a unified model, the research aims to improve stock selection strategies and assist portfolio managers in making informed, data-backed decisions.

Limitations of the Study:

- Data limitations related to missing or incomplete financial data for some NIFTY 50 stocks.
- The research does not account for market anomalies or investor behavior, focusing primarily on objective financial metrics.
- The model may not capture all nuances, such as qualitative company performance or unexpected market disruptions.

3.5 Sampling Design

The sampling design for this study is structured to ensure a representative and methodologically sound selection of stocks within the NIFTY 50 index, while also addressing the research objectives of integrating the Price-to-Earnings (P/E) ratio with CAPM-based expected returns for stock analysis. The following details outline the sampling strategy:

1. *Target Population:*

The target population for this study consists of the 50 constituent stocks of the NIFTY 50 index. These stocks represent a broad cross-section of the Indian equity market, encompassing a wide range of sectors, such as technology, finance, consumer goods, and healthcare. The NIFTY 50 is an appropriate sample due to its status as a benchmark index for the Indian market, which makes it relevant for the study of stock valuation and performance.

2. *Inclusion Criteria:*

- **Constituent Stocks of NIFTY 50**: Only stocks listed on the NIFTY 50 index as of the study's reference period are included. This ensures that the sample is consistent and well-represented across a diverse set of industries.
- **Availability of Historical** Data: The stocks selected for analysis must have a sufficient amount of historical financial data available from Yahoo Finance, including stock prices, earnings per share (EPS), and other necessary financial metrics such as beta values. Due to limitations in data availability, stocks with missing or incomplete data are excluded from the study.
- **Time Period**: The analysis considers data spanning from a minimum of three to five years (depending on availability) to capture historical performance and allow for reliable forecasting using ARIMA models.

3. *Exclusion Criteria:*

- **Incomplete Data**: Any stocks that lack essential historical data—such as consistent stock prices, earnings data, or missing values for CAPM variables like market returns or risk-free rates—are excluded from the sample. This ensures the robustness and accuracy of the study.
- **Stocks with Extreme Market Behavior**: In cases where a stock experiences extreme volatility or unusual market behavior during the study period (e.g., due to extraordinary events like mergers or acquisitions), it may be excluded to avoid skewing the results.

3.6 Data Collection Methods

The data collection method for this study is designed to gather reliable and relevant financial information to assess the Price-to-Earnings (P/E) multiples and expected returns from the Capital Asset Pricing Model (CAPM) for NIFTY 50 stocks. This method involves collecting secondary data from credible and authoritative sources to ensure the accuracy and robustness of the analysis. The following outlines the specific steps and sources used in this study:

1. *Primary Data Sources:*

- **Yahoo Finance**: The primary source for historical financial data on the selected NIFTY 50 stocks is Yahoo Finance. This platform provides comprehensive datasets, including daily stock prices, earnings per share (EPS), market returns, and other financial metrics needed for the study. Yahoo Finance is chosen due to its accessibility, consistency, and breadth of available data for Indian stocks.
- **NSE India**: The National Stock Exchange (NSE) of India is used as a secondary data source for verifying stock prices, historical returns, and any missing data from Yahoo Finance. NSE India provides official and accurate data related to stock performance and financial indicators.
- **Financial Reports and Filings**: In addition to Yahoo Finance and NSE, financial reports and filings available on the respective company websites or through the NSE can be used to cross-verify earnings data and other relevant financial information when needed.

2. *Data Variables:*

The study collects data on the following key variables:

- **Stock Prices**: Daily closing prices of the stocks in the NIFTY 50 index are collected for a period spanning 3–5 years. This data is used to calculate returns and analyse price performance.
- **Earnings Per Share (EPS)**: Annual EPS values are retrieved from Yahoo Finance, which are essential for calculating the P/E ratios.
- **Beta Values**: The beta coefficient for each stock, which measures its volatility relative to the market, is sourced from Yahoo Finance or the NSE.
- **Market Returns**: Historical market returns, often measured through broad indices like the NIFTY 50 or BSE Sensex, are collected to compute the market's risk premium.

- **Risk-Free Rate**: The risk-free rate is typically derived from government bond yields, such as 10-year government securities. These rates are sourced from the Reserve Bank of India (RBI) or reliable financial platforms.

- **Expected Returns (CAPM)**: Using the CAPM formula, expected returns for each stock are calculated based on the risk-free rate, the stock's beta, and the market return.

3. *Data Time Period:*

- The study uses a historical data period of 3–5 years, ensuring a sufficient timeframe for capturing meaningful stock performance trends. This period allows for the calculation of CAPM-based expected returns and the analysis of the relationship between P/E ratios and actual stock performance.
- **Monthly Data Points**: Stock prices, EPS, market returns, and other relevant financial data are collected on a monthly basis. This frequency is ideal for capturing meaningful trends and accounting for monthly variations in stock performance.

4. *Forecasting Data (ARIMA):*

- For forecasting stock performance, the study applies the ARIMA (Auto Regressive Integrated Moving Average) model. This model requires historical stock price data, which is used to generate forecasts of future stock prices for overvalued and underperforming stocks identified through the P/E and CAPM analysis.
- The ARIMA model will be applied to stocks that have been classified as overvalued or underperforming, using historical stock price data to predict their future price trajectories for the next three months.

3.7 Statistical Tools Applied for Data Analysis

To achieve the research objectives and systematically classify NIFTY 50 stocks based on valuation and performance, the study employs a range of statistical tools and techniques. These tools are selected to ensure accurate computation, meaningful classification, and reliable forecasting. The tools applied are as follows:

Descriptive Statistics

- Used to summarize the basic features of the dataset, including measures such as mean, median, standard deviation, and range.
- Helps in understanding the distribution of P/E ratios, CAPM expected returns, and historical stock performance.

CAPM (Capital Asset Pricing Model)

- The CAPM formula is applied to calculate the expected return of each stock:

$$\boldsymbol{Expected\ Return = Rf + \beta(Rm - Rf)}$$

- where Rf is the risk-free rate, β is the stock's beta, and Rm is the market return

Valuation Classification using P/E Ratio Benchmarks

- P/E ratios are evaluated against sectoral and index-wide benchmarks to categorize stocks as Overvalued or Undervalued.
- Sector-based comparison helps adjust for industry-specific valuation norms.

Quadrant Mapping Model (Composite Framework)

- A quadrant-based classification model is created by combining P/E-based valuation (Overvalued vs. Undervalued) with CAPM-based performance (Outperforming vs. Underperforming).
- This results in four distinct quadrants:
  - Overvalued & Outperforming
  - Overvalued & Underperforming
  - Undervalued & Outperforming
  - Undervalued & Underperforming

ARIMA (Auto Regressive Integrated Moving Average) Forecasting

- Applied to the Overvalued & Underperforming stocks to forecast stock prices for the next three months.
- Helps assess whether these stocks are likely to continue underperforming or show signs of recovery.
- ACF and PACF plots are used for parameter selection, followed by model fitting and forecast visualization.

Data Visualization Tools

- Line charts, scatter plots, and quadrant diagrams are generated to visually interpret the classification and forecast results.
- These visuals aid in presenting insights clearly and effectively for academic and practical use.

Python & Excel-Based Analysis

Python (with libraries such as ***pandas***, ***statsmodels***, ***matplotlib***, and ***seaborn***) is used for quantitative analysis, CAPM calculations, and ARIMA modelling.

3.8 Limitation of the Study

While this study provides a structured and data-backed approach to classifying NIFTY 50 stocks based on valuation and performance, certain limitations must be acknowledged:

- *Data Availability and Incompleteness*
    - The primary data source, Yahoo Finance, had missing or inconsistent data for a few NIFTY 50 stocks, particularly in historical price and earnings records.
    - As a result, a few stocks were excluded from parts of the analysis, slightly limiting the comprehensiveness of the overall evaluation.
- *Simplified Assumptions of CAPM*
    - The Capital Asset Pricing Model assumes a linear relationship between risk and return, constant risk-free rates, and normally distributed returns—all of which may not hold true in real market conditions.

- Consequently, CAPM may not fully capture the complexities of stock performance, especially during periods of volatility or economic shocks.

- *P/E Ratio Interpretation Limitations*
  - The P/E multiple is influenced by both earnings' volatility and investor sentiment, which can distort its predictive value.
  - High P/E ratios do not always signal overvaluation, particularly in growth sectors, while low P/E may reflect risk or uncertainty rather than undervaluation.
- *Exclusion of Qualitative Factors*
  - Factors such as management quality, competitive positioning, brand value, regulatory changes, or investor sentiment are not included in the model, although they can significantly influence stock valuation and performance.
- *Short-Term Forecasting Window*
  - The ARIMA model was used to forecast stock prices for only a short horizon (3 months) and for a limited group of stocks (Overvalued & Underperforming).
  - Longer-term forecasting may require additional modelling and external economic indicators.
- *No Consideration of Dividend Yields or Payout Ratios*
  - The study does not incorporate dividend-related metrics, which are important for total return analysis and may affect the investment attractiveness of certain stocks.
- *Lack of Multi-Factor Models*
  - While CAPM is used for expected return estimation, more advanced models (such as Fama-French Three-Factor or Five-Factor models) were deliberately not included.
  - This may limit the depth of performance analysis, especially where size and value effects are relevant.

Chapter 4

Data Analysis and Interpretation

4.1 Introduction to Data Analysis and Interpretation

This chapter focuses on the analytical processes undertaken to evaluate and interpret the financial data collected for the NIFTY 50 stocks. The core objective is to classify stocks into four strategic quadrants—Overvalued & Outperforming, Overvalued & Underperforming, Undervalued & Outperforming, and Undervalued & Underperforming—using a composite framework based on the Price-to-Earnings (P/E) multiple and CAPM-based expected returns.

The analysis involves a step-by-step evaluation of each stock's relative valuation (via P/E) and its return performance (via CAPM and actual returns). Stocks are first assessed for overvaluation or undervaluation by benchmarking their P/E ratios against industry averages and NIFTY 50 mean values. In parallel, CAPM is used to compute expected returns using beta values, market return estimates, and the risk-free rate, which are then compared with the actual 3-month price changes.

The stocks that simultaneously appear overvalued (based on P/E) and underperforming (based on returns lower than CAPM expectations) are of special interest, as they potentially represent poor investment choices. These were subjected to further analysis through time-series forecasting using ARIMA to evaluate whether underperformance is temporary or part of a continuing trend.

Throughout this chapter, descriptive statistics, classification matrices, scatter plots, and ARIMA-based forecast graphs are used to aid interpretation. The aim is not only to statistically validate the quadrant classification but also to draw practical insights that inform investment decisions.

By combining valuation and return metrics in a single framework, this analysis offers a more holistic understanding of stock attractiveness and lays the groundwork for data-driven portfolio strategies.

4.2 Interpretation of Data

The interpretation of data in this study is centred around analysing how each NIFTY 50 stock performs when evaluated simultaneously on two axes: valuation (using the P/E multiple) and return performance (using CAPM-based expected returns vs actual returns). The resulting classification offers insights into the strategic positioning of each stock within the broader market context.

1. *Valuation-Based Interpretation (P/E Analysis)*

Stocks with significantly higher P/E ratios than their respective industry or index averages were categorized as **Overvalued**, while those with lower P/E ratios were considered **Undervalued**. Interpretation of this data showed that:

- Certain sectors (e.g., IT and FMCG) consistently exhibited higher average P/E ratios, reflecting growth expectations.
- A few companies had elevated P/E ratios without a corresponding growth in earnings, raising concerns of speculative overvaluation.

2. *Performance-Based Interpretation (CAPM vs Actual Returns)*

Expected returns were calculated using the CAPM formula. These were then compared with actual 3-month price changes for each stock:

- Stocks outperforming their CAPM-expected returns were labelled **Outperforming**.
- Stocks returning less than the CAPM benchmark were considered **Underperforming**.

This comparison helped in identifying mismatches between investor expectations and actual stock behavior, suggesting the presence of market inefficiencies or sentiment-driven pricing.

3. *Quadrant Classification Outcomes*

Based on the combined valuation and performance assessment, stocks were categorized into the four quadrants:

- **Overvalued & Outperforming**: Indicates investor confidence or strong short-term momentum.
- **Overvalued & Underperforming**: Highlights potential correction risks; possibly overhyped or declining companies.

- **Undervalued & Outperforming**: Represents optimal investment opportunities—mispriced stocks gaining momentum.
- **Undervalued & Underperforming**: May indicate value traps or fundamentally weak businesses.

4. *ARIMA Forecast Insights*

For stocks in the "Overvalued & Underperforming" quadrant, ARIMA-based forecasting was conducted to examine whether recent underperformance is likely to continue. Results suggested that:

- Some stocks show signs of continued weakness, validating their quadrant classification.
- Others may rebound, signalling short-term underperformance due to external shocks rather than fundamental issues.

5. *Cross-Sector Interpretation*

When analysing patterns across industries:

- Tech and Pharma stocks tended to cluster in the **Overvalued** quadrants.
- Banking and Infrastructure stocks often appeared in the **Undervalued** zones, some with underperformance due to macroeconomic stress.

4.3 Computational Framework and Python Implementation

4.3.1 Importing Essential Libraries and APIs:

To facilitate accurate valuation and performance analysis, the study requires robust historical and real-time financial data. This necessitates a combination of web scraping and structured data extraction through APIs. The following libraries are imported to enable these functions:

- **yfinance**: Allows programmatic access to historical stock price data and financial indicators via Yahoo Finance.
- **datetime**: Supports the definition of specific date ranges for backtesting and time-series operations.
- **pandas as pd**: Provides data structures like DataFrames, enabling easy handling, manipulation, and analysis of data in tabular form.

- **BeautifulSoup**: Used for parsing HTML documents during web scraping tasks, particularly when automated API data is unavailable.
- **time**: Manages delays in scraping loops to respect web server request limits.
- requests: Facilitates HTTP requests for retrieving web-based financial information.

These libraries lay the foundation for collecting structured and unstructured financial data from both Yahoo Finance and third-party sources.

Code:

- import yfinance as yf
- from datetime import datetime
- import pandas as pd
- from bs4 import BeautifulSoup
- import time
- import requests

4.3.2 Importing Additional Libraries for Stock Data Retrieval:

In this part of the study, additional libraries are imported to streamline the process of accessing stock data and performing necessary date manipulations. These libraries are essential for ensuring smooth data retrieval and handling time-sensitive operations during analysis.

- **pandas_datareader**: This library is used for reading stock data directly from Yahoo Finance (among other sources), enabling seamless integration with Pandas for data analysis.
- **datetime**: The datetime library is once again utilized to work with timestamps and time-based operations, essential for organizing the stock data by specific periods.
- **_future_.division**: This import ensures that the division operation in Python 2.x behaves as in Python 3.x, where division always returns a float (rather than performing integer division).

These tools are necessary for pulling stock prices and handling time-series data effectively, setting up the data structure for further analysis.

Code:

- from pandas_datareader import DataReader

- from datetime import datetime
- from __future__import division

4.3.3 Fetching NIFTY 50 Stock Symbols and P/E Ratios:

This section of the code focuses on fetching and processing real-time stock data from Yahoo Finance, specifically targeting the NIFTY 50 index. It involves fetching the list of NIFTY 50 stock symbols, retrieving their corresponding P/E ratios, and handling common issues such as missing data and API request failures.

- Step 1: Fetch NIFTY 50 Stock Symbols with Retry Mechanism
  The first function get_nifty50_symbols () dynamically retrieves the list of NIFTY 50 stock symbols from a publicly available CSV file hosted on the NSE website. The function incorporates a retry mechanism, allowing up to five attempts to fetch the data in case of connection errors. Each stock symbol is appended with. NS to make it compatible with Yahoo Finance.
- Step 2: Fetch P/E Ratios for NIFTY 50 Stocks
  The get_pe_ratios() function retrieves the trailing P/E ratios for each stock in the NIFTY 50 list using the yfinance API. It handles missing data and API rate limits gracefully by including a delay between requests. If a stock's P/E ratio is unavailable, the function prints a message and moves on to the next stock.
- Step 3: Run the Process
  This step executes the functions to fetch the stock symbols and their P/E ratios, printing error messages if any step fails.
- Step 4: Save Data to CSV
  Finally, the P/E ratios are saved to a CSV file (**nifty50_pe_ratios.csv**) for later use in analysis, ensuring that data is stored in a structured format.

Code:

- def get_nifty50_symbols(max_retries=5):
- retries = 0
- while retries < max_retries:
- try:
  - url = "https://archives.nseindia.com/content/indices/ind_nifty50list.csv"
  - df = pd.read_csv(url)

- stock_symbols = df["Symbol"].tolist()
  - return [symbol + ".NS" for symbol in stock_symbols] # Append .NS for Yahoo Finance
- except requests.exceptions.RequestException as e:
  - print(f"Error fetching NIFTY 50 list: {e}")
  - retries += 1
  - time.sleep(5) # Retry after 5 seconds
- return []
- def get_pe_ratios(stock_list):
- pe_data = []
- for stock in stock_list:
- try:
  - ticker = yf.Ticker(stock)
  - pe_trailing = ticker.info.get("trailingPE", None)
  - if pe_trailing is not None:
  - pe_data.append({"Symbol": stock, "Trailing P/E": pe_trailing})
  - else:
  - print(f"Skipping {stock} due to missing P/E ratio.")
- except Exception as e:
  - print(f"Error fetching data for {stock}: {e}")
  - time.sleep(5) # Delay to avoid hitting rate limits
  - continue
- return pe_data
- nifty50_symbols = get_nifty50_symbols()
- if not nifty50_symbols:
- print("Could not fetch NIFTY 50 list after multiple retries. Exiting.")
- else:
- pe_ratios = get_pe_ratios(nifty50_symbols)
- if pe_ratios:
- df_pe_ratios = pd.DataFrame(pe_ratios)
- df_pe_ratios.to_csv("nifty50_pe_ratios.csv", index=False)
- print("P/E ratios saved to 'nifty50_pe_ratios.csv'")
- else:
- print("No valid P/E ratios found.")

Output:

Symbol	Trailing P/E
ADANIENT.NS	78.0509
ADANIPORTS.NS	25.98035
APOLLOHOSP.NS	76.94212
ASIANPAINT.NS	55.530136
AXISBANK.NS	13.165891
BAJAJ-AUTO.NS	30.055061
BAJFINANCE.NS	35.211433
BAJAJFINSV.NS	38.785885
BEL.NS	46.42753
BHARTIARTL.NS	44.29925
CIPLA.NS	24.967606
COALINDIA.NS	6.988147
DRREDDY.NS	18.269379
EICHERMOT.NS	34.457413
ETERNAL.NS	308.14667

4.3.4 Fetching NIFTY 50 Stock Symbols and Industry-wise P/E Analysis:

This code provides a detailed workflow for fetching, analyzing, and summarizing the P/E ratios and industry-wise performance of NIFTY 50 stocks using Yahoo Finance data.

1. *Fetching NIFTY 50 Stock Symbols:*

   The get_nifty50_symbols function dynamically fetches the list of NIFTY 50 stock symbols from a CSV file hosted on the NSE India website. A retry mechanism is implemented to handle potential request errors, ensuring robustness in case of network issues.

2. *Fetching Industry and P/E Data:*

   The get_industry_pe_data function iterates through each stock symbol, retrieves its data from Yahoo Finance using the yfinance API, and extracts the industry and trailing P/E ratio for each stock. This data is collected in a dictionary, with industry names as keys and P/E ratios as values.

3. *Computing Industry-wise P/E Statistics:*

   The compute_industry_pe_statistics function takes the collected P/E data and calculates statistics for each industry, including:

   - Average P/E ratio
   - Minimum and maximum P/E ratios
   - Standard deviation of P/E ratios
   - Count of stocks in each industry

   The function returns a Pandas DataFrame for easy data manipulation and presentation.

4. *Running the Process and Saving Results:*

   The script fetches the NIFTY 50 symbols and retrieves the P/E ratios along with industry information. After computation, the results are saved in a CSV file, **nifty50_industry_pe_analysis.csv**, for further analysis.

5. *Displaying Results:*

   The script also prints the top 10 industries sorted by average P/E ratio, providing a quick overview of the industries with the highest P/E values in the NIFTY 50 index.

Code:

- def get_nifty50_symbols(max_retries=5):
- retries = 0
- while retries < max_retries:
- try:
  - url = "https://archives.nseindia.com/content/indices/ind_nifty50list.csv"
  - df = pd.read_csv(url)
  - stock_symbols = df["Symbol"].tolist()
  - return [symbol + ".NS" for symbol in stock_symbols] # Append .NS for Yahoo Finance
- except requests.exceptions.RequestException as e:
  - print(f"Error fetching NIFTY 50 list: {e}")

- retries += 1
- time.sleep(5) # Retry after 5 seconds
- return []
- def get_industry_pe_data(stock_list):
- industry_data = {}
- for stock in stock_list:
- try:
    - ticker = yf.Ticker(stock)
    - info = ticker.info
    - industry = info.get("industry", "Unknown")
    - pe_trailing = info.get("trailingPE", None)
    - if pe_trailing is not None:
    - if industry not in industry_data:
        - industry_data[industry] = []
    - industry_data[industry].append(pe_trailing)
    - time.sleep(1) # Avoid hitting API rate limits
- except Exception as e:
    - print(f"Error fetching data for {stock}: {e}")
- return industry_data
- def compute_industry_pe_statistics(industry_data):
- industry_list = []
- for industry, pe_values in industry_data.items():
- industry_list.append({
    - "Industry": industry,
    - "Avg P/E": round(sum(pe_values) / len(pe_values), 2),
    - "Min P/E": round(min(pe_values), 2),
    - "Max P/E": round(max(pe_values), 2),
    - "Std Dev P/E": round(pd.Series(pe_values).std(), 2),
    - "Stocks Count": len(pe_values)
- })
- return pd.DataFrame(industry_list)
- nifty50_symbols = get_nifty50_symbols()
- if not nifty50_symbols:
- print("Could not fetch NIFTY 50 list. Exiting.")

- else:
- industry_pe_data = get_industry_pe_data(nifty50_symbols)
- industry_df = compute_industry_pe_statistics(industry_pe_data)
- output_file = "nifty50_industry_pe_analysis.csv"
- industry_df.to_csv(output_file, index=False)
- print(f"Industry-wise P/E analysis saved as {output_file}")
- print(industry_df.sort_values(by="Avg P/E", ascending=False).head(10)) # Show top 10 highest P/E industries

Output:

Industry	Avg P/E	Min P/E	Max P/E	Std Dev P/E	Stocks Count
Thermal Coal	42.52	6.99	78.05	50.25	2
Marine Shipping	25.98	25.98	25.98		1
Medical Care Facilities	76.94	76.94	76.94		1
Specialty Chemicals	55.53	55.53	55.53		1
Banks - Regional	15.29	9	20.67	5.54	6
Auto Manufacturers	23.8	7.69	34.46	9.48	6
Credit Services	23.74	12.26	35.21	16.23	2
Financial Conglomerates	38.79	38.79	38.79		1
Aerospace & Defense	46.43	46.43	46.43		1
Telecom Services	44.3	44.3	44.3		1
Drug Manufacturers - Specialty & Generic	27.03	18.27	37.85	9.95	3
Internet Retail	308.15	308.15	308.15		1
Building Materials	51	47.26	54.74	5.29	2
Information Technology Services	24.77	19.27	31.19	4.35	5
Insurance - Life	78.46	71.7	85.23	9.57	2

4.3.5 Segregating Stocks by Industry and Saving the Data:

This code snippet focuses on organizing and saving NIFTY 50 stocks based on their respective industries along with their trailing P/E ratios. Here's a detailed breakdown of the steps:

1. *Loading Data:*
    - The first part of the code loads the necessary CSV files:
        - **nifty50_pe_ratios.csv** which contains the stock symbols and their respective trailing P/E ratios.

- **nifty50_industry_pe_analysis.csv** which contains industry data for the NIFTY 50 stocks. From this file, the allowed industries are extracted into a list.

2. *Fetching Industry Data:*

   - For each stock in the df DataFrame (which contains the P/E ratios), the code fetches the stock's industry information from Yahoo Finance using the yfinance API.
   - Only industries that are present in the allowed_industries list (from the industry CSV file) are considered.
   - The stock's P/E ratio is retrieved from the df DataFrame and stored in the industry_data dictionary, grouped by industry names.

3. *Compiling Data into a List:*

   - After processing all stocks, the industry data is compiled into a list of dictionaries, where each dictionary contains the industry name, stock symbol, and the trailing P/E ratio for that stock.

4. *Saving the Data:*

   - The compiled list is converted into a Pandas DataFrame and saved to a new CSV file, nifty50_stocks_by_industry.csv. This file contains the NIFTY 50 stocks categorized by their respective industries along with their P/E ratios.

5. *Output Confirmation:*

   - The script prints a confirmation message once the data has been saved successfully, ensuring that the process has completed as expected.

This approach efficiently organizes stock data by industry, enabling better analysis of P/E ratios across sectors within the NIFTY 50 index.

Code:

- file_path = "nifty50_pe_ratios.csv"
- df = pd.read_csv(file_path)

- industry_file_path = "nifty50_industry_pe_analysis.csv"
- industry_df = pd.read_csv(industry_file_path)
- allowed_industries = industry_df.iloc[:, 0].tolist()
- industry_data = {}
- for stock in df["Symbol"]:
- try:
- ticker = yf.Ticker(stock)
- industry = ticker.info.get("industry", "Unknown")
- if industry in allowed_industries:
    - pe_value = df.loc[df["Symbol"] == stock, "Trailing P/E"].values[0]
    - if industry not in industry_data:
    - industry_data[industry] = []
    - industry_data[industry].append({"Symbol": stock, "Trailing P/E": pe_value})
- time.sleep(1) # Avoid API rate limits
- except Exception as e:
- print(f"Error fetching industry for {stock}: {e}")
- industry_list = []
- for industry, stocks in industry_data.items():
- for stock in stocks:
- industry_list.append({
    - "Industry": industry,
    - "Symbol": stock["Symbol"],
    - "Trailing P/E": stock["Trailing P/E"]
- })
- output_file = "nifty50_stocks_by_industry.csv"
- industry_df = pd.DataFrame(industry_list)
- industry_df.to_csv(output_file, index=False)
- print(f" Stocks segregated by industry and saved as {output_file}")

Output:

Industry	Symbol	Trailing P/E
Thermal Coal	ADANIENT.NS	78.0509
Thermal Coal	COALINDIA.NS	6.988147
Marine Shipping	ADANIPORTS.NS	25.98035
Medical Care Facilities	APOLLOHOSP.NS	76.94212
Specialty Chemicals	ASIANPAINT.NS	55.530136
Banks - Regional	AXISBANK.NS	13.165891
Banks - Regional	HDFCBANK.NS	20.666956
Banks - Regional	ICICIBANK.NS	20.095573
Banks - Regional	INDUSINDBK.NS	8.997528
Banks - Regional	KOTAKBANK.NS	19.696293

4.3.6 Classifying NIFTY 50 Stocks Based on P/E Ratios Thumb Rule:

This code snippet focuses on classifying NIFTY 50 stocks based on their Price-to-Earnings (P/E) ratios into different valuation categories. Here's a detailed breakdown of the steps involved:

1. *Loading Data:*
   - The script begins by loading the **nifty50_pe_ratios.csv** file into a Pandas DataFrame, which contains the stock symbols along with their respective trailing P/E ratios.
2. *Handling Missing or Invalid P/E Values:*
   - To ensure that P/E ratios are valid for classification, the Trailing P/E column is converted to numeric values. If there are any non-numeric or missing values, they are replaced with NaN using the errors="coerce" argument.
   - The script then drops rows where the Trailing P/E value is NaN, ensuring that only valid P/E ratios are processed.
3. *Classifying Stocks:*
   - The function classify_pe(pe) is defined to classify stocks based on their P/E ratio:
     - **Undervalued**: Stocks with a P/E ratio below 10.

- **Fairly Valued**: Stocks with a P/E ratio between 10 and 40 (inclusive).
- **Overvalued**: Stocks with a P/E ratio greater than 40.

- This function is applied to each stock in the dataset using apply (), and a new column, Valuation, is created to store the classification results.

4. *Saving Classified Data:*
   - After classification, the updated DataFrame is saved to a new CSV file, nifty50_pe_valuation_thumbrule.csv, which contains the stock symbols, P/E ratios, and their respective valuations (Undervalued, Fairly Valued, or Overvalued).
5. *Displaying Results:*
   - The script prints a confirmation message indicating that the classification process has been completed and the data has been saved.
   - Additionally, it displays the first 5 rows of the classified data to give a preview of the results.

This process effectively categorizes NIFTY 50 stocks into valuation segments based on their P/E ratios, providing a clear and structured overview of the market's current valuation trends.

Code:

```
o file_path = "nifty50_pe_ratios.csv"
o df = pd.read_csv(file_path)
o df["Trailing P/E"] = pd.to_numeric(df["Trailing P/E"], errors="coerce")
o df = df.dropna(subset=["Trailing P/E"])
o def classify_pe(pe):
    o if pe < 10:
        o return "Undervalued"
    o elif 10 <= pe <= 40:  # Combining both normal & high-growth ranges into
      "Fairly Valued"
        o return "Fairly Valued"
    o else:  # P/E > 50
        o return "Overvalued"
o df["Valuation"] = df["Trailing P/E"].apply(classify_pe)
```

- output_file = "nifty50_pe_valuation_thumbrule.csv"
- df.to_csv(output_file, index=False)
- print(f"✅ NIFTY 50 P/E valuation completed and saved as {output_file}")
- print("\nÇ First 5 Rows of Classified Data:")
- print(df.head(5)) # Display first 5 rows

Output:

Symbol	Trailing P/E	Valuation
ADANIENT.NS	78.0509	Overvalued
ADANIPORTS.NS	25.98035	Fairly Valued
APOLLOHOSP.NS	76.94212	Overvalued
ASIANPAINT.NS	55.530136	Overvalued
AXISBANK.NS	13.165891	Fairly Valued
BAJAJ-AUTO.NS	30.055061	Fairly Valued
BAJFINANCE.NS	35.211433	Fairly Valued
BAJAJFINSV.NS	38.785885	Fairly Valued
BEL.NS	46.42753	Overvalued
BHARTIARTL.NS	44.29925	Overvalued
CIPLA.NS	24.967606	Fairly Valued

4.3.7 Classifying NIFTY 50 Stocks Based on Industry P/E Ratios Range:

This code snippet performs a detailed valuation range analysis for NIFTY 50 stocks based on their P/E ratios and compares them with industry averages and statistics. The steps involved are as follows:

1. *Loading Data:*
    - The script begins by loading the **nifty50_stocks_by_industry.csv** file, which contains stock symbols, their P/E ratios, and associated industries.
    - It also loads the **nifty50_industry_pe_analysis.csv** file, which contains the average P/E, standard deviation, minimum, and maximum P/E values for each industry.
2. *Building Industry Data Dictionary:*
    - The script processes the industry analysis data and stores each industry's details (such as average P/E, standard deviation, etc.) in a dictionary for easy lookup.

- It handles potential missing values in the standard deviation by assigning a default value (10% of the average P/E or at least 1), ensuring the calculations are robust.

3. *Processing Stock Valuation:*
    - For each stock, the script retrieves the corresponding industry information and compares the stock's P/E ratio against the industry's average, standard deviation, and other statistics.
    - The script then defines a valuation range for each stock using a lower and upper bound based on the industry's average P/E and standard deviation. The stock is classified as:
        - **Undervalued**: If the stock's P/E is below the lower bound.
        - **Overvalued**: If the stock's P/E is above the upper bound.
        - **Fairly Valued**: If the stock's P/E is within the defined range.
4. *Storing the Analysis:*
    - For each stock, the results are stored in a dictionary, capturing the stock's symbol, industry, P/E ratio, industry statistics, and the valuation classification.
5. *Saving Results to a CSV:*
    - The analysis is converted into a DataFrame and saved as **nifty50_stocks_valuation_range_analysis.csv**, allowing for further review or use in reports.
6. *Displaying Results:*
    - The script prints a confirmation message indicating that the analysis has been completed and saved.
    - It also displays the first 5 rows of the valuation analysis to give a preview of the results.

This process enables a deeper understanding of how each stock compares with its industry peers, providing a more nuanced approach to classifying stocks as undervalued, fairly valued, or overvalued.

Code:

- file_path = "nifty50_stocks_by_industry.csv"
- df = pd.read_csv(file_path)
- industry_file_path = "nifty50_industry_pe_analysis.csv"
- industry_df = pd.read_csv(industry_file_path)
- industry_data = {}
- for _, row in industry_df.iterrows():
- avg_pe = row['Avg P/E']
- std_dev_pe = row['Std Dev P/E']
- if pd.isna(std_dev_pe) or std_dev_pe == 0:
- std_dev_pe = max(0.1 * avg_pe, 1) # Use 10% of Avg P/E or at least 1
- industry_data[row['Industry']] = {
- 'Avg P/E': avg_pe,
- 'Min P/E': row['Min P/E'],
- 'Max P/E': row['Max P/E'],
- 'Std Dev P/E': std_dev_pe,
- 'Stocks Count': row['Stocks Count']
- }
- industry_analysis = []
- for _, stock_row in df.iterrows():
- symbol = stock_row['Symbol']
- stock_pe = stock_row['Trailing P/E']
- industry = stock_row['Industry'] # Directly get the industry from the stock row
- if industry in industry_data:
- avg_pe = industry_data[industry]['Avg P/E']
- min_pe = industry_data[industry]['Min P/E']
- max_pe = industry_data[industry]['Max P/E']
- std_dev_pe = industry_data[industry]['Std Dev P/E']
- lower_bound = max(avg_pe - std_dev_pe, 0) # Ensure lower bound is not negative
- upper_bound = min(avg_pe + std_dev_pe, avg_pe * 2) # Cap the upper bound
- if stock_pe < lower_bound:
    - valuation = "Undervalued"
- elif stock_pe > upper_bound:

    - valuation = "Overvalued"
- else:
    - valuation = "Fairly Valued"
- industry_analysis.append({
    - "Industry": industry,
    - "Symbol": symbol,
    - "Trailing P/E": stock_pe,
    - "Avg P/E (Industry)": avg_pe,
    - "Min P/E (Industry)": min_pe,
    - "Max P/E (Industry)": max_pe,
    - "Std Dev P/E (Industry)": std_dev_pe,
    - "Lower Bound": lower_bound,
    - "Upper Bound": upper_bound,
    - "Valuation": valuation
- })
- analysis_df = pd.DataFrame(industry_analysis)
- output_file = "nifty50_stocks_valuation_range_analysis.csv"
- analysis_df.to_csv(output_file, index=False)
- print(f"Stocks valuation analysis complete and saved as {output_file}")
- print("\nFirst 5 Rows of Valuation Analysis:")
- print(analysis_df.head(5)) # Display first 5 rows

Output:

Industry	Symbol	Trailing P/E	Avg P/E (Industry)	Min P/E (Industry)	Max P/E (Industry)	Std Dev P/E (Industry)	Lower Bound	Upper Bound	Valuation
Thermal Coal	ADANIENT.NS	78.0509	42.52	6.99	78.05	50.25	0	85.04	Fairly Valued
Thermal Coal	COALINDIA.NS	6.988147	42.52	6.99	78.05	50.25	0	85.04	Fairly Valued
Marine Shipping	ADANIPORTS.NS	25.98035	25.98	25.98	25.98	2.598	23.382	28.578	Fairly Valued
Medical Care Facilities	APOLLOHOSP.NS	76.94212	76.94	76.94	76.94	7.694	69.246	84.634	Fairly Valued
Specialty Chemicals	ASIANPAINT.NS	55.530136	55.53	55.53	55.53	5.553	49.977	61.083	Fairly Valued
Banks - Regional	AXISBANK.NS	13.165891	15.29	9	20.67	5.54	9.75	20.83	Fairly Valued
Banks - Regional	HDFCBANK.NS	20.666956	15.29	9	20.67	5.54	9.75	20.83	Fairly Valued
Banks - Regional	ICICIBANK.NS	20.095573	15.29	9	20.67	5.54	9.75	20.83	Fairly Valued
Banks - Regional	INDUSINDBK.NS	8.997528	15.29	9	20.67	5.54	9.75	20.83	Undervalued

Banks - Regional	KOTAKBANK.NS	19.696293	15.29	9	20.67	5.54	9.75	20.83	Fairly Valued
Banks - Regional	SBIN.NS	9.136962	15.29	9	20.67	5.54	9.75	20.83	Undervalued
Auto Manufacturers	BAJAJ-AUTO.NS	30.055061	23.8	7.69	34.46	9.48	14.32	33.28	Fairly Valued
Auto Manufacturers	EICHERMOT.NS	34.457413	23.8	7.69	34.46	9.48	14.32	33.28	Overvalued
Auto Manufacturers	HEROMOTOCO.NS	18.58335	23.8	7.69	34.46	9.48	14.32	33.28	Fairly Valued

4.3.8 Performance Analysis of NIFTY 50 Stocks Based on Valuation Methods:

This code snippet performs a performance analysis of NIFTY 50 stocks by comparing two valuation methods: the **Thumb Rule** method and the **Range-Based** method. The analysis is based on the stock price change over the last three months (**3M price change**). The steps involved in this analysis are as follows:

1. *Loading Data:*
    - The script begins by loading two CSV files:
        - **nifty50_pe_valuation_thumbrule.csv**: Contains stock symbols along with their valuation based on the Thumb Rule method.
        - **nifty50_stocks_valuation_range_analysis.csv**: Contains stock symbols with their valuation based on the Range-Based method.
2. *Renaming Columns:*
    - To enhance clarity, the columns representing the valuation methods in both datasets are renamed. The Thumb Rule valuation column is renamed to Valuation_Thumbrule, and the Range-Based valuation column is renamed to Valuation_Range.
3. *Merging Datasets:*
    - The two datasets are merged based on the stock symbol (Symbol) using an inner join. This results in a single DataFrame that contains both valuation columns for each stock.
4. *Formatting Symbols for Yahoo Finance:*
    - A new column, Yahoo_Symbol, is created, which formats the stock symbols to be compatible with Yahoo Finance (appending .NS for NIFTY 50 stocks).
5. *Fetching Stock Price Changes:*
    - The get_stock_price_change function is defined to fetch the 3-month price change for each stock using Yahoo Finance. It calculates the percentage change

in closing prices between the start and end of the specified period (default is 3 months).

- The 3M Price Change (%) is added as a new column in the merged DataFrame.

6. *Data Cleaning:*
   - The script removes rows with missing price change data, ensuring that only stocks with valid price information are analyzed.
7. *Calculating Average Returns:*
   - The average price change over the last three months is calculated for each valuation category (Thumb Rule and Range-Based). This is done using the groupby function to group the stocks by their valuation category, followed by calculating the mean price change for each group.
8. *Comparing the Two Valuation Methods:*
   - The script compares the average performance (price change) of the stocks based on each valuation method:
     - *Thumb Rule Method Performance*
     - *Range-Based Method Performance*
   - It prints the performance results for both methods, providing insights into which method better aligns with stock performance.
9. *Saving Results:*
   - The merged dataset, which includes the stock symbols, valuation methods, and performance data, is saved as **nifty50_valuation_performance.csv**.
10. *Displaying Results:*
    - The script prints the average performance for each valuation category and displays a message confirming that the detailed performance report has been saved successfully.

This analysis helps to assess which valuation method (Thumb Rule or Range-Based) correlates better with stock performance in terms of price change, providing valuable insights for decision-making.

Code:

- thumbrule_file = "nifty50_pe_valuation_thumbrule.csv"
- range_analysis_file = "nifty50_stocks_valuation_range_analysis.csv"
- df_thumbrule = pd.read_csv(thumbrule_file)
- df_range = pd.read_csv(range_analysis_file)
- df_thumbrule = df_thumbrule.rename(columns={"Valuation": "Valuation_Thumbrule"})
- df_range = df_range.rename(columns={"Valuation": "Valuation_Range"})
- merged_df = pd.merge(df_thumbrule, df_range, on="Symbol", how="inner")
- def format_symbol(symbol):
- if not symbol.endswith(".NS"):
- return symbol + ".NS"
- return symbol
- merged_df["Yahoo_Symbol"] = merged_df["Symbol"].apply(format_symbol)
- def get_stock_price_change(symbol, period="3mo"):
- try:
- stock = yf.Ticker(symbol)
- history = stock.history(period=period)
- if len(history) < 2:
    - return None # Not enough data
- start_price = history["Close"].iloc[0] # First available price
- end_price = history["Close"].iloc[-1]   # Latest price
- price_change = ((end_price - start_price) / start_price) * 100 # % Change
- return round(price_change, 2)
- except Exception:
- return None # Handle errors silently
- merged_df["3M Price Change (%)"] = merged_df["Yahoo_Symbol"].apply(lambda x: get_stock_price_change(x, period="3mo"))
- merged_df = merged_df.dropna(subset=["3M Price Change (%)"])
- thumbrule_performance = merged_df.groupby("Valuation_Thumbrule")["3M Price Change (%)"].mean()
- range_performance = merged_df.groupby("Valuation_Range")["3M Price Change (%)"].mean()

- # Step 2: Compare which method aligns better
- print("\n **Performance Based on Price Changes**:")
- print("\n◆ **Thumb Rule Method Performance:**")
- print(thumbrule_performance)
- print("\n◆ **Range-Based Method Performance:**")
- print(range_performance)
- output_file = "nifty50_valuation_performance.csv"
- merged_df.to_csv(output_file, index=False)
- print(f"\n **Detailed performance report saved as {output_file}**")

Output:

```
📊 **Performance Based on Price Changes**:

🔹 **Thumb Rule Method Performance:**
Valuation_Thumbrule
Fairly Valued     3.021154
Overvalued        8.736111
Undervalued      -0.975000
Name: 3M Price Change (%), dtype: float64

🔹 **Range-Based Method Performance:**
Valuation_Range
Fairly Valued     6.223023
Overvalued        0.420000
Undervalued      -9.725000
Name: 3M Price Change (%), dtype: float64

📊 **Detailed performance report saved as nifty50_valuation_performance.csv**
```

4.3.9 Final Stock Valuation Analysis for NIFTY 50:

This code merges the valuation data from both the Thumb Rule and Range-Based methods, then determines the **Final Valuation** based on a consensus between the two methods. The analysis also includes the 3-month price change for each stock. Here's a breakdown of the process:

1. *Loading Data:*
   - The script loads the following CSV files:

- **nifty50_pe_valuation_thumbrule.csv**: Contains stock symbols with their valuation based on the Thumb Rule method.
- **nifty50_stocks_valuation_range_analysis.csv**: Contains stock symbols with their valuation based on the Range-Based method.
- **nifty50_valuation_performance.csv**: Contains stock symbols with their 3-month price change percentage.

2. *Merging Data:*
   - The data is merged into a single DataFrame, where:
     - The Thumb Rule and Range-Based valuations are merged based on the Symbol.
     - The 3-month price change is added to the merged data.
3. *Determining Final Valuation:*
   - A function (determine_final_valuation) is used to create a final consensus valuation:
     - If both the Thumb Rule and Range-Based methods agree on the stock's valuation (i.e., both classify the stock as "Undervalued", "Fairly Valued", or "Overvalued"), that value is taken as the **Final Valuation**.
     - If the two methods disagree, the stock is flagged as **Overvalued** for further analysis.
4. *Selecting Columns for Final Output:*
   - The final DataFrame contains the following columns:
     - **Symbol**: The stock symbol.
     - **Valuation_Thumbrule**: The valuation based on the Thumb Rule method.
     - **Valuation_Range**: The valuation based on the Range-Based method.
     - **Final Valuation**: The final consensus valuation.

- **3M Price Change (%)**: The percentage change in the stock's price over the last three months.

5. *Saving and Displaying the Final Output:*
   - The final output is saved as nifty50_final_stock_valuation.csv.
   - A sample of the first 10 rows is displayed to show the result.

Code:

- # Load the two valuation CSVs
- thumb_rule_file = "nifty50_pe_valuation_thumbrule.csv"
- range_based_file = "nifty50_stocks_valuation_range_analysis.csv"
- performance_file = "nifty50_valuation_performance.csv"
- thumb_df = pd.read_csv(thumb_rule_file)
- range_df = pd.read_csv(range_based_file)
- performance_df = pd.read_csv(performance_file)
- merged_df = thumb_df.merge(range_df, on="Symbol", suffixes=("_Thumbrule", "_Range"))
- merged_df = merged_df.merge(performance_df[['Symbol', '3M Price Change (%)']], on="Symbol", how="left")
- def determine_final_valuation(row):
- if row["Valuation_Thumbrule"] == row["Valuation_Range"]:
- return row["Valuation_Thumbrule"] # If both methods agree, take that value
- else:
- return "Overvalued" # If there's disagreement, flag it for further analysis
- merged_df["Final Valuation"] = merged_df.apply(determine_final_valuation, axis=1)
- final_output = merged_df[["Symbol", "Valuation_Thumbrule", "Valuation_Range", "Final Valuation", "3M Price Change (%)"]]
- output_file = "nifty50_final_stock_valuation.csv"
- final_output.to_csv(output_file, index=False)
- print(f"Final stock valuation analysis saved as {output_file}")
- print("\nSample of the final output:")
- print(final_output.head(10)) # Show first 10 rows

Output:

Symbol	Valuation_Thumbrule	Valuation_Range	Final Valuation	3M Price Change (%)
ADANIENT.NS	Overvalued	Fairly Valued	Overvalued	0.52
ADANIPORTS.NS	Fairly Valued	Fairly Valued	Fairly Valued	10.89
APOLLOHOSP.NS	Overvalued	Fairly Valued	Overvalued	3.21
ASIANPAINT.NS	Overvalued	Fairly Valued	Overvalued	10.25
AXISBANK.NS	Fairly Valued	Fairly Valued	Fairly Valued	20.73
BAJAJ-AUTO.NS	Fairly Valued	Fairly Valued	Fairly Valued	-6.35
BAJFINANCE.NS	Fairly Valued	Fairly Valued	Fairly Valued	17.18
BAJAJFINSV.NS	Fairly Valued	Fairly Valued	Fairly Valued	15.44
BEL.NS	Overvalued	Fairly Valued	Overvalued	19.33
BHARTIARTL.NS	Overvalued	Fairly Valued	Overvalued	13.96

4.3.10 CAPM Analysis for Overvalued Stocks:

This code performs a CAPM analysis for stocks that are classified as "Overvalued" in the NIFTY 50 index. It calculates the expected return for each stock based on the CAPM model, which is used to determine a stock's expected return based on its beta (systematic risk) and the overall market's expected return.

1. *Loading Data:*

The script loads the following CSV files:

- **nifty50_final_stock_valuation.csv**: Contains the final valuation of NIFTY 50 stocks, including the classification of stocks (e.g., "Overvalued").
- **nifty50_historical_return.csv**: Contains the historical annual returns for the NIFTY 50 index, which is used to calculate the Market Return (Rm). This is the only csv file which has been taken manually by downloading it from the Website: Primeinvestor.
    - **URL**: https://primeinvestor.in/nifty-50-returns/
- **nifty50_stocks_by_industry.csv**: Contains the stock symbols with their respective industry information.

2. *Market Return Calculation (Rm):*

The Market Return (Rm) is calculated as the average of the "Annual" column in the market returns data. It is then converted from percentage to decimal form.

3. *Risk-Free Rate (Rf):*

The Risk-Free Rate (Rf) is set to the 10-year Government bond yield for India, which is 6.58% (or 0.0658 in decimal form).

4. *Filtering Overvalued Stocks:*

Stocks that are classified as "Overvalued" in the Final Valuation column are filtered from the valuation dataset.

5. *Beta Fetching:*

For each overvalued stock, the script tries to fetch the **beta** value using the Yahoo Finance API. If a stock's beta is unavailable, the script uses the industry average beta instead (calculated by averaging the beta values of all stocks in the same industry).

6. *CAPM Calculation:*

The script calculates the expected return (Re) for each stock using the CAPM formula:

$$\boldsymbol{Re = Rf + \beta \cdot (Rm - Rf)}$$

Where:

- Rf is the Risk-Free Rate.
- β is the stock's beta (a measure of its sensitivity to market movements).
- Rm is the Market Return.

7. *Storing and Saving the Results:*

The results for each stock, including:

- **Symbol**: Stock symbol.
- **Industry**: Industry the stock belongs to.
- **Beta**: The stock's beta.
- **Expected Return (Re)**: The calculated expected return based on the CAPM model.

These results are saved into a CSV file named **overvalued_stocks_capm.csv**.

Code:

- valuation_file = "nifty50_final_stock_valuation.csv"
- valuation_df = pd.read_csv(valuation_file)
- market_file = "nifty50_historical_return.csv"
- market_df = pd.read_csv(market_file)
- industry_file = "nifty50_stocks_by_industry.csv"
- industry_df = pd.read_csv(industry_file)
- Rm = market_df['Annual'].mean() / 100 # Convert percentage to decimal
- print(f"Calculated Market Return (Rm): {Rm:.4f} ({Rm*100:.2f}%)")
- Rf = 0.0658 # 6.58%
- overvalued_stocks = valuation_df[valuation_df['Final Valuation'] == "Overvalued"]
- overvalued_stocks = overvalued_stocks.merge(industry_df, on="Symbol", how="left")
- industry_betas = {}
- for symbol in overvalued_stocks['Symbol']:
- try:
- ticker = yf.Ticker(symbol)
- beta = ticker.info.get('beta', None)
- if beta is not None:
        - industry = overvalued_stocks.loc[overvalued_stocks['Symbol'] == symbol, 'Industry'].values[0]
    - if industry not in industry_betas:
    - industry_betas[industry] = []
    - industry_betas[industry].append(beta)
- except Exception as e:
- print(f"Error fetching beta for {symbol}: {e}")
- for industry, betas in industry_betas.items():
- industry_betas[industry] = sum(betas) / len(betas) if betas else 1.0 # Default to 1.0 if missing
- capm_results = []
- for index, row in overvalued_stocks.iterrows():
- symbol = row['Symbol']
- industry = row['Industry']

- try:
- ticker = yf.Ticker(symbol)
- beta = ticker.info.get('beta', None)
- if beta is None:
  - if industry in industry_betas:
  - beta = industry_betas[industry] # Use industry average beta
    - print(f"¦ Warning: Missing beta for {symbol}. Using industry average beta ({industry}) = {beta:.2f}")
  - else:
  - beta = 1.0 # Default market beta if no industry beta is available
    - print(f"¦ Warning: Missing beta for {symbol} and industry {industry}. Using default beta = 1.0")
- Re = Rf + beta * (Rm - Rf)
- # Append data to results list
- capm_results.append({
  - 'Symbol': symbol,
  - 'Industry': industry,
  - 'Beta': beta,
  - 'Expected Return (Re)': Re * 100 # Convert to percentage
- })
- except Exception as e:
- print(f"Error fetching data for {symbol}: {e}")
- capm_df = pd.DataFrame(capm_results)
- print("\nCAPM Analysis for Overvalued Stocks:")
- print(capm_df.head(10)) # Display first 10 rows
- capm_df.to_csv("overvalued_stocks_capm.csv", index=False)
- print("\nCAPM analysis completed and saved as 'overvalued_stocks_capm.csv'")

Output:

Symbol	Industry	Beta	Expected Return (Re)
ADANIENT.NS	Thermal Coal	0.664	12.3931872
APOLLOHOSP.NS	Medical Care Facilities	0.372	9.8367856
ASIANPAINT.NS	Specialty Chemicals	0.389	9.9856172
BEL.NS	Aerospace & Defense	0.376	9.8718048
BHARTIARTL.NS	Telecom Services	0.194	8.2784312
COALINDIA.NS	Thermal Coal	0.015	6.711322
EICHERMOT.NS	Auto Manufacturers	0.286	9.0838728
ETERNAL.NS	Internet Retail	0.252	8.7862096
GRASIM.NS	Building Materials	0.272	8.9613056
HDFCLIFE.NS	Insurance - Life	0.741	13.0673068
HINDALCO.NS	Aluminum	1.168	16.8056064
HINDUNILVR.NS	Household & Personal Products	0.335	9.512858
JSWSTEEL.NS	Steel	0.804	13.6188592
JIOFIN.NS	Asset Management	1	15.3348

4.3.11 CAPM vs Price Change Performance Analysis:

This code compares the expected returns from the CAPM model with the actual 3-month price change for stocks that are classified as "Overvalued" in the NIFTY 50 index. The comparison helps assess how well the CAPM model's expected return aligns with the stock's real-world performance.

1. *Loading Data:*
   - The script loads the following CSV files:
     - **overvalued_stocks_capm.csv**: Contains the CAPM results for the overvalued stocks, including their expected returns (Re).
     - **nifty50_final_stock_valuation.csv**: Contains information about the NIFTY 50 stocks, including their 3-month price changes.
2. *Merging Data:*
   - The CAPM results are merged with the valuation data (specifically, the 3-month price change) based on the stock symbol (Symbol).

3. *Handling 3M Price Change Data:*

   - The '3M Price Change (%)' column is converted to numeric values, with any errors (e.g., missing data) being handled by setting the invalid entries as NaN (Not a Number).

4. *Comparing Expected Return with 3M Price Change:*

   - A function (compare_returns) is defined to compare the CAPM-based expected return (Expected Return (Re)) with the actual 3-month price change (3M Price Change (%)).
   - If the 3-month price change data is missing, the result is marked as "Data Missing."
   - If the difference between the expected return and the actual return is within ±1%, the stock is classified as "Matched CAPM."
   - Otherwise, the stock is categorized as either "Underperformed" or "Outperformed" based on whether the expected return is greater or less than the actual return.

5. *Applying the Comparison:*

   - The function is applied to each row in the merged dataframe, creating a new column (CAPM Performance) to store the results.

6. *Saving the Results:*

   - The results, including the stock symbol, expected return, actual price change, and performance comparison, are saved to a new CSV file (capm_vs_price_change.csv).

7. *Output:*

   - The script displays the first 10 rows of the analysis, showing a summary of the stock's expected return, actual return, and performance comparison.

8. *Result File:*

   - The analysis is saved in a file named capm_vs_price_change.csv for further review.

Code:

```
o	capm_file = "overvalued_stocks_capm.csv"
o	capm_df = pd.read_csv(capm_file)
o	valuation_file = "nifty50_final_stock_valuation.csv"
o	valuation_df = pd.read_csv(valuation_file)
o	merged_df = pd.merge(capm_df, valuation_df[['Symbol', '3M Price Change (%)']], on="Symbol", how="left")
o	merged_df['3M Price Change (%)'] = pd.to_numeric(merged_df['3M Price Change (%)'], errors='coerce')
o	def compare_returns(row):
o	expected_return = row['Expected Return (Re)']
o	actual_return = row['3M Price Change (%)']
o	if pd.isna(actual_return):  # If 3M Price Change data is missing
o	return "Data Missing"
o	if abs(expected_return - actual_return) <= 1:  # Tolerance of ±1%
o	return "Matched CAPM"
o	elif expected_return > actual_return:
o	return "Underperformed"
o	else:
o	return "Outperformed"
o	merged_df['CAPM Performance'] = merged_df.apply(compare_returns, axis=1)
o	output_file = "capm_vs_price_change.csv"
o	merged_df.to_csv(output_file,  index=False)
o	print("\nCAPM Performance Analysis:")
o	print(merged_df[['Symbol', 'Expected Return (Re)', '3M Price Change (%)', 'CAPM Performance']].head(10))
o	print(f"\nAnalysis saved as '{output_file}'")
```

Output:

Symbol	Industry	Beta	Expected Return (Re)	3M Price Change (%)	CAPM Performance
ADANIENT	Thermal C	0.664	12.3931872	0.52	Underperformed
APOLLOHO	Medical Ca	0.372	9.8367856	3.21	Underperformed
ASIANPAIN	Specialty C	0.389	9.9856172	10.25	Matched CAPM
BEL.NS	Aerospace	0.376	9.8718048	19.33	Outperformed
BHARTIAR	Telecom S	0.194	8.2784312	13.96	Outperformed
COALINDIA	Thermal C	0.015	6.711322	4.06	Underperformed
EICHERMO	Auto Manu	0.286	9.0838728	7.64	Underperformed
ETERNAL.N	Internet Re	0.252	8.7862096	10.25	Outperformed
GRASIM.N	Building M	0.272	8.9613056	11.9	Outperformed
HDFCLIFE.	Insurance	0.741	13.0673068	13.8	Matched CAPM
HINDALCO	Aluminum	1.168	16.8056064	6.58	Underperformed
HINDUNIL	Household	0.335	9.512858	-2.61	Underperformed

4.3.12 Overvalued Stocks - Final Performance Data:

This code performs the final data cleaning and merges the "Overvalued" stocks with their CAPM performance results, saving a refined dataset for further analysis.

1. *Loading Data:*
    - The script first loads the nifty50_final_stock_valuation.csv file, which contains the final valuation of the NIFTY 50 stocks.
    - The file capm_vs_price_change.csv is also loaded, which contains the CAPM performance results, including the comparison of expected returns with the 3-month price change.
2. *Filtering Overvalued Stocks:*
    - The script filters the rows where the "Final Valuation" column is labeled "Overvalued," isolating only those stocks that are considered overvalued based on the valuation criteria.
3. *Merging the Data:*
    - The data from the filtered overvalued stocks is merged with the CAPM performance data based on the stock symbol (Symbol), ensuring that only the overvalued stocks with corresponding CAPM performance data are included.

4. *Selecting Relevant Columns:*

   - After merging, the script retains only the relevant columns: Symbol, Final Valuation, and CAPM Performance.

5. *Saving the Cleaned Data:*

   - The cleaned and merged data is saved to a new CSV file (overvalued_stocks_final_performance.csv) for future use.

6. *Result:*

   - A sample of the first 10 rows of the cleaned dataset is printed to the console, showing the overvalued stocks along with their final valuation and CAPM performance status.
   - The cleaned comparison dataset is saved in the file overvalued_stocks_final_performance.csv.

Code:

- valuation_file = "nifty50_final_stock_valuation.csv"
- valuation_df = pd.read_csv(valuation_file)
- overvalued_stocks = valuation_df[valuation_df["Final Valuation"] == "Overvalued"]
- capm_performance_file = "capm_vs_price_change.csv"
- capm_df = pd.read_csv(capm_performance_file)
- comparison_df = overvalued_stocks.merge(capm_df[['Symbol', 'CAPM Performance']], on="Symbol", how="inner")
- final_df = comparison_df[['Symbol', 'Final Valuation', 'CAPM Performance']]
- output_file = "overvalued_stocks_final_performance.csv"
- final_df.to_csv(output_file, index=False)
- print("\n■ Overvalued Stocks - Final Performance Data:")
- print(final_df.head(10))
- print(f"\n✅ Cleaned comparison saved as '{output_file}'")

Output:

Symbol	Final Valuation	CAPM Performance
ADANIENT.NS	Overvalued	Underperformed
APOLLOHOSP.NS	Overvalued	Underperformed
ASIANPAINT.NS	Overvalued	Matched CAPM
BEL.NS	Overvalued	Outperformed
BHARTIARTL.NS	Overvalued	Outperformed
COALINDIA.NS	Overvalued	Underperformed
EICHERMOT.NS	Overvalued	Underperformed
ETERNAL.NS	Overvalued	Outperformed
GRASIM.NS	Overvalued	Outperformed
HDFCLIFE.NS	Overvalued	Matched CAPM
HINDALCO.NS	Overvalued	Underperformed
HINDUNILVR.NS	Overvalued	Underperformed
JSWSTEEL.NS	Overvalued	Underperformed

4.3.13 Filtering Underperforming Overvalued Stocks:

This code identifies and saves NIFTY 50 stocks that are both overvalued and have underperformed based on CAPM analysis.

1. *Loading Data:*

The script begins by loading the file overvalued_stocks_final_performance.csv, which contains stock symbols along with their final valuation status and CAPM-based performance classification.

2. *Filtering Stocks:*

It filters the dataset to select only those stocks where:

- Final Valuation is "Overvalued"
- CAPM Performance is "Underperformed"
  These are the stocks that are priced higher than justified by fundamentals and have delivered returns below their expected CAPM return.

3. *Extracting Stock Symbols:*

From the filtered dataset, it extracts a list of stock symbols for use or display.

4. *Saving the Results:*

The filtered DataFrame (with Symbol, Final Valuation, and CAPM Performance) is saved as underperforming_overvalued_stocks.csv.

<u>*Code:*</u>

- file_path = "overvalued_stocks_final_performance.csv"
- df = pd.read_csv(file_path)
- underperforming_overvalued = df[
- (df["Final Valuation"] == "Overvalued") &
- (df["CAPM Performance"] == "Underperformed")
-]
- stock_symbols = underperforming_overvalued["Symbol"].tolist()
- output_file = "underperforming_overvalued_stocks.csv"
- underperforming_overvalued.to_csv(output_file, index=False)
- if stock_symbols:
- print("\n■ The stocks that are both Overvalued and also Underperformed are:")
- print(", ".join(stock_symbols))
- print(f"\n Stock names saved to '{output_file}'")
- else:
- print("\nNo Overvalued stocks Underperformed.")

<u>*Output:*</u>

Symbol	Final Valuation	CAPM Performance
ADANIENT.NS	Overvalued	Underperformed
APOLLOHOSP.NS	Overvalued	Underperformed
COALINDIA.NS	Overvalued	Underperformed
EICHERMOT.NS	Overvalued	Underperformed
HINDALCO.NS	Overvalued	Underperformed
HINDUNILVR.NS	Overvalued	Underperformed
JSWSTEEL.NS	Overvalued	Underperformed

4.3.14 ARIMA Forecasting for Overvalued & Underperformed Stocks:

This code applies ARIMA time series forecasting on NIFTY 50 stocks that are both overvalued and underperformed to predict their prices for the next three months.

1. *Loading Data:*

The code begins by reading the overvalued_stocks_final_performance.csv file, which contains the valuation and CAPM-based performance status of NIFTY 50 stocks. It filters stocks that are marked as "Overvalued" and "Underperformed".

2. *Defining Forecast Period:*

The forecasting horizon is defined as 90 days (~3 months).

3. *Looping Through Selected Stocks:*

For each filtered stock symbol:

Historical closing prices for the last 2 years are fetched using Yahoo Finance via the yfinance API.

The datetime index is ensured to be continuous with daily frequency using forward fill for missing dates to maintain ARIMA requirements.

4. *ARIMA Model Fitting:*

An ARIMA(1,1,1) model is fitted to the adjusted time series. This is a simple model assuming a first difference with one lag and one moving average term.

5. *Forecasting and Plotting:*

- The model forecasts stock prices for the next 90 days.
- The historical price curve is plotted in blue.
- The forecasted prices are plotted in dashed red.
- The last predicted price is annotated on the graph for easy interpretation.
- Each graph is titled with the respective stock symbol.

6. *Result:*

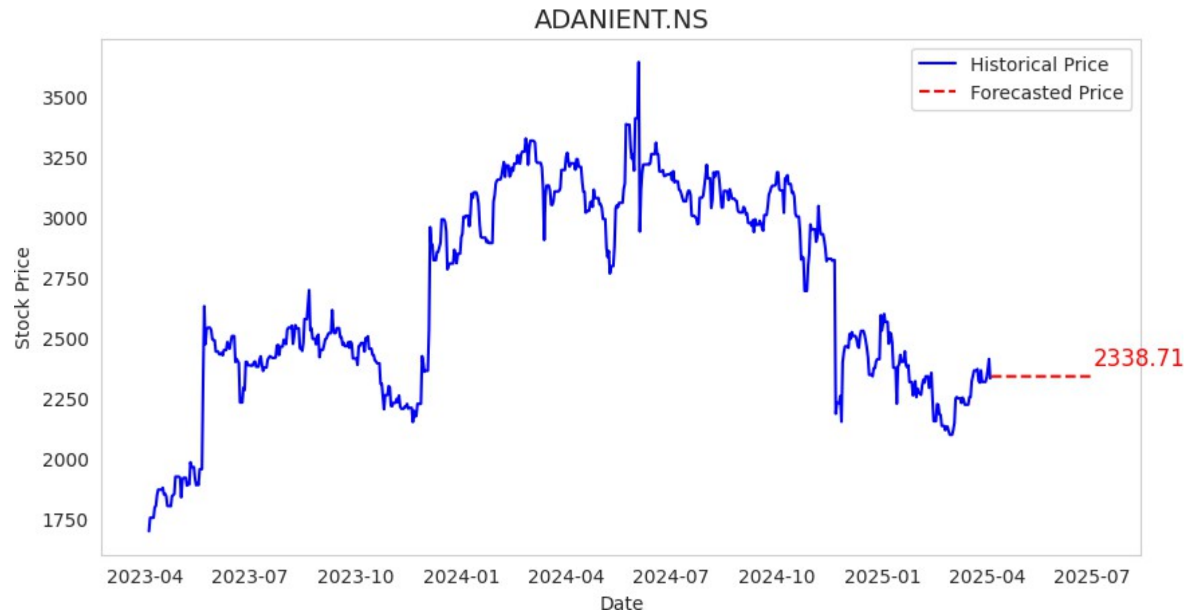
A plot is shown for each stock displaying its historical trend and 3-month ARIMA forecast. If an error occurs during any stock's processing, it is caught and printed for debugging without stopping the entire loop.

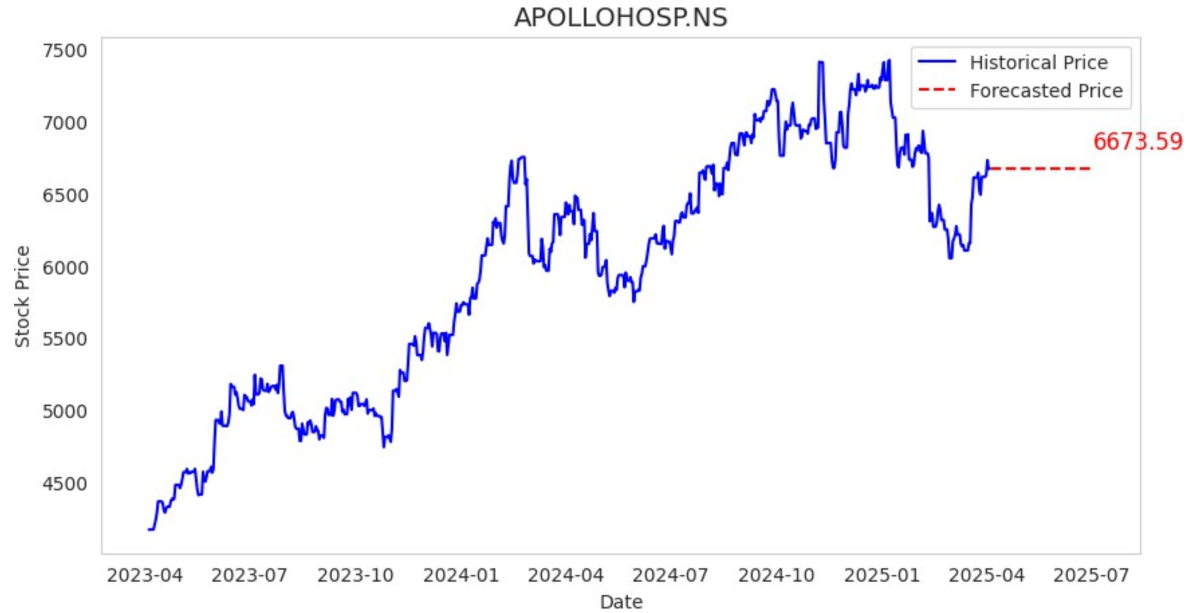
Code:

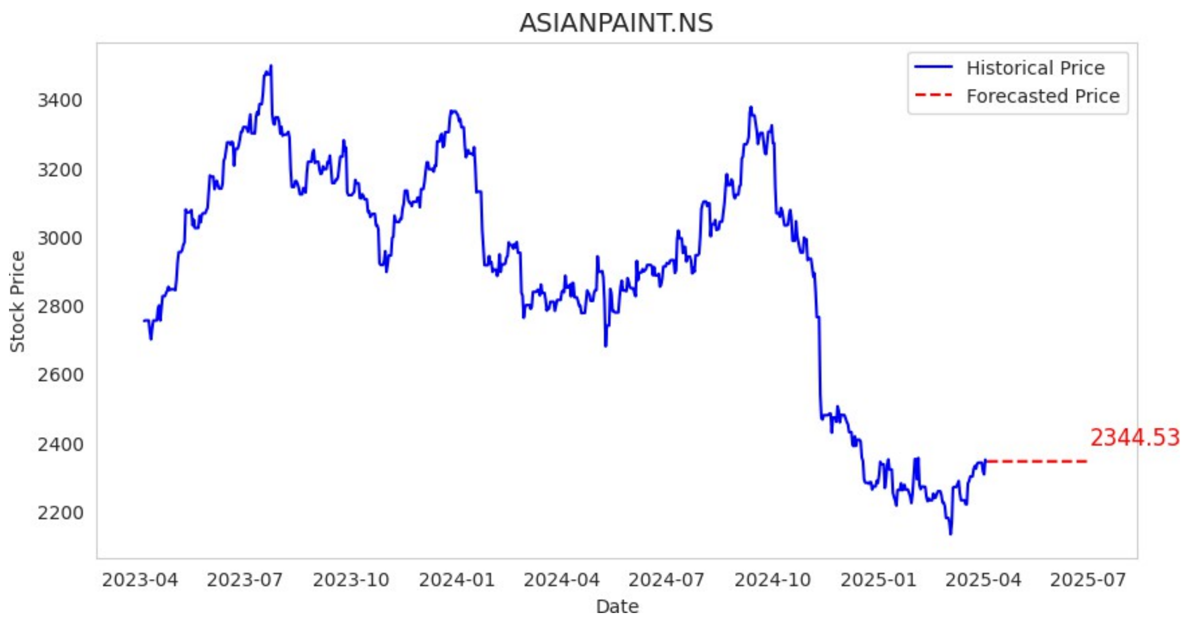
- import pandas as pd
- import yfinance as yf
- import matplotlib.pyplot as plt
- import numpy as np
- from statsmodels.tsa.arima.model import ARIMA
- file_path = "overvalued_stocks_final_performance.csv"
- df = pd.read_csv(file_path)
- selected_stocks = df[(df["Final Valuation"] == "Overvalued") & (df["CAPM Performance"] == "Underperformed")]["Symbol"]
- forecast_period = 90 # ~3 months
- for symbol in selected_stocks:
- try:
- stock_data = yf.download(symbol, period="2y", interval="1d")["Close"].dropna()
- stock_data.index = pd.to_datetime(stock_data.index)
- stock_data = stock_data.asfreq('D').ffill() # Fill missing dates if any
- model = ARIMA(stock_data, order=(1,1,1)) # (p,d,q) = (1,1,1) as a simple start
- model_fit = model.fit()
    - forecast_index = pd.date_range(stock_data.index[-1], periods=forecast_period+1, freq='D')[1:]
- forecast = model_fit.forecast(steps=forecast_period)
- plt.figure(figsize=(10,5))
- plt.plot(stock_data.index, stock_data, label="Historical Price", color="blue")
        - plt.plot(forecast_index, forecast, label="Forecasted Price", color="red", linestyle="dashed")
- last_forecasted_price = forecast.iloc[-1]
- last_forecasted_date = forecast_index[-1]
- plt.annotate(f"{last_forecasted_price:.2f}",
        - xy=(last_forecasted_date, last_forecasted_price),
        - xytext=(last_forecasted_date, last_forecasted_price + (last_forecasted_price * 0.02)),
        - fontsize=12, color="red",
        - arrowprops=dict(facecolor='red', arrowstyle='->'))

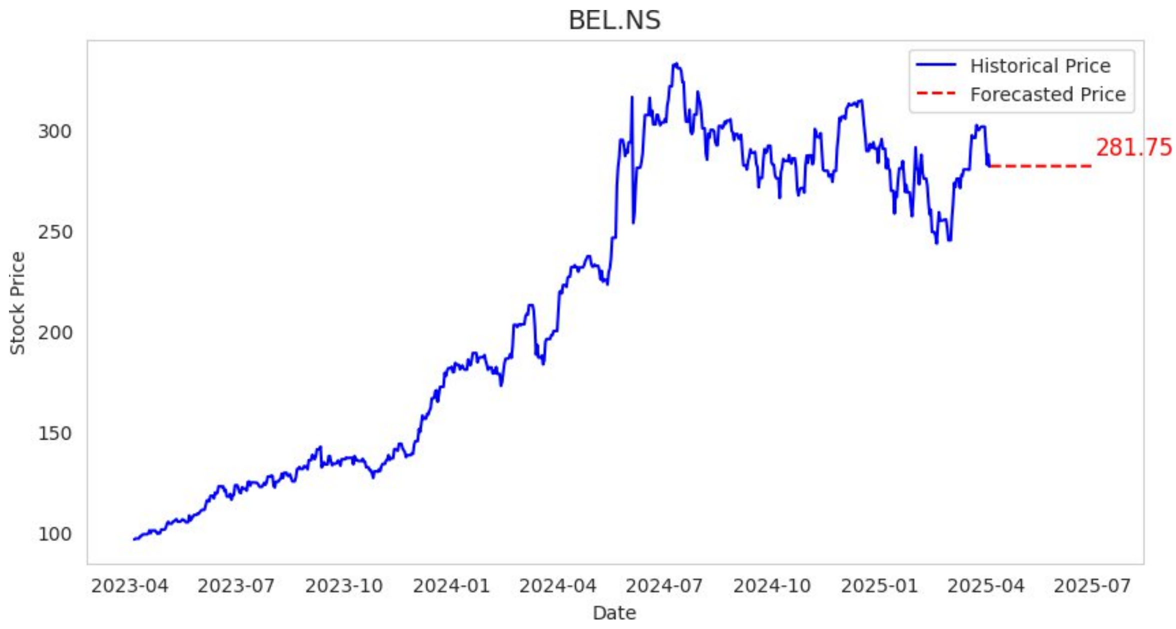
- plt.title(f"{symbol}", fontsize=14)
- plt.xlabel("Date")
- plt.ylabel("Stock Price")
- plt.legend()
- plt.grid()
- plt.show()
- except Exception as e:
- print(f"Error processing {symbol}: {e}")

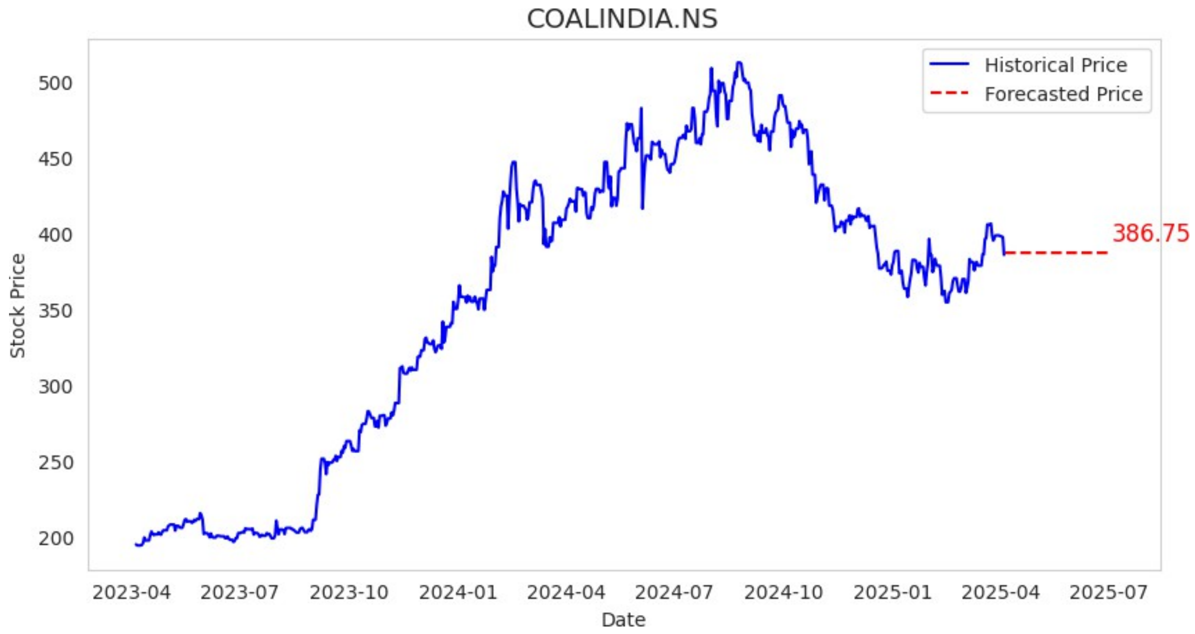
Output:

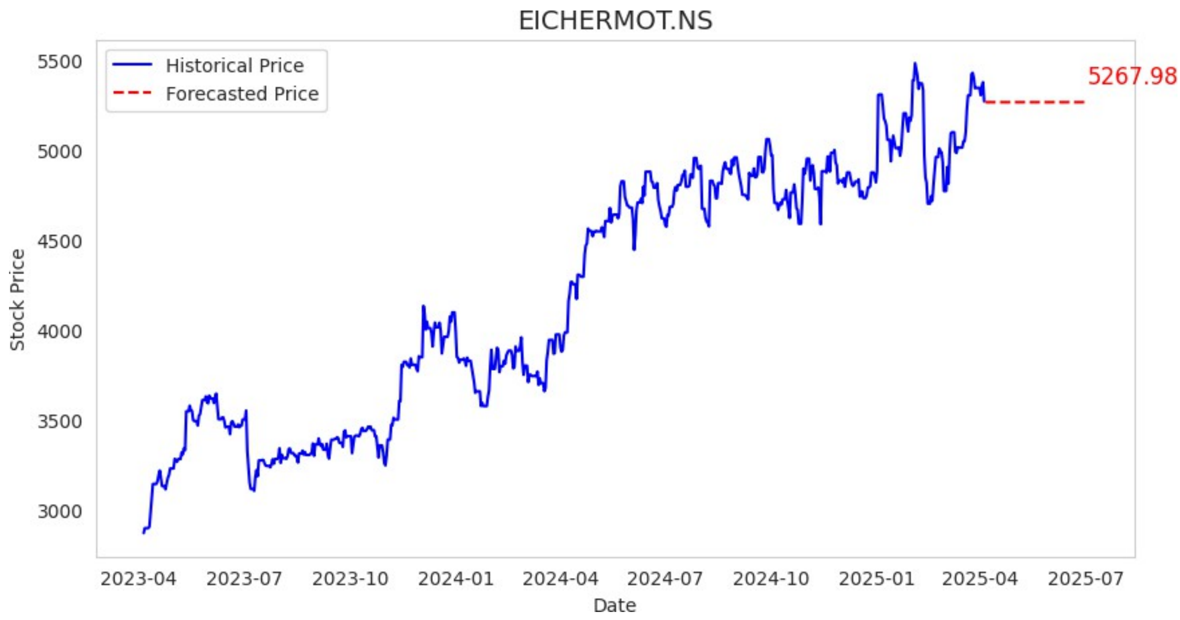


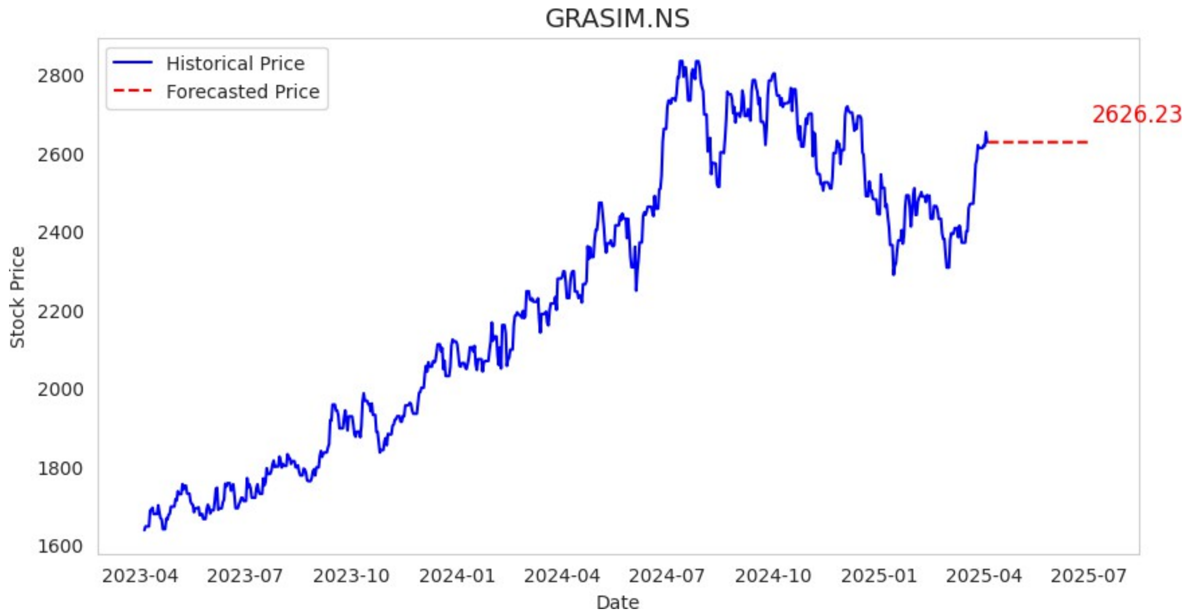


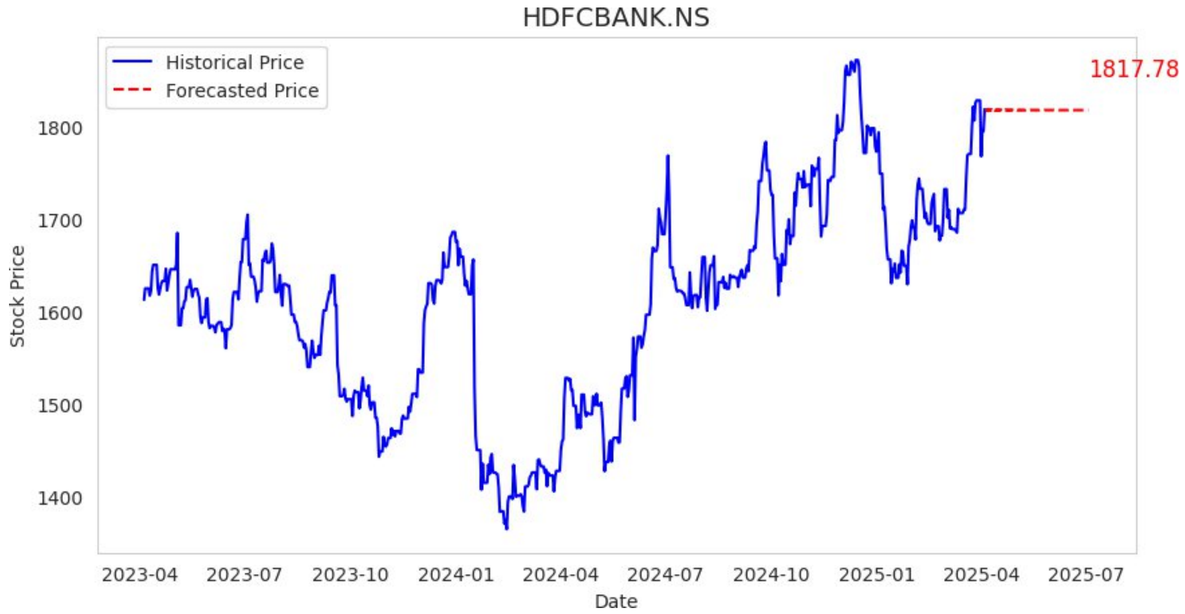

ASIANPAINT.NS
Historical Price
Forecasted Price
2344.53
Stock Price
3400
3200
3000
2800
2600
2400
2200
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

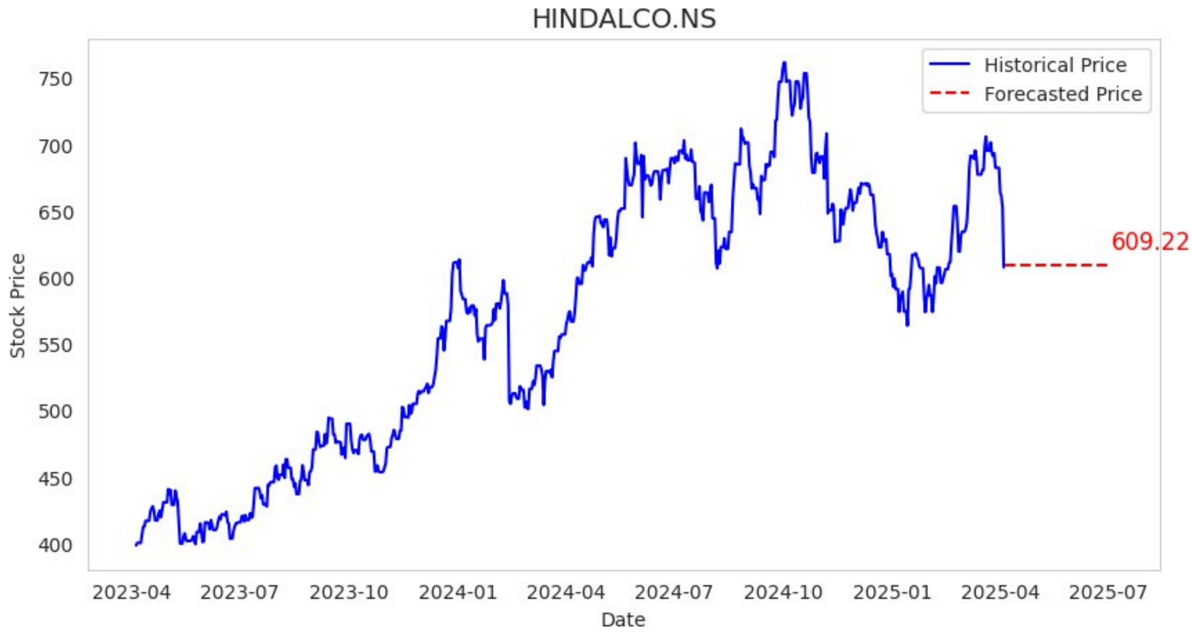

BEL.NS
Historical Price
Forecasted Price
281.75
Stock Price
300
250
200
150
100
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

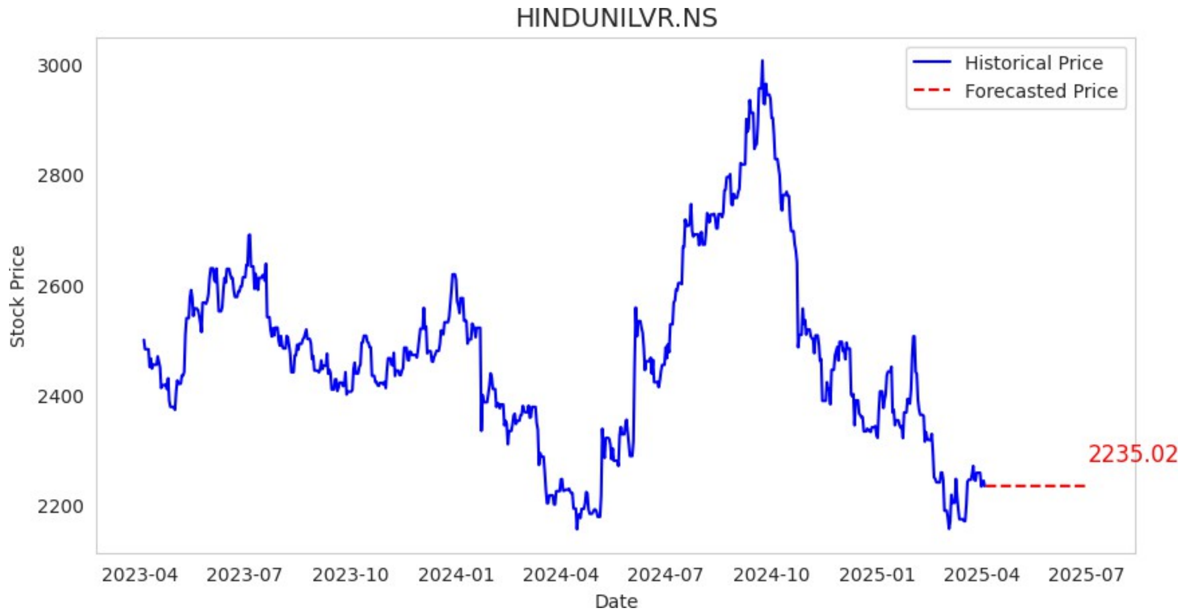

COALINDIA.NS
Historical Price
Forecasted Price
386.75
Stock Price
500
450
400
350
300
250
200
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

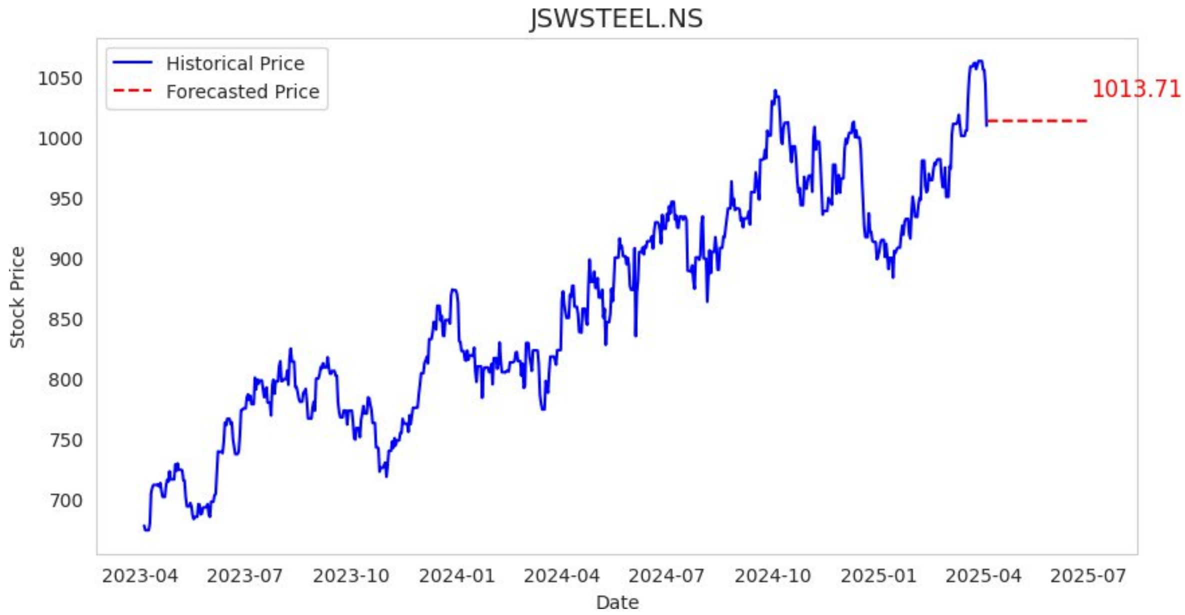

EICHERMOT.NS
Historical Price
Forecasted Price
5267.98
Stock Price
5500
5000
4500
4000
3500
3000
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

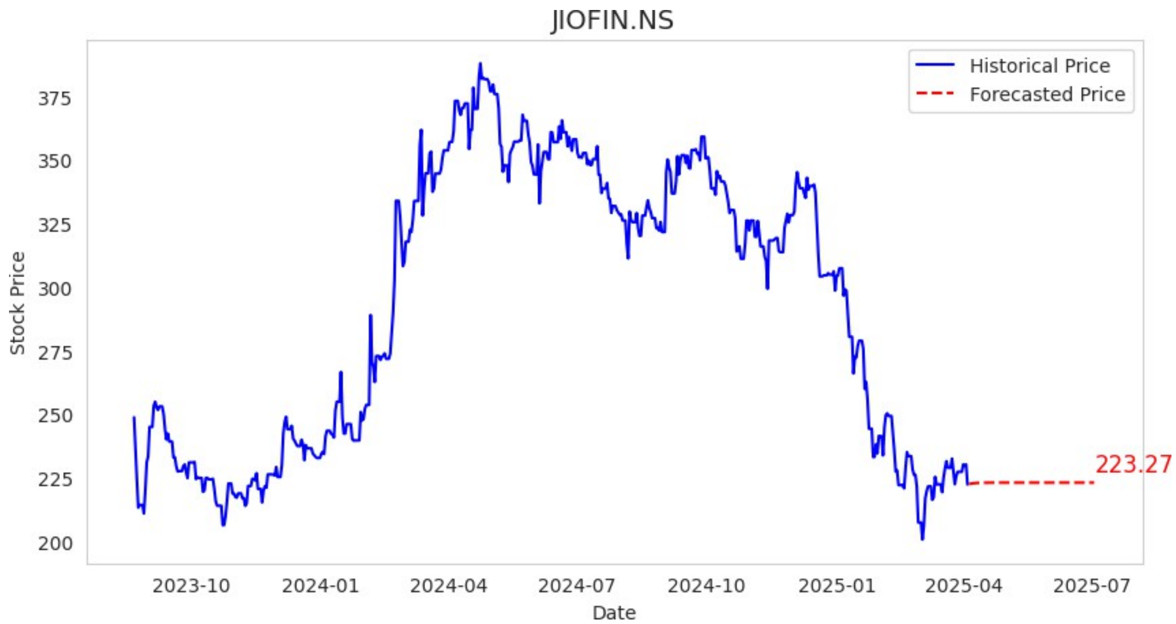

GRASIM.NS
Historical Price
Forecasted Price
2626.23
Stock Price
Date
2800
2600
2400
2200
2000
1800
1600
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07

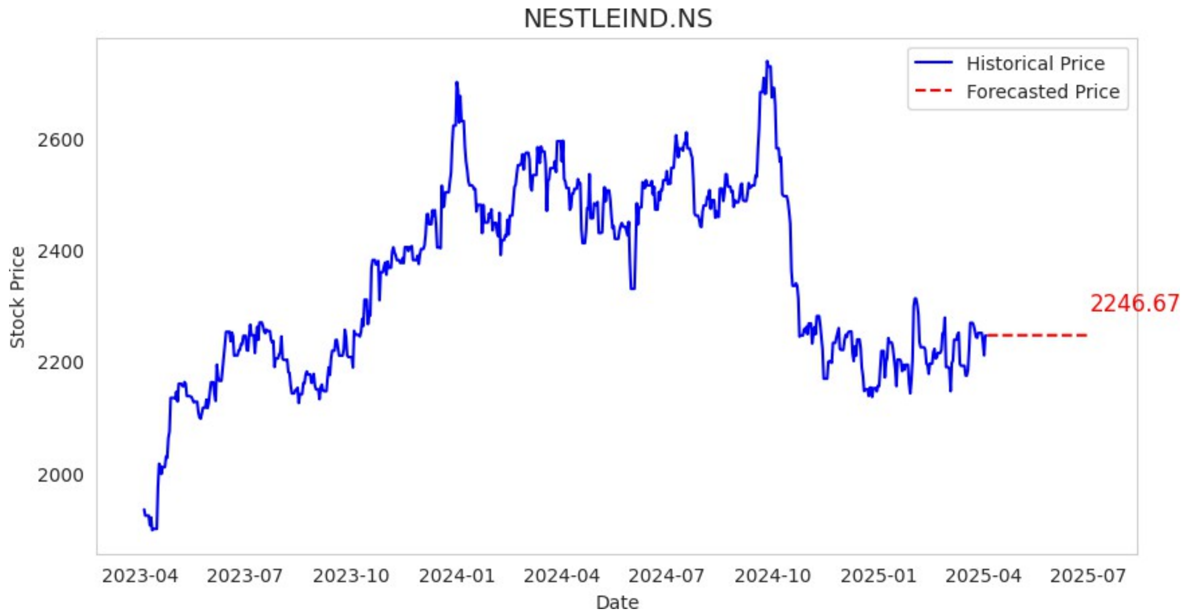

HDFCBANK.NS
Historical Price
Forecasted Price
1817.78
Stock Price
Date
1800
1700
1600
1500
1400
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07

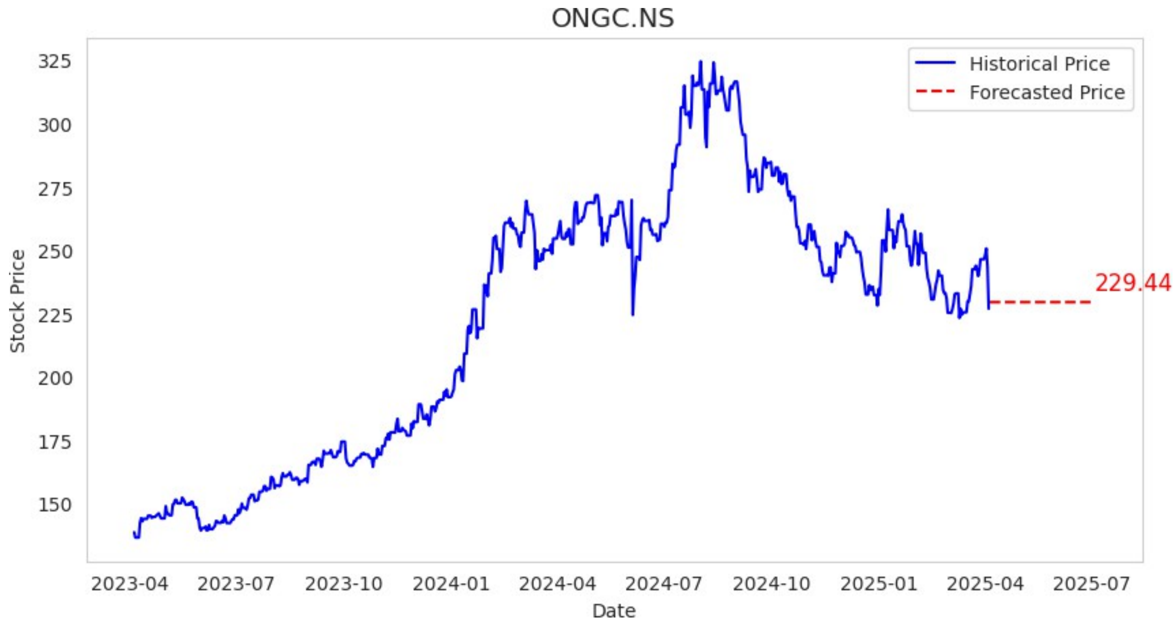

HINDALCO.NS
Historical Price
Forecasted Price
609.22
750
700
650
600
550
500
450
400
Stock Price
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

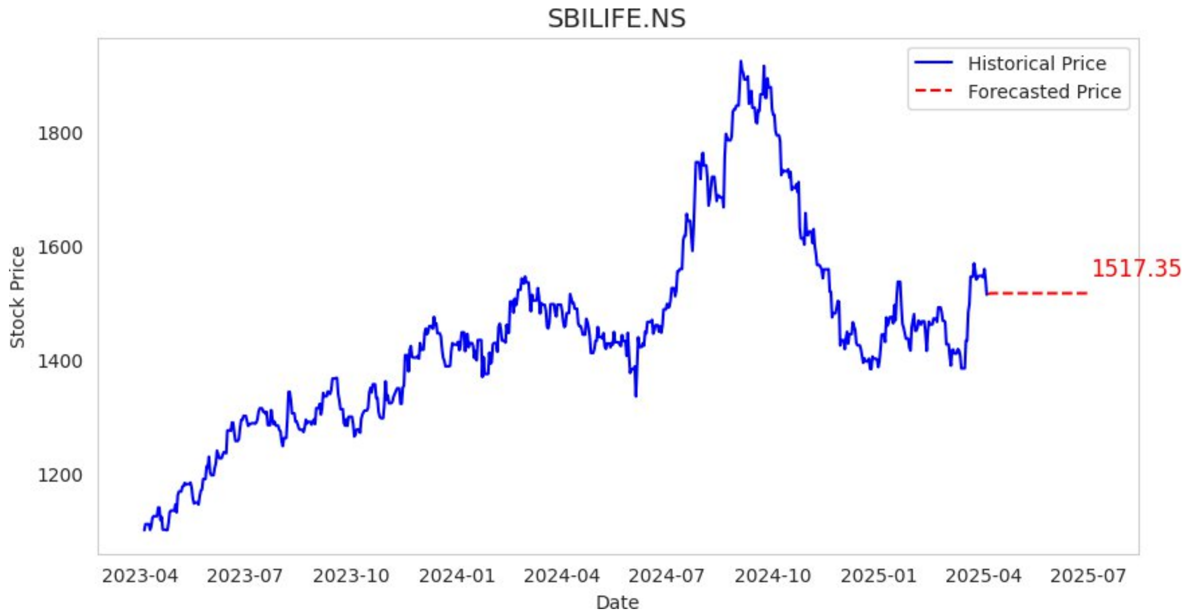

HINDUNILVR.NS
Historical Price
Forecasted Price
2235.02
3000
2800
2600
2400
2200
Stock Price
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

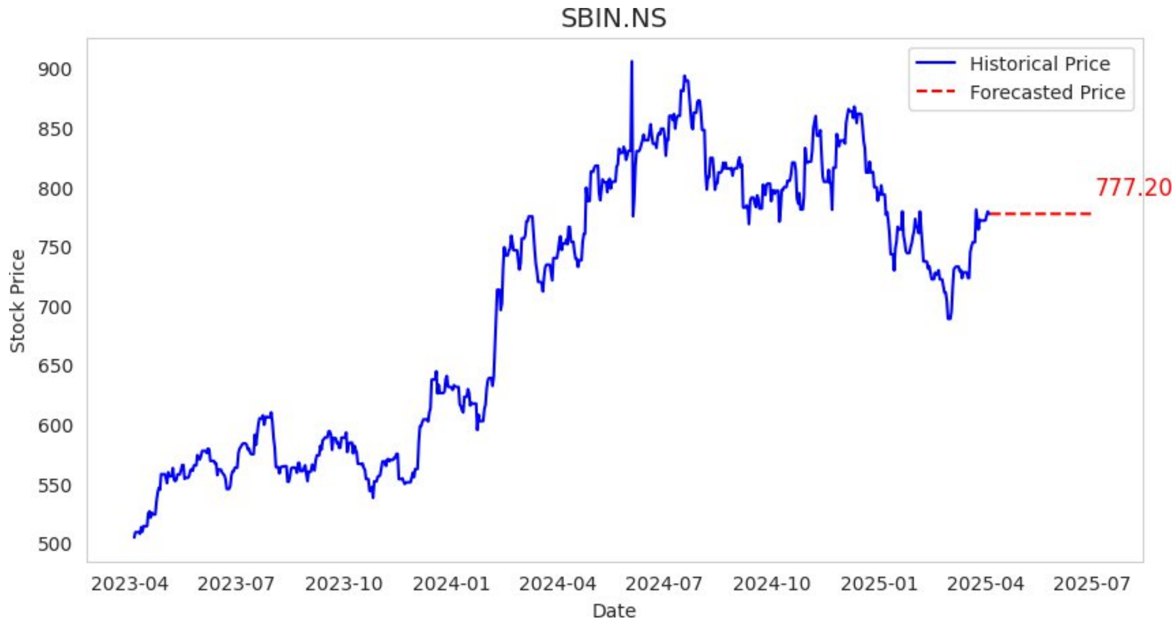

JSWSTEEL.NS
Historical Price
Forecasted Price
1013.71
Stock Price
Date
1050
1000
950
900
850
800
750
700
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07

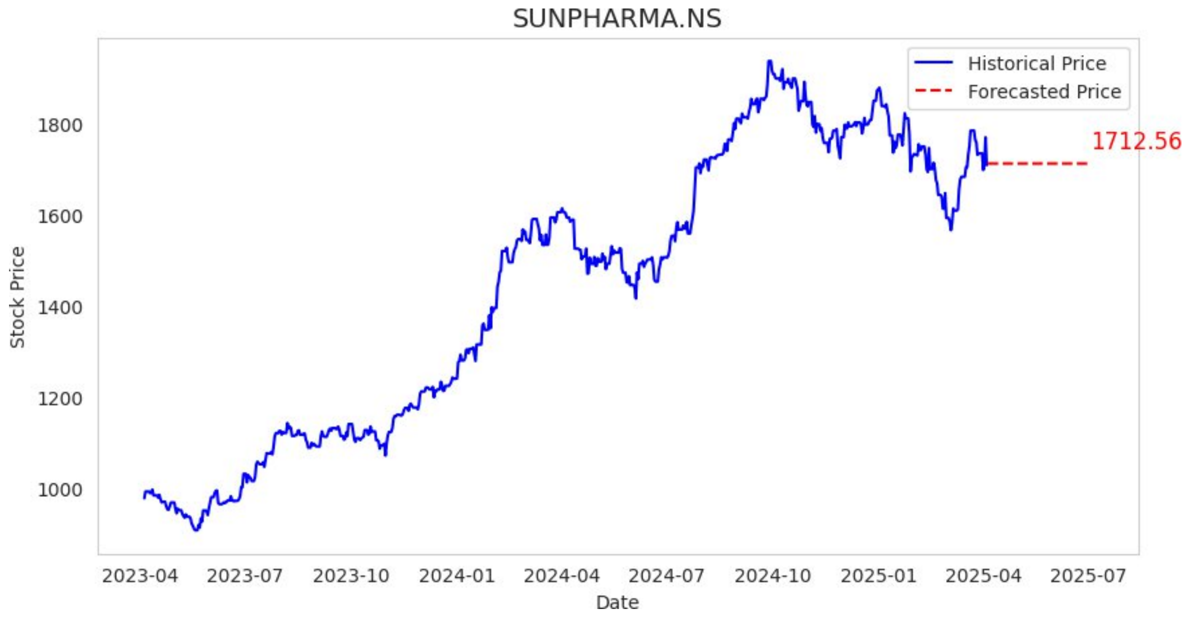

JIOFIN.NS
Historical Price
Forecasted Price
223.27
Stock Price
Date
375
350
325
300
275
250
225
200
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07

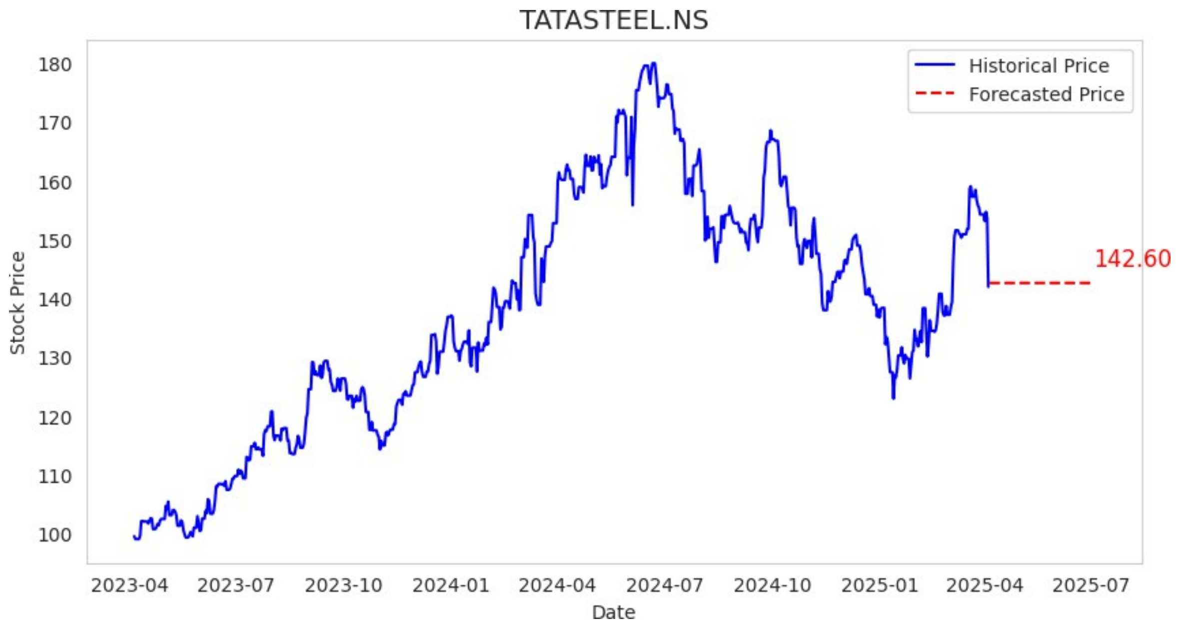

NESTLEIND.NS
Historical Price
Forecasted Price
2246.67
Stock Price
2000
2200
2400
2600
Date
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07

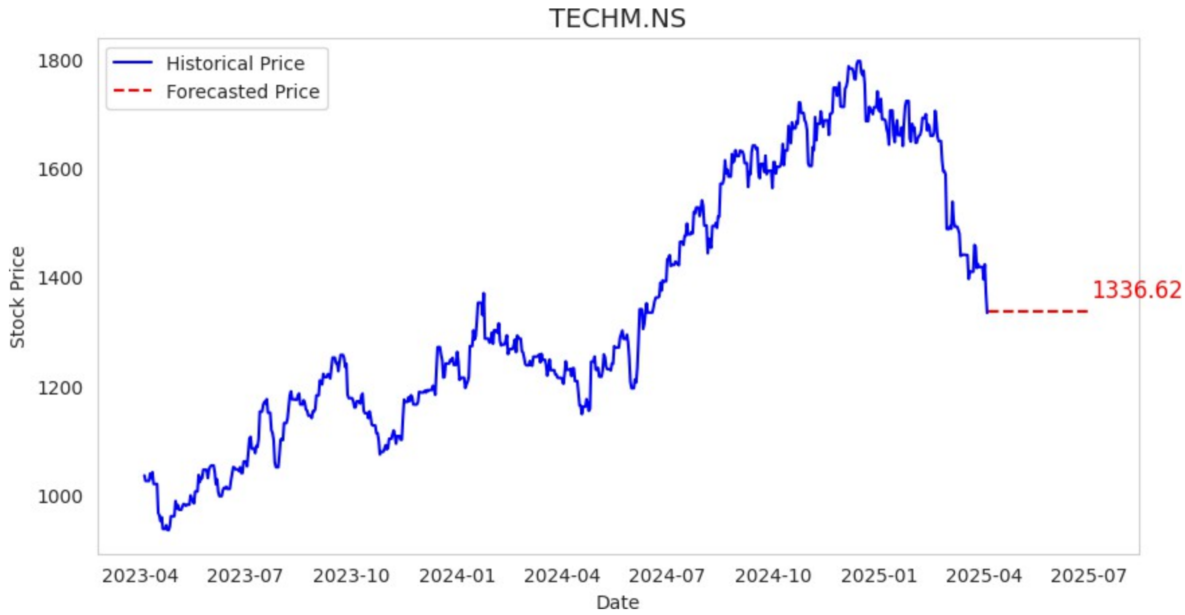

ONGC.NS
Historical Price
Forecasted Price
229.44
Stock Price
150
175
200
225
250
275
300
325
Date
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07

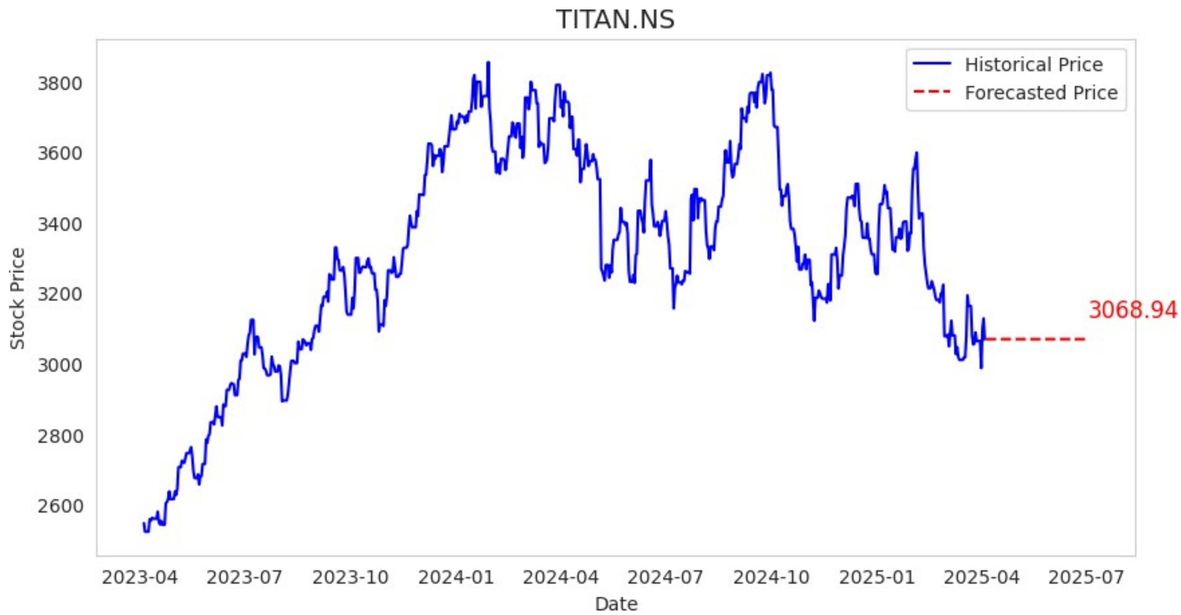

SBILIFE.NS
Historical Price
Forecasted Price
1517.35
Stock Price
1200
1400
1600
1800
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

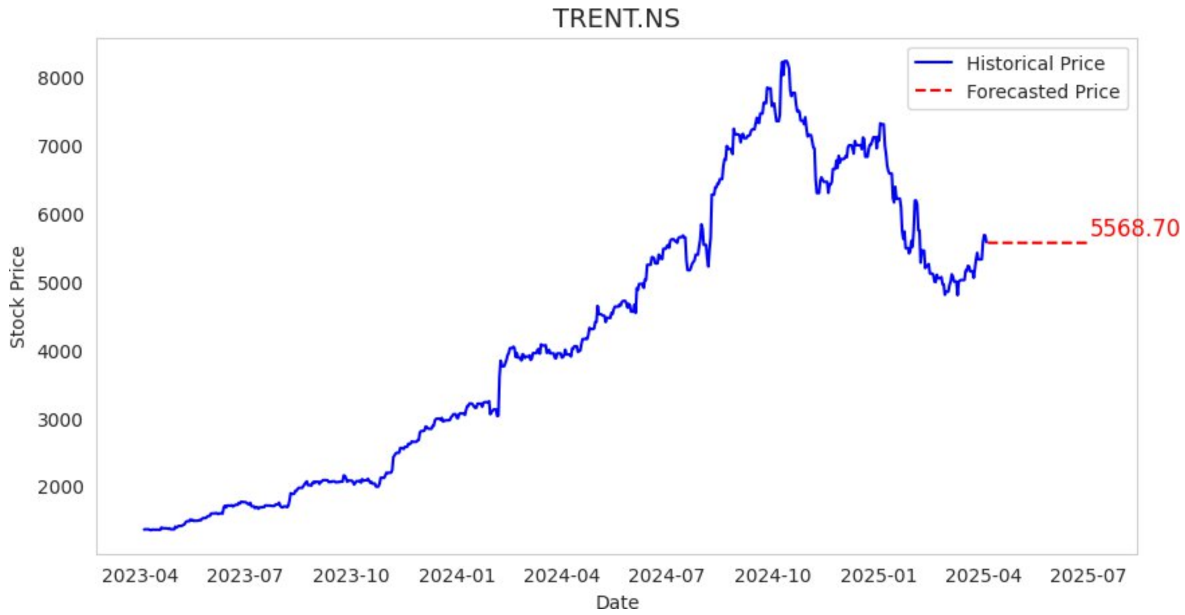

SBIN.NS
Historical Price
Forecasted Price
777.20
Stock Price
500
550
600
650
700
750
800
850
900
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

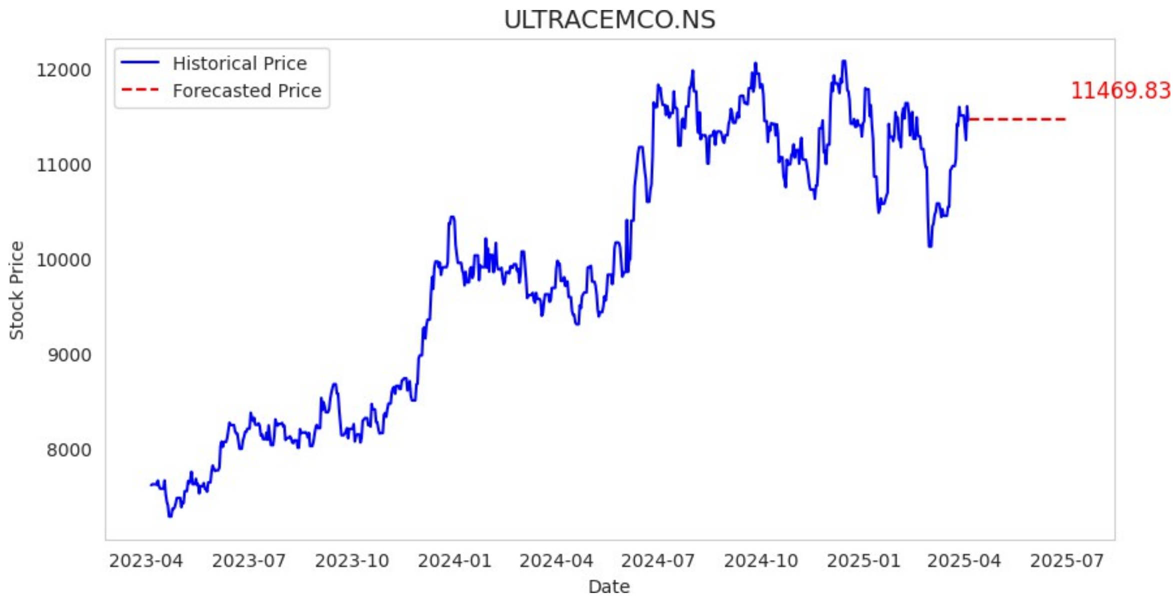

SUNPHARMA.NS
Historical Price
Forecasted Price
1712.56
Stock Price
1000
1200
1400
1600
1800
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

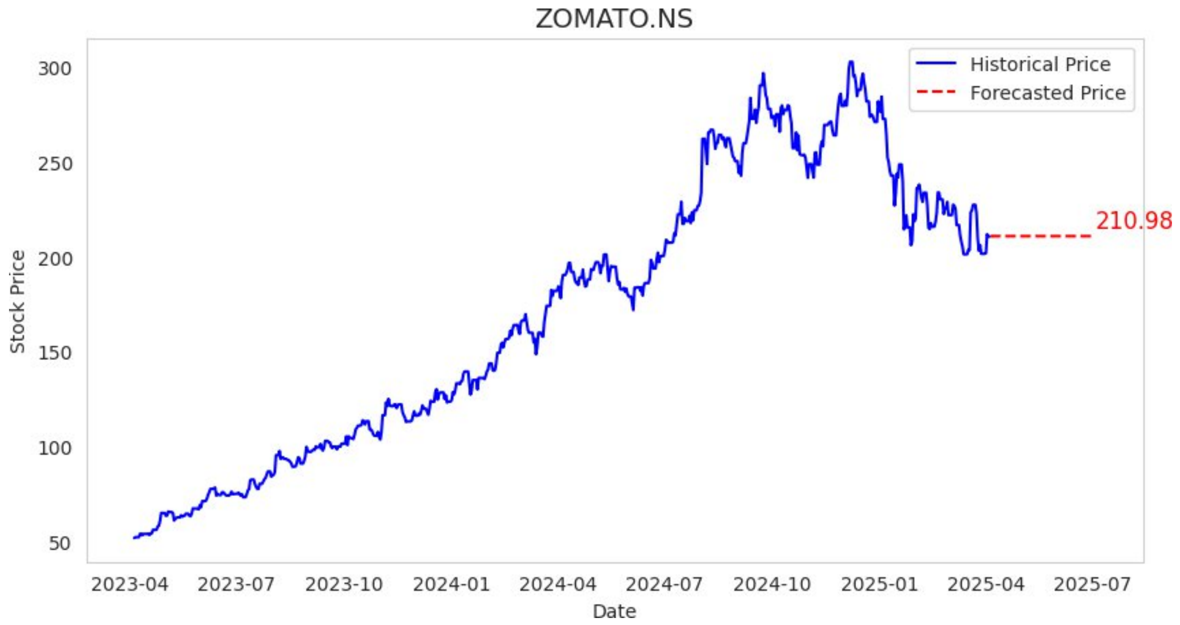

TATASTEEL.NS
Historical Price
Forecasted Price
142.60
Stock Price
100
110
120
130
140
150
160
170
180
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date


TECHM.NS
Historical Price
Forecasted Price
1800
1600
1400
1200
1000
Stock Price
1336.62
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date


TITAN.NS
Historical Price
Forecasted Price
3800
3600
3400
3200
3000
2800
2600
Stock Price
3068.94
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date


TRENT.NS
Historical Price
Forecasted Price
5568.70
Stock Price
8000
7000
6000
5000
4000
3000
2000
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date


ULTRACEMCO.NS
Historical Price
Forecasted Price
11469.83
Stock Price
12000
11000
10000
9000
8000
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date


ZOMATO.NS
Historical Price
Forecasted Price
210.98
Stock Price
300
250
200
150
100
50
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
2025-04
2025-07
Date

Chapter 5

Summary Of Findings, Suggestions and Conclusions

5.1 Introduction to Summary of Findings, Suggestions, and Conclusion

The summary of findings revisits each major insight uncovered during the research process—from identifying mismatches between valuation and performance to classifying stocks into meaningful quadrants. The suggestions section translates these analytical observations into actionable recommendations that investors, portfolio managers, and analysts can adopt to enhance decision-making. The conclusion distils the academic and practical implications of the research and reflects on the study's contribution to existing financial literature. This closing synthesis reinforces how the integrated framework can address real-world challenges in equity selection and valuation analysis within the Indian capital market.

5.2 Summary of Findings

The research aimed to bridge the analytical gap between valuation and performance by proposing a unified quadrant-based framework that evaluates NIFTY 50 stocks using Price-to-Earnings (P/E) multiples and Capital Asset Pricing Model (CAPM) expected returns. The following key findings were derived from the study:

- *Disparity Between Valuation and Performance*

  Several stocks labeled as "Overvalued" based on their P/E multiples were found to outperform market expectations when assessed using CAPM returns. Conversely, certain "Undervalued" stocks failed to generate returns commensurate with their perceived valuation advantage. This mismatch confirms that relying solely on one metric—valuation or performance—can lead to suboptimal investment decisions.

- *Effective Use of PE Multiples Against Benchmarks*

  By comparing each stock's P/E ratio against both the NIFTY 50 index and sectoral averages, the study identified stocks that appeared mispriced within their relative contexts. This dual benchmarking revealed instances where a stock was overvalued in the broader index but fairly valued within its sector, or vice versa.

- *CAPM Provided Risk-Adjusted Insights*

  The CAPM model, incorporating each stock's beta, risk-free rate, and market return, effectively revealed whether a stock's actual return aligned with its risk-adjusted expectation. This helped uncover cases of underperformance despite favorable valuation, highlighting the importance of incorporating market risk into performance assessments.

- *Quadrant-Based Classification Was Intuitively Effective*

  The integration of P/E-based valuation and CAPM-based performance enabled the categorization of stocks into four quadrants:

  - **Undervalued & Outperforming**: Ideal investment opportunities
  - **Overvalued & Underperforming**: High-risk, unattractive stocks
  - **Undervalued & Underperforming**: Potential turnaround candidates
  - **Overvalued & Outperforming**: Momentum-driven, speculative plays This framework offered a simplified yet powerful tool for investment screening.

- *Underperforming Overvalued Stocks Identified for Risk Monitoring*

  A focused analysis of stocks falling into the Overvalued and Underperformed quadrant identified high-risk equities that may warrant caution or rebalancing. These stocks were further studied using ARIMA forecasting and performance comparison with the Nifty 50, confirming their lagging trends.

- *Historical Investment Simulation Validated Performance Trends*

  Simulating ₹10,000 investments across the underperforming overvalued stocks versus the Nifty 50 index demonstrated how these stocks eroded relative investor wealth during the observed period. This validated the practicality of the quadrant model and its ability to flag weak investment candidates ahead of time.

5.3 Implications of the Study

5.3.1 For Investors and Portfolio Managers

The composite framework developed in this research—integrating P/E multiples with CAPM returns—offers investors a more comprehensive and reliable way to assess stock attractiveness.

The quadrant-based classification system enables better decision-making, as it goes beyond relying on single indicators (such as P/E ratios or CAPM returns) in isolation. By using this method, investors can avoid common pitfalls such as overvaluing stocks with good past performance but poor future potential or undervaluing stocks that are underperforming due to temporary factors. The study emphasizes the importance of incorporating both valuation and performance metrics, which can significantly enhance risk-adjusted return assessments in real-world portfolio management.

5.3.2 For Financial Analysts and Equity Researchers

The research underscores the need for a more integrated approach when evaluating stocks. Traditional valuation techniques, such as P/E multiples, can often be misleading if not complemented with performance metrics like CAPM, which account for the risk factors in stock returns. Analysts who apply this dual-metric approach will be able to provide more balanced recommendations, ensuring that stocks classified as overvalued do not still hold hidden potential for high returns due to positive risk-adjusted performance. This approach provides a more holistic method for identifying the true attractiveness of stocks, especially when assessing large indices such as the NIFTY 50.

5.3.3 For Academic Research

This study contributes to the growing body of knowledge on stock valuation by highlighting the limitations of using single metrics for stock analysis and proposing a novel, unified framework. The approach presented in this dissertation could be extended to other stock indices or markets globally, which may offer further insights into cross-market valuation-performance relationships. Additionally, it encourages future research to explore more complex hybrid models that integrate other fundamental, technical, and behavioural finance factors, possibly enhancing predictive power.

5.3.4 For Stock Market Participants

Market participants—ranging from retail investors to institutional investors—can benefit from understanding the interaction between stock valuation and risk-adjusted performance. The study provides practical tools for identifying stocks that are underperforming or mispriced relative to their risk, which can help in portfolio rebalancing and risk management. By leveraging the results of this research, market participants can better align their investment decisions with both valuation benchmarks and expected market returns.

5.4 Suggestions

For Investors:

- Integrate Valuation and Performance Metrics: Investors should adopt the quadrant-based framework proposed in this study, combining valuation metrics (such as P/E multiples) and performance indicators (like CAPM returns). This dual-metric approach provides a more robust basis for evaluating stocks and reduces the risk of over-relying on a single metric.
- Focus on Risk-Adjusted Returns: Rather than relying solely on price or earnings figures, investors should incorporate risk-adjusted returns into their decision-making process. Stocks that may seem overvalued based on their P/E ratios could still outperform if they deliver strong risk-adjusted returns, which should be factored into investment strategies.
- Monitor Stock Performance Periodically: Regularly update the performance of stocks, especially in volatile market conditions. The CAPM performance analysis in this study indicates that performance can significantly change over short periods (like 3 months), so continuous monitoring can help adjust portfolios accordingly.

For Portfolio Managers:

- Adopt a Holistic Evaluation Model: Portfolio managers should consider both valuation and performance when selecting stocks for investment. By using a composite framework that integrates P/E multiples and CAPM-based expected returns, they can construct more diversified and balanced portfolios that minimize risks while maximizing potential returns.
- Use the Framework for Portfolio Rebalancing: This study suggests that the overvalued and underperforming stocks identified in the research should be closely monitored for potential portfolio rebalancing opportunities. Regular portfolio reviews based on this framework will help ensure that portfolios are aligned with the current market dynamics.

For Financial Analysts:

- Expand Analytical Approaches: Financial analysts should expand their analysis beyond traditional valuation methods (e.g., P/E ratios) and incorporate performance-based metrics such as CAPM expected returns and risk-adjusted returns. This will provide a more nuanced understanding of stock potential, especially in cases where traditional metrics might suggest misleading conclusions.
- Focus on Sectoral Analysis: Analysts should consider applying the same dual-metric approach at the sectoral level, examining how stocks within specific industries perform

relative to the broader market or sector index. Sectoral performance analysis will help identify industry-specific trends and potential outperformers.

For Academics and Researchers:

- Explore Additional Hybrid Models: Future research could explore the integration of other metrics, such as price momentum, earnings growth, and macroeconomic factors, into the composite framework. Developing hybrid models that combine both quantitative and qualitative factors could offer more predictive power in identifying undervalued or overvalued stocks.
- Apply the Framework to Other Markets: While this study focuses on the NIFTY 50 index, similar methodologies can be applied to other stock indices and markets globally. Expanding the framework's application could validate its effectiveness across different economic environments and provide a broader understanding of stock valuation-performance relationships.

For Regulatory Bodies and Investment Advisors:

- Encourage the Use of Dual Metrics in Stock Evaluation: Regulators and investment advisors can encourage the use of a broader set of evaluation metrics that include both valuation and performance measures. This could be done by promoting standardized risk-adjusted performance measures alongside traditional valuation ratios in company disclosures.
- Improve Investor Education: Regulatory bodies and investment advisors should focus on educating investors about the importance of considering multiple factors when evaluating stocks. Financial literacy programs can incorporate the composite valuation-performance framework, emphasizing the importance of balancing P/E ratios and expected returns.

For Stock Market Participants:

- Consider Risk-Adjusted Metrics in Trading Decisions: Stock traders should be aware that high-performing stocks are not always the best investment choices if they carry high risks. Focusing on risk-adjusted metrics will provide a clearer picture of the actual potential for returns, allowing traders to make more informed decisions.
- Diversify Using the Quadrant Approach: The quadrant-based classification system proposed in the study can be used by traders and investors alike to create diversified portfolios. This model allows for balancing risk and reward by investing in stocks from

various quadrants that are overvalued but outperforming, or undervalued but underperforming.

5.5 Conclusion

This study contributes significantly to the field of stock valuation by addressing the limitations of using standalone metrics like the Price-to-Earnings (P/E) ratio and Capital Asset Pricing Model (CAPM) returns. It proposes a novel composite framework that integrates both valuation and performance measures, providing a more comprehensive approach to stock evaluation. By classifying NIFTY 50 stocks into four quadrants—Overvalued & Outperforming, Overvalued & Underperforming, Undervalued & Outperforming, and Undervalued & Underperforming—the study enhances investors' ability to identify mispriced stocks and reduce the risk of poor investment decisions.

The findings indicate that traditional valuation metrics alone do not capture the full picture of a stock’s potential, as certain overvalued stocks may still outperform the market in terms of risk-adjusted returns, while others may underperform. The study’s integrated approach demonstrates that investors should look beyond basic valuation to assess performance metrics, such as CAPM expected returns, to make more informed investment choices. The performance analysis further highlights the dynamic nature of stock returns, emphasizing the need for continuous monitoring to assess whether a stock remains undervalued or overvalued relative to its market performance.

Bibliography

1. Rossi, M. (2016). *The capital asset pricing model: A critical literature review*. ResearchGate. https://www.researchgate.net/publication/307180424_The_capital_asset_pricing_model_a_critical_literature_review

2. Rabha. (2021). *Is CAPM still valid in today's market scenario?* Indian Journal of Capital Markets. https://www.indianjournalofcapitalmarkets.com/index.php/IJF/article/view/169518

3. Musawa, N., Kapena, S., & Shikaputo, C. (2020). *A better stock pricing model: A systematic literature review*. Journal of Economic and Financial Sciences, 13(1). https://jefjournal.org.za/index.php/jef/article/view/472

4. Various Authors. (2023). *The six decades of the capital asset pricing model: A research agenda*. Journal of Risk and Financial Management, 16(8), 356. https://www.mdpi.com/1911-8074/16/8/356

5. Financial Times. (2024). *The valuation mystery: More clues*. https://www.ft.com/content/62c362f2-0c58-42ed-a8ff-19ace3a11821

6. Chaudhuri, A. (2024, June). *Predicting stock prices using future EPS estimates and historical PE ratios: A dual-bound approach*. ResearchGate. https://www.researchgate.net/publication/381406895_Predicting_Stock_Prices_Using_Future_EPS_Estimates_and_Historical_PE_Ratios_A_Dual-_Bound_Approach

7. [Author not specified]. (2025, March). *Analysing and forecasting P/E ratios using investor sentiment in panel data regression and LSTM models*. Journal of Forecasting. https://www.sciencedirect.com/science/article/pii/S1059056025000036

8. Baird, G., Dodd, J., & Middleton, L. (2020, January). *A growth adjusted price-earnings ratio*. arXiv. https://arxiv.org/abs/2001.08240

9. Rossi, M. (2016, September). *The capital asset pricing model: A critical literature review*. International Journal of Business and Emerging Markets, 8(3), 265–279. https://www.inderscience.com/info/inarticle.php?artid=78682

10. [Author not specified]. (n.d.). *Implementing the capital asset pricing model in forecasting stock returns: A literature review*. International Journal of Business Administration. https://journal.formosapublisher.org/index.php/ijba

11. Beyaztas, U., Ji, K., Shang, H. L., & Wu, E. (2025). *Stock return prediction based on a functional capital asset pricing model*. https://arxiv.org/abs/2504.01239

12. Hatemi-J, A. (2024). *An asymmetric capital asset pricing model*. https://arxiv.org/abs/2404.14137

13. Atsiwo, A., & Sarantsev, A. (2024). *Capital asset pricing model with size factor and normalizing by volatility index*. https://arxiv.org/abs/2411.19444

14. Wu, L., & Xu, S. (2023). *A capital asset pricing model based on the value at risk under asymmetric Laplace distribution*. https://www.emerald.com/insight/content/doi/10.1108/K-03-2023-0416/full/html