

Chapter 1: Introduction

1.1. Background and Context

In a rapidly changing world where almost everyone is affected by rising inflation, people are increasingly becoming aware of the importance of investing, and their own financial goals. The purpose of this project is to create a data-driven solution for such individuals to help them select equity portfolios in the Indian market. In contrast to conventional investment methods, this strategy will analyse market benchmark(s) and their constituents and find outperforming assets within those indices. The end product will be a code repository and a web application that will display portfolios suggested by the strategy developed. Using that, a potential investor can select a portfolio that aligns with their financial goals and make a well-informed investment decision. With the help of historical data, this project intends to offer a quantitative investment strategy that could be customised to suit the specific demands and risk tolerances of individual investors.

Traditional algorithmic trading strategies often rely on simplistic buy/sell signals. In contrast, the primary objective here is to utilise historical data, quantitative analysis, and self-defined metrics to construct investment portfolios that offer varying risk-return profiles. This approach is critical in today's investment landscape, where traditional methods may no longer suffice. The project proposes a multi-faceted methodology that goes beyond standard market analysis. It includes an extensive backtesting phase, spanning various market cycles, to validate the robustness of the strategies developed. Additionally, the strategies developed in this project are not limited to selecting optimal portfolios but also give a date on which the portfolio should be rebalanced, i.e., which existing stocks must be sold which must be bought, and on which date.

1.2. Motivation and Significance

Modern investors want more than simply financial rewards; they also want empowerment, ownership, and the capacity to make data-driven decisions. In the exponentially growing Indian financial market, there are many investment options, which can be overwhelming for individual investors. This project intends to help simplify investing for them with a structured approach. I am also motivated by the idea that historical data can greatly assist investment decisions. My project will use strategies that consider various statistical factors to make investment choices more organised. The goal is to provide individuals with the tools they need to make informed decisions in the dynamic Indian financial markets. Throughout the project, I will try to keep my code as modular as possible, making it easier to adapt the same to other markets and customisable to tweak the strategy in the future.

1.3. Problem Statement

To create a novel quantitative approach to portfolio selection and time-based rebalancing to outperform market benchmark indices.

1.4. Objectives

- Develop a historical data-driven investment strategy that focuses on analysing market indices and identifying a good combination of assets within those indices.
- Create a user-friendly app/interface that allows individual investors to choose their portfolios based on the strategy's recommendations.

- Evaluate the performance of the strategy through backtesting and historical data analysis, providing investors with insights into potential portfolio performance.

1.5. Scope and Limitations

The scope of this project includes data collection, data wrangling, data analysis, formation of strategies, backtesting of strategies, analysis of results, and creation of a web application for the same. The scope does not include consideration of market microstructure to a complete extent, although some aspects of market microstructure have been considered. These include the consideration of current market prices of the stocks for weight allocation of the total investment amount and rebalancing costs that include brokerages, taxes, etc. There are a few more identified limitations of the project that are beyond the scope of the project such as making the investment strategies survivorship-bias free. This will be discussed in detail in a later section of this report.

Chapter 2: Review of Related Works

2.1. Literature Review

In the dynamic realm of portfolio management, where risk and return often go hand-in-hand, the progression from fundamental ideas to modern quantitative methods reflects an intellectually stimulating, innovative, and adaptive journey. This chapter takes readers on a tour, tracing the development of portfolio theory from its conception to the complex methods used today, with a particular emphasis on the rapidly expanding Indian market.

Harry Markowitz's revolutionary mean-variance optimisation theory, paved the way for many future generations of investors. The path towards an optimal portfolio is described in "60 Years of Portfolio Optimisation: Practical Challenges and Current Trends". The article shows how this pursuit of finding a balance between risk and return has changed over time, according to the shifting dynamics of the financial markets. It not only demonstrates the strength of Markowitz's basic ideas but also outlines the various ways in which they have been expanded upon, adapted, and improved to meet the demands of modern markets. The methodology of this project is based along the same lines of exploring the nuances of Markowitz's theory and creating a novel customised solution using it.

Another study "Quantitative Portfolio Selection: Using Density Forecasting to Find Consistent Portfolios", emphasises a significant transition towards quantitative and data-driven approaches that utilise computational finance to uncover coherent portfolios. The authors explain the accuracy and adaptability that these approaches provide, emphasising how effective they are in interpreting the intricate patterns found in market data. This paradigm shift is consistent with our project's main goal, which is to use historical data and advanced data analytics to create profitable and resilient portfolios.

The process of creating optimised portfolios involves intricate interactions between three main components: risk factors, excess return factors, and objective-specific constraints. Factor Alignment Problems (FAP) are the collective term for the issues that result from these three elements' mutual misalignment. This study, "Factor Alignment Problems and Quantitative Portfolio Management" explores the complex issues surrounding portfolio management in addition to FAPs, such as handling high-dimensional data. The study provides an overview of the difficulties faced by portfolio managers in the volatile markets of today, highlighting the necessity for creative approaches that can deftly handle these situations.

The article "Creating Value in Small-cap Firms by Mitigating Risks of Market Volatility" dives deeper into the diverse world of the Indian market. This research examines the difficulties for investors, particularly for small-cap companies. My aim of including small-cap and mid-cap companies in the strategy of my project is complemented by this research and the insightful observations made by the writers about the resilience of such stocks, making a strong argument for their inclusion in investment portfolios.

The collective knowledge extracted from these academic publications forms a distinct vision for my project. I want to bridge the gap between theory and practice by building an approach that is both novel and firmly rooted in the realities of the Indian market and fundamental portfolio management concepts. This project aims to rework the characteristics of investing techniques, making them more approachable, flexible, and sensitive to the investor's demands.

To summarise, this section of the report credits the journey of ideas that have moulded the discipline of portfolio management. The core methodology of the project acts as the next natural step in the journey of innovation in investing that engages with the present, looks forward to the future, and borrows from the past.

2.2 Theoretical Foundations and Important Terminologies

Apart from leveraging my coding and analytical skills, this project requires prerequisite knowledge of some concepts mostly used in the financial industry for portfolio selection, optimisation, rebalancing, and evaluating trading and investing strategies. This section will describe some of the important theoretical concepts and technical terminologies that will be used frequently throughout the report and are crucial to the project.

- **Sharpe Ratio:** The Sharpe ratio is an important ratio in the investment world, which is used to measure the performance of a security or a portfolio compared to a risk-free asset, after adjusting for its risk. A higher Sharpe ratio is considered to be a better investment choice. By definition, it is the difference between the return of the portfolio and the return of the risk-free asset divided by the standard deviation of the portfolio. Sharpe Ratio is denoted by:

$$\text{Sharpe Ratio} = \frac{(R_p - R_f)}{\sigma_p}$$

where, R_p is the return on the investment, R_f is the return of a risk-free asset, and p is the standard deviation of the investment.

- **Annualised Return:** Annualised return is a geometric average of an investment's earnings in a year. The formula for annualised return is:

$$\text{Annualised Return} = (1 + \text{Total Return})^{\frac{1}{n}} - 1$$

where n is the duration (in years) for which the total return is calculated.

- **Backtesting:** Backtesting in the context of a trading/investing strategy means testing a strategy to assess its accuracy using historical data. It involved devising a strategy and then simulating its application on a historical date so that its performance could be measured and analysed.
- **Stock Screening/Selection:** Stock screening or stock selection is a process of selecting stocks based on certain quantitative criteria or financial metrics. In a quantitative context, such as this project, it heavily relies on creating mathematical and statistical models to choose stocks that are likely to give good returns. The criteria can vary depending on the investment goals and objectives of the investor.
- **Weight Allocation:** In portfolio management, weight allocation is the process of figuring out how much of each investment or asset should be in a portfolio. Usually, this allocation is stated as a percentage of the entire value of the portfolio. Weight allocation can be done in many ways and there is no defined way to do so. The method of weight allocation depends on the investor's unique objectives, risk tolerance, investment horizon, and market outlook is the aim of weight allocation.
- **Markowitz Portfolio Theory:** The Markowitz Portfolio Theory, also known as Modern Portfolio Theory (MPT), is a framework that was developed by Harry Markowitz in 1952. The framework is for putting together a portfolio in a way that the expected return is maximised for a particular degree of risk, or the expected risk is minimised for a given level of returns. The theory showcases the advantages of diversity and the idea that a portfolio's risk and return should be evaluated based on the correlation of its stocks instead of considering them independently.
- **Tangency Portfolio:** The portfolio on the efficient frontier that provides the maximum predicted return per unit of risk is represented by the Tangency Portfolio, an idea derived from

the Markowitz Portfolio Theory. Because it is the point where the capital market line (CML) is tangent to the efficient frontier, the portfolio is known as the "tangency" portfolio. The risk-free rate is the intercept of the capital market line, which illustrates the risk-return trade-off for efficient portfolios. Because it offers the finest mix of risk and return, the Tangency Portfolio is regarded as ideal.

- **Survivorship Bias:** The tendency to analyse the entities that have "survived" a process of selection while disregarding those that have not is known as "survivorship bias." It has an impact on the evaluation of stocks or investment funds in the context of finance. Specific to my project, the impact of survivorship bias and the details will be mentioned in a later section. A consequence of this bias is that the backtesting process will not consider failed or delisted companies, which might result in an excessively positive performance assessment for a strategy, which is not true.
- **Benchmark Index:** A portfolio or security's performance can be evaluated with a benchmark index. Typically, benchmarks are indexes made up of securities that represent a specific market or a subset of it. They make it possible for managers and investors to assess how their risk, volatility, and return on investment compare to industry or market averages. Typical examples are the S&P 500, which represents the US equities market, and the NIFTY 50 and SENSEX which represent the Indian equity market.
- **Time-Based Rebalancing:** Time-based rebalancing is a portfolio management method in which the re-allocation of funds, and in some cases even removal and addition of assets, in the portfolio is done at regular intervals, making the portfolio adapt to the dynamic market conditions. This could occur on a monthly, quarterly, yearly, or other predetermined schedule. Maintaining the portfolio's initial or intended risk level and asset allocation over time is the goal.

- **Correlation Matrix:** In statistics and finance, the correlation matrix is an instrument used to measure the relationship between several assets. It is used in portfolio management to comprehend the links among various asset classes or equities. When building a portfolio, this matrix is essential for risk management and diversification.
- **Market Capitalisation Brackets:** The term "brackets" describes how businesses are categorised according to their market capitalisation, or the total market value of their outstanding shares of stock. The current share price multiplied by the total number of shares in the market is how market capitalisation is calculated for a company. Generally, businesses are divided into groups like large-cap, mid-cap, small-cap, and micro-cap, and a wide range of market capitalisations and factors like firm size, risk, and growth potential are reflected in each bracket. Large-cap companies, on the other hand, are typically thought of as being less hazardous and more stable, whereas small-cap companies might have better growth potential, but at the expense of greater risk.

Chapter 3: Methodology

3.1. Strategy Development and Overview of Variations

The foremost step in developing strategies is to choose a base universe, i.e., a common pool of stocks that act as the foundation for the strategies to be built upon. The selection of the base universe is an essential step in strategy development because it can highly influence the performance of the strategies. The selection of the base universe for this project was made by focusing on getting a large pool of diverse stocks that include large-cap, mid-cap, small-cap, and even micro-cap companies. The idea was to create a strategy that capitalised on the extremely high returns of small-cap and micro-cap companies while giving a safety net by including the more stable and less risky large-cap and mid-cap companies.

The most commonly referred benchmark indices for the Indian market include SENSEX and NIFTY 50, which only include 30 and 50 companies respectively, that too from a very small segment of companies with an extremely large market capital. Apart from the fact that both of these indices don't include companies with diverse market capital, they also include a very small number of companies and building a strategy on the analysis of such few companies wouldn't have yielded the kind of results that the project aimed for.

Hence, my focus shifted to considering the constituents of indices like NIFTY 500 or NIFTY TOTAL MARKET which include 500 and 750 companies respectively. NIFTY 500 includes the top 500 listed companies in India by market capitalisation and it was a very good base to start with given that it represents 96.1% of the total free-float market capitalisation of India and 96.5% of the total turnover of the National Stock Exchange (NSE). This would include not just large-cap and mid-cap companies but also some small-cap companies. However, I wanted to

include companies that are emerging and even smaller by market capitalisation and do not fall in the top 500 which is why I chose the NIFTY TOTAL MARKET index which included the same 500 companies from the NIFTY 500 but also include 250 other companies from the NIFTY MICROCAP 250 index. Having 750 companies gave both, a huge and diverse market-cap pool for the base universe for the strategy.

The strategy can be broken down into three main components. This section will get into the details of all of these three components and the variations between them and then discuss how they are used together in conjunction for successful outcomes.

3.1.1. Stock Selection Methods

There are a total of 8 stock selection methods defined for this project, coded from S1 to S8. Odd methods, i.e., S1, S3, S5, and S7 have a holding/rebalancing period of 1 quarter (3 months), whereas the even methods, S2, S4, S6, and S8, have a holding/rebalancing period of 1 month. Except for the differences in the holding periods of the methods, the consecutive methods have no other difference. For example, S1 and S2 are the same except for the difference in their holding period, and so on till S7 and S8. However, a different holding period would lead to differences in the way the stock prices are being analysed.

Step 1: The foremost step in all of the methods is to generate periods of the duration of the defined holding period for the previous 3 years of the buying date. This means for methods with a one-month holding period, all the one-month periods will be generated for the last three years of the buying date that start every 10, 20, and 30 days, i.e., for a buying date of 01/01/2024 the code would create all the one-month periods starting every 1st, 11th, and 21st of every month from January 2021 to December 2023. This would result in about 36 one-month periods

in a year, and a total of 108 different one-month periods in three years. Similarly, for methods with a three-month holding period, all the three-month periods will be generated for the last three years of the buying date that start every 30 days, i.e., for a buying date of 01/01/2024, the code would create all 3 month periods starting every 1st of every month from January 2021 to December 2023. This would result in about 12 three-month periods in a year, and a total of 36 different three-month periods in three years.

Step 2: This step includes a combination of various filters applied to all the.

There are 6 different filters in total:

- Filter 1: select stocks with daily average returns of all the quarterly/monthly periods > 0 .
- Filter 2: select stocks with daily average returns of all the quarterly/monthly periods in the last 'X' year > 0 .
- Filter 3: select stocks with a maximum limit of either 10/15 (depending on the strategy) on total non-positive returns count for all the quarterly/monthly periods.
- Filter 4: select stocks with maximum non-positive returns count for all the quarterly/monthly periods in the last 'X' year.
- Filter 5: select 50 stocks that have the least non-positive returns count for all the quarterly/monthly periods in the last 'X' year.
- Filter 6: select 30 stocks that have the highest daily average returns in the last 'X' year.

NOTE: The use of 'X' in some of these filters is related to the third component of the strategy which is 'Analysis Time Frames'. For now, you can consider them to be a number less than 3, which is the total years of data considered before the buying date.

Stock Selection Method	Holding/Rebalancing Period	Step 1 - Period Generation	Step 2 - Filters Applied
S1	1 quarter (3 months)	- generate 3-month periods - starting every 1st of a month in the last 3 years.	Filters 1-4 (Limit of 15 on Filter 3)
S2	1 month	- generate 1-month periods - starting every 1st, 11th, and 21st of a month in the last 3 years.	Filters 1-4 (Limit of 15 on Filter 3)
S3	1 quarter (3 months)	- generate 3-month periods - starting every 1 st of a month in the last 3 years.	Filters 1-4 (Limit of 10 on Filter 3)
S4	1 month	- generate 1-month periods - starting every 1st, 11th, and 21st of a month in the last 3 years.	Filters 1-4 (Limit of 10 on Filter 3)
S5	1 quarter (3 months)	- generate 3-month periods - starting every 1 st of a month in the last 3 years.	Filters 1, 2, and 4
S6	1 month	- generate 1-month periods - starting every 1st, 11th, and 21st of a month in the last 3 years.	Filters 1, 2, and 4
S7	1 quarter (3 months)	- generate 3-month periods - starting every 1 st of a month in the last 3 years.	Filters 5 and 6
S8	1 month	- generate 1-month periods - starting every 1st, 11th, and 21st of a month in the last 3 years.	Filters 5 and 6

Table 3.1: Stock Selection Methods

Methods S1 and S2: They have the first 4 filters in a sequential order with a limit of 15 on the third filter. These were the first two methods that I started the stock selection process with.

Methods S3 and S4: They have the first 4 filters in a sequential order with a limit of 10 on the third filter. These variations were created after noticing an impact on the results making the requirement of filter 3 more selective. Keeping the limit to 15 allowed many companies to get selected that had an extremely high variance, and could lead to losses.

Methods S5 and S6: They have the filter 1, 2, and 4, in the same order but do not have filter 3. After noticing the impact of making the process more selective in S3 and S4, it seemed like the logical move to remove the filter 3 requirement as it was focusing on the full 3-year performance and not the recent performance which was more important considering that the strategies were either for a monthly rebalance or a quarterly rebalance. Filter 3 with a limit of 10 selected very few companies most of the time (around 3 or 4) and that was not an adequate number to move forward with the weight allocation.

Method S7 and S8: After looking at the results from the previous methods, I got more and more clarity and insight on what filters are beneficial and how can the other filters be tweaked to get a good filter of stocks. S7 and S8 have only 2 filters which are filters 5 and 6 in that same order. Here, the focus had completely shifted to only looking at the more recent data and not the entire 3-year data which improved the performance a lot and some of the best outcomes of the project were through S7 and S8 which will be discussed later in the backtesting results and analysis section.

Step 3: After the stocks are filtered in step 2 through various possible methods, they go ahead for weight allocation.

3.1.2. Weight Allocation Methods

There are a total of 11 weight allocation methods defined for this project, coded from A1 to A11. A1 is the most basic weight allocation method, i.e., to equally weigh all the stocks. Methods A2, A3, A6, and A7 are based on the mean-variance analysis given by the Markowitz Portfolio Theory where A2 and A3 aim to maximise returns with slightly modified constraints, and A6 and A7 aim to minimise the risk by minimising variance with slightly modified

constraints. Method A4 and A5 are tangency portfolios that aim to maximise the risk-adjusted returns, i.e., the Sharpe ratio, with some variations in the constraints. Methods A8 to A11 are some of the more complicated and self-devised methods that aim to factor in the weightage given to stock for different objectives (returns, risk, and risk-adjusted returns) and try to pick stocks that have a significant weightage in all of those cases. Although the fundamental idea behind the methods A8 to A11 are the same, there are some differences in the constraints that will be discussed in this section. The weight allocation strategies are also based on the historical stock price data of a specific period ‘Y’, which is again related to the third component of the strategy which is ‘Analysis Time Frames’.

Method A1: Weight all stock selected equally

Method A2: If the number of stocks selected in the first component of the strategy is more than 8 then this method runs an optimisation function to maximise the returns with the constraints of not letting the annual standard deviation of the portfolio exceed 40% and while making sure that no single stock gets a weightage of more than 25% of the total funds, and then picks the 8 highest weighted stocks from the obtained portfolio and run the optimisation process again on those 8 stocks.

Method A3: Method A3 is the same as Method A2 with the only difference being more leniency on one of the constraints, i.e., no single stock should get a weightage of more than 40%, instead of 25% in A2.

Method A4: If the number of stocks selected in the first component of the strategy is more than 8 then this method runs an optimisation function to maximise the Sharpe ratio while making sure that no single stock gets a weightage of more than 25% of the total funds, and then picks the 8 highest weighted stocks from the obtained portfolio and run the optimisation process again on those 8 stocks.

Method A5: Method A5 is the same as Method A4 with the only difference being more leniency on the constraint, i.e., no single stock should get a weightage of more than 40%, instead of 25% in A4.

Method A6: If the number of stocks selected in the first component of the strategy is more than 8 then this method runs an optimisation function to minimise the variance with the constraints of having a minimum annual return of the portfolio of 30% while making sure that no single stock gets a weightage of more than 25% of the total funds, and then picks the 8 highest weighted stocks from the obtained portfolio and run the optimisation process again on those 8 stocks.

Method A7: Method A7 is the same as Method A6 with the only difference being more leniency on one of the constraints, i.e., no single stock should get a weightage of more than 40%, instead of 25% in A6.

Method A8: Method A8 applies a minimised variance optimisation on the selection stocks with no single stock being allocated more than 25% of the total fund and a constraint on the annual returns being a minimum of 20%. At the same time, it applies a maximised returns optimisation on the same stocks with no single stock being allocated more than 25% of the total fund and a constraint on the annual standard deviation being a maximum of 20%. From the two portfolios obtained, it picks the 4 highest weighted stocks from each to get a total of 8 stocks (4 giving a high return, and 4 giving a low risk). On these 8 stocks, the tangency portfolio maximises the Sharpe ratio, i.e., the highest risk-adjusted returns with a constraint on no single stock being more than 25% of the total fund.

Method A9: Method A9 is the same as Method A8 with the only difference being more leniency on one of the constraints, i.e., no single stock should get a weightage of more than 40%, instead of 25% everywhere in A8.

Method A10: Method A10 applies a minimised variance optimisation on the selection stocks with no single stock being allocated more than 40% of the total fund and a constraint on the annual returns being a minimum of 30%. At the same time, it applies a maximised returns optimisation on the same stocks with no single stock being allocated more than 40% of the total fund and a constraint on the annual standard deviation being a maximum of 30%, and simultaneously it also applies a maximised Sharpe ratio optimisation with the same constraint of no single stock being allocated more than 40% of the funds. From the three portfolios obtained, it picks all the stocks with a minimum of 10% weightage in all of them to get the best of all optimisations. After which it runs a maximised returns optimisation again on all of these stocks with constraints of no single stock being allocated more than 20% of the total fund and the maximum limit on the annual standard deviation being 30%.

Method A11: Method A11 is the same as Method A10 with the only difference being more leniency on one of the constraints, i.e., no single stock should get a weightage of more than 25%, instead of 40% everywhere in A10.

NOTE: The calculations and optimisation processes involving the Sharpe ratio use the Indian Government's bond yield historical data for deducing a risk-free rate of return which is used in the calculation of the Sharpe ratio.

Weight Allocation Method	Summary
A1	- equally weight all selected stocks
A2	<ul style="list-style-type: none"> - objective: maximise returns - constraints: <ul style="list-style-type: none"> 1. no stock > 25% 2. annual standard deviation < 40% - if selected stocks > 8, then choose the top 8 by weight and repeat the optimisation

A3	- same as A2 with the only change in one of the constraints being no stock > 40% instead of 25%		
A4	- objective: maximise Sharpe ratio - constraint: no stock > 25% - if selected stocks > 8, then choose the top 8 by weight and repeat the optimisation		
A5	- same as A4 with the only change in the constraint being no stock > 40% instead of 25%		
A6	- objective: minimise variance - constraints: no stock > 25%, annual returns > 30% - if selected stocks > 8, then choose the top 8 by weight and repeat the optimisation		
A7	- same as A6 with the only change in one of the constraints being no stock > 40% instead of 25%		
A8	- objective 1: minimise variance - constraints: no stock > 25%, annual returns > 20%	- objective 2: maximise returns - constraints: no stock > 25%, annual standard deviation < 20%	
	- pick the top 4 stocks by weight from the portfolios obtained to get 8 stocks and apply the optimisation for objective 3. - objective 3: maximise Sharpe ratio - constraint: no stock > 25%		
A9	- same as A8 with the only change in one of the constraints being no stock > 40% instead of 25% for all 3 objectives.		
A10	- objective 1: minimise variance - constraints: 1. no stock > 40% 2. annual returns > 30%	- objective 2: maximise returns - constraints: 1. no stock > 40% 2. annual standard deviation < 30%	- objective 3: maximise Sharpe ratio - constraint: no stock > 40%
	- pick all the stocks with a minimum weightage of 10% in all of the three portfolios and optimise for objective 4. - objective 4: maximise returns - constraints: 1. no stock > 20% 2. annual standard deviation < 30%		
A11	- same as A10 with the only change in one of the constraints being no stock > 25% instead of 40% for the first 3 objectives, constraints for objective 4 remain the same.		

Table 3.2. Weight Allocation Methods

3.1.3. Analysis Time Frames

Analysis Time Frames, within the context of stock selection and portfolio management, relate to the precise periods that are used to evaluate historical data in order to guide decisions for buying or selling securities. To make accurate data-backed judgements about the selection and weight allocation of securities for a portfolio, choosing the correct time span to analyse its performance and risk is highly important. Given the time-varying nature of financial markets, selecting the appropriate time range can have a big impact on how investment strategies perform.

Performance Evaluation: For the same asset, different periods may reveal different performance trends. While the long-term analysis may show stability and development potential, a short-term analysis may emphasise volatility and the possibility for rapid gains.

Risk Assessment: Over a variety of time periods, an asset's associated risk can change dramatically. While long-term analysis may show resilience and stability, short-term analysis may reveal more volatility.

Since my project involves two kinds of strategies, rebalancing every month and rebalancing every quarter, it does not require a huge analysis period. While some traditional methods suggest that the analysis period should be at least 10 times longer than the holding period of the portfolio, some suggest 5 times, and some suggest 3 times. Which is why I decided to experiment with different analysis time frames. It is important to note that as described above, the stock selection method is a completely independent process to the weight allocation method except for the fact that which stocks to allocate weight to depends on which stock got selected. Apart from that, the time frames for both of these processes need not necessarily be the same and can be different. These are the periods referred to as 'X' and 'Y' in the above sections. As

a base case, I started with an analysis time frame of 1 year each (for stock selection and weight allocation), and then I tried both, increasing the period, and decreasing the period to compare the results. The results, which will be discussed in much more detail in later sections, were much better for smaller periods, which was expected since the holding periods of both types of strategies are monthly and quarterly, which aren't huge periods. I also experimented with taking larger time frames for the stock selection methods and smaller ones for the weight allocation methods and vice-versa. Considering the results, I went ahead with decreasing the time frames more and eventually reached 10 different combinations of periods for stock selection and weight allocation.

Analysis Time Frame Combination	Historical Time Period Considered for Stock Selection Methods	Historical Time Period Considered for Weight Allocation Methods
T1	1 year	1 year
T2	1 year	2 years
T3	2 years	1 year
T4	2 years	2 years
T5	6 months (0.5 year)	6 months (0.5 year)
T6	1 year	6 months (0.5 year)
T7	6 months (0.5 year)	1 year
T8	3 months (0.25 year)	3 months (0.25 year)
T9	6 months (0.5 year)	3 months (0.25 year)
T10	3 months (0.25 year)	6 months (0.5 year)

Table 3.3. Analysis Time Frames

Method T1:

Stock Selection Historical Time Frame(X): 1 year

Weight Allocation Historical Time Frame (Y): 1 year

Method T2:

Stock Selection Historical Time Frame(X): 1 year

Weight Allocation Historical Time Frame (Y): 2 years

Method T3:

Stock Selection Historical Time Frame(X): 2 years

Weight Allocation Historical Time Frame (Y): 1 year

Method T4:

Stock Selection Historical Time Frame(X): 2 years

Weight Allocation Historical Time Frame (Y): 2 years

Method T5:

Stock Selection Historical Time Frame(X): 6 months (0.5 years)

Weight Allocation Historical Time Frame (Y): 6 months (0.5 years)

Method T6:

Stock Selection Historical Time Frame(X): 1 year

Weight Allocation Historical Time Frame (Y): 6 months (0.5 years)

Method T7:

Stock Selection Historical Time Frame(X): 6 months (0.5 years)

Weight Allocation Historical Time Frame (Y): 1 year

Method T8:

Stock Selection Historical Time Frame(X): 3 months (0.25 years)

Weight Allocation Historical Time Frame (Y): 3 months (0.25 years)

Method T9:

Stock Selection Historical Time Frame(X): 6 months (0.5 years)

Weight Allocation Historical Time Frame (Y): 3 months (0.25 years)

Method T10:

Stock Selection Historical Time Frame(X): 3 months (0.25 years)

Weight Allocation Historical Time Frame (Y): 6 months (0.5 years)

3.2. Strategy Combinations and Evaluation Process

Having devised 8 stock selection methods, 11 weight allocation methods, and 10 analysis time frame combinations, I thought of trying every combination of all of them to come up with the best possible strategy out of all of them that consider so many different parameters. This gave me a huge total of 880 unique strategies for every combination of S1 to S8, A1 to A11, and T1 to T10.

For easy referencing, I decided to give a unique code to each of the 880 strategies that are a combination of their stock selection method, weight allocation method, and analysis time frame combination, which is given in the table on the next page.

Sr. No.	Stock Selection Method	Weight Allocation Method	Analysis Time Frame	Combination Code
1	S1	A1	T1	S1-A1-T1
2	S1	A1	T2	S1-A1-T2
...
110	S1	A11	T10	S1-A11-T10
...
880	S8	A11	T10	S8-A11-T10

Table 3.4. Strategy Combinations

To evaluate these strategies I decided to backtest all 880 strategies for 4 years from 1st January 2020 to 1st January 2024. Based on the results of the evaluation I had 4 different selection criteria defined to find the best of them. These 4 criteria were:

1. Strategy with the highest 4Y CAGR among the ones with the highest success rate
2. Strategy with the lowest 4Y variance among the ones with the highest success rate
3. Strategy with the highest 4Y CAGR
4. Strategy with the lowest 4Y variance

Another important component of the evaluation process is deciding the backtesting dates. The backtesting dates would vary depending on the strategy's rebalancing rule/holding period. As described earlier in the 'Stock Selection Methods' section, for the strategies with quarterly rebalancing, the backtesting would start with buying the first portfolio on the 1st of January 2020 and rebalancing it on the 1st of every month

3.3. Tools and Technologies Used

For this project, a variety of tools and techniques were used. The foundation of the project is laid on Python programming language in addition to many in-built and external libraries and modules.

The external libraries and modules include:

- **Pandas:** Pandas library is used for data collection, wrangling, and analysis. Pandas help in creating and managing data in the form of data frames.
- **Numpy:** Numpy is used for large scientific calculations including calculations on multi-dimensional arrays and matrices along with a huge collection of high-level mathematical functions that can be used. Using Numpy for the project helped with reducing time and space complexity and optimising the computing processes involved in the project.
- **Yfinance:** Yfinance is an open-source API by Yahoo Finance that was the primary source for all the data needed for the project. The YFinance API enables Python programmers to access both live and stock data including opening and closing prices and volumes for analysis and strategy creation.
- **Matplotlib:** Matplotlib is a Python charting toolkit that is used to create static, interactive, and animated visuals. It has features for generating many different plots and charts, including scatter plots, bar charts, line charts, and histograms. Matplotlib played a key role in this project's data and result visualisation, which helped to clarify patterns, trends, and anomalies in the dataset.
- **Seaborn:** Seaborn is a high-level interface for creating visually appealing and educational statistical visualisations offered by this Python visualisation toolkit, which is built on top

of Matplotlib. Making intricate graphs from data in arrays and data frames is one of its main uses. Seaborn was utilised in this project to produce more complex statistics charts and to improve the plots' aesthetic appeal.

- **Scipy:** The library Scipy is used for scientific calculations in Python. It contains features for optimisation, linear algebra, integration, differential equation solvers, etc., which are common in engineering and operations research-related projects. In this project, Scipy was used within various weight allocation methods to cater for various objectives such as maximising returns, maximising the Sharpe ratio, and minimising risk.
- **Python-dateutil:** Python-dateutil is an extension to the standard in-built datetime module in Python. It offers a wide range of operations on dates in various formats, which is particularly useful in a project like this where the entire strategy is built on time-series data analysis.
- **Streamlit:** Streamlit is a framework available in Python that helps create minimalistic and interactive web applications for Python programmers without having to use any frontend web development programming languages like HTML, CSS, or JavaScript. It enables programmers to create interactive apps in Python itself and in this project, it was used to build an interface that allows users to explore the strategies and get live recommendations for their portfolios.

The in-built libraries and modules include:

- **DateTime:** Functions for modifying dates and times are provided by the datetime module. To properly handle time-series data, it was utilised in this project to work with dates along with the extension Python-dateutil.

- **Sys:** The sys module gives access to functions that have a close relationship with the Python interpreter as well as some variables that are utilised or maintained by the interpreter. It was applied to system-specific functions and parameters in this project, like changing the directory of the Python interpreter to access data files in an external directory.
- **OS:** The OS module gives users access to functionality that depends on their operating system. It was utilised in this project to handle files and directories by interfacing with the operating system along with the sys module.
- **Random:** For a variety of distributions, the random module applies pseudo-random number generators. It was employed to produce random numbers, which help assign random values when necessary or for use in simulations.
- **Math:** The mathematical functions specified by the C standard are accessible through the math module. It was used for constants and elementary mathematical operations that need not be defined repeatedly in the code.
- **Ast:** The ast module facilitates the processing of Python abstract syntax grammar trees by Python applications. It is not frequently utilised in data projects, but when it is, it can be quite helpful for the project's dynamic expression evaluation, like the conversion of strings from a data file to dictionaries.
- **JSON:** Data encoding and decoding in JSON is made simple by the JSON module. This project utilised it to parse JSON files or data, which is frequently found in web-based data sources. This project also used the JSON library to save backtesting results and read them back for evaluation and analysis.

- **Statistics:** Functions for computing mathematical statistics of numerical data are available in the statistics module. This project utilises it to do fundamental statistical operations like variance, median, and mean calculations.

3.4. Code Documentation

- `'streamlit_app.py'`

Purpose: It acts as the point of entry for the Streamlit-built user interface. Through a web-based application, users may interact with the system, select investing strategies, see suggestions for their portfolios, and review backtesting results.

Interactions: It exchanges data with other modules to obtain performance indicators, portfolio suggestions, and strategy choices. The user input that is gathered here is sent to other components for handling.

- `'recommendations.py'`

Purpose: To provide individualised investing recommendations, it combines the techniques of weight allocation and stock selection. This module suggests particular portfolios by using the selected approach and the most recent market data.

Interactions: To assemble and offer investment recommendations, it coordinates the flow between stock selection methods and weight allocation methods through the `functions.py` module. It interacts with `streamlit_app.py` to show the user these suggestions.

- *'functions.py'*

Purpose: This utility module serves as the project's central repository for common functions. These consist of pre-processing data, financial computations, and support functions for validating and manipulating data.

Interactions: This module ensures code reuse and maintainability by being heavily relied upon by other system components to carry out repetitive tasks. It receives function calls from the *recommendations.py* file and calls functions from the *weight_allocation_strategies.py* for weight allocation tasks after the stocks are selected.

- *'weight_allocation_strategies.py'*

Purpose: It implements the defined logic for the various weight allocation methods (A1 to A11). The functions in this module decide how the investment amount is split across the various selected stocks.

Interactions: It works closely with *optimization_functions.py* to get the outputs of various optimisation techniques that the defined weight allocation methods rely on. Outputs from this module are directly used to generate portfolio recommendations.

- *'optimization_functions.py'*

Purpose: It is focused on work related to optimisation functions for weight allocation. It is used to find the best stock weightings based on predetermined criteria, such as maximising returns, minimising variance, or maximising the Sharpe ratio, and a variety of constraints. It applies a variety of mathematical models and optimisation methods.

Interactions: The functions defined in this file are used by functions within the *weight_allocation_strategies.py* file whenever required as per the predetermined methods including objectives and constraints.

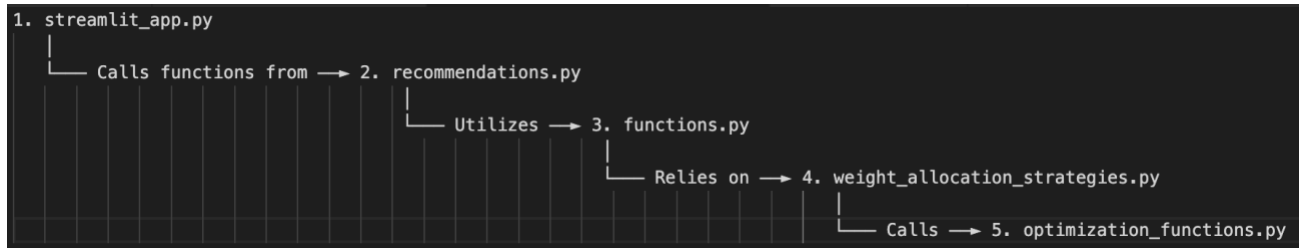


Figure 3.1: Flow of the code

Chapter 4: Evaluation and Results

4.1. Purpose of Evaluation

The evaluation process and its analysis have been designed in a manner to guarantee the accomplishment of several goals, which include a thorough evaluation of the investment strategies produced during the project. The prime ambition of the review is to very rigorously confirm the effectiveness of the strategies on past market data, which itself is very crucial requisite to ascertain the robustness and advantage over conventional market benchmarks. This validation is a test that the decision adopted by us is confirmed as workable under actual market conditions and theoretically sounds.

Performance Evaluation: This is an assessment that involves a critical evaluation of the performance of each strategy in the chosen period. For that, examination of their effectiveness in terms of capital appreciation through varied performance criteria like total return, compound annual growth rate (CAGR), etc. The perception I hope to get from this assessment is the potential behaviour of the strategies under different market conditions, which provides an empirical way for a realistic outlook toward their usefulness and suitability.

Insight into Risk Management: The understanding of the risk-return profile of any strategy ranks as the second-most important aspect of the evaluation process. What I mean is that with these, I will get to know more about how promising strategies can manage risks by how they respond to volatility in markets. This helps identify the kind of strategies which would give attractive returns but at the same time manage risks effectively, such that the invested capital is not exposed to undue market fluctuations. It is such types of strategies that maintain a balanced portfolio in line with the various risk tolerances of investors.

Validation and Improvement: The assessment process will be critical to validate the strategic decisions made during the development process of these investment strategies, performance evaluation, and risk. This helps one to be able to note areas where strategies are working and be improved by comparing theoretical expectations with actual historical results. This kind of process itself continually enhances and validates the strategies to be both flexible and adaptive to the shifting market conditions and changing demands of the investors.

4.2. Backtesting Process

It is a very critical technique in the process of developing financial strategies, and through it, a strategy will have to be put into simulation concerning how it would have performed, using historical data. It is something of a time machine that allows traders and investors to see how their strategies would fare back in the past. Through this, one can glean valuable insight into the historical performance of a strategy and further gauge its feasibility and potential effectiveness within the real world of trading or investing. The following constitute the main components of backtesting:

History Performance Analysis: This involves the assessment of the historical performance of a strategy in varied market conditions. It is very important to understand how the strategy behaves in different economic cycles, such as between bull and bear markets and from periods of high volatility to those of low volatility.

Comprehensive Testing: The reliability of backtesting is pinned on the choice of dates, timespan, and the specific metrics used for strategy evaluation. When such a wide range of dates are taken into consideration, then this means that testing of the strategy cuts across such a wide range of market conditions.

Metrics: The performance objective metrics related to the success of the strategy comprise compound annual growth rate (CAGR) over different periods (e.g., 4 years, 1 year), variance in returns, and other connected metrics in which the strategy performs. The metrics put forward the multi-dimensional view of the effectiveness of the strategy, which includes aspects of profitability, risk, and consistency.

Backtesting plays an indispensable role with reference to the formulated investment strategies within the project. It will evaluate the devised investment strategies in the following way:

Simulation Details: For each developed strategy, the process of investment is designed by backtesting for the designated iterations. Beginning each iteration with selling the existing holdings, and buying a new set of recommendations every time using the proceeds from the sale. The backtesting was initiated with a hypothetical amount of investment of Rs. 1,00,000 and permitted its tracking over successive periods of rebalancing. Additionally, I incorporated rebalancing costs after every iteration before the amount gets reinvested to get a more realistic outlook. The rebalancing costs include brokerage, taxes, and other such charges.

Dates and Timespan: This research's backtesting is done with historical market data from 1st January 2020 through 1st January 2024. This way, it encapsulates diverse market conditions within the four years and thus gives quite a comprehensive view of how the strategies would have performed.

Rebalancing Strategies: For monthly rebalancing strategies, 48 iterations of buying and selling of the portfolio have been done. Equivalently, for quarterly rebalancing strategies, 16 iterations corresponded to the number of quarters in the 4 years.

4.3 Limitation: Survivorship Bias

One very important concept to be considered in my project is the idea of survivorship bias, where many of its implications are laid in the general and, above all, in the financial realm. Survivorship bias is a cognitive or analytical mistake that focuses only on those entities or elements that 'survived' or went through a particular process of selection while ignoring those that did not. Too much optimism may result in this bias, where successful investments are put into full view, and examples of lessons or warnings that should be considered are neglected or given too little weight.

General understanding of survivorship bias: This bias may be included in various other fields such as research, business analytics, and historical studies, influencing the respective outcome. For instance, analysing only the successful firms from which to draw out the factors of business success would be potentially dangerous and would yield perverse results, as it fails to tap valuable learning from the study of failure. This bias covers the view of the whole scope, pointing out that all relevant data points need to be considered in such a way that misleading conclusions are not formed.

Survivorship bias in financial context: In the financial sector, survivorship bias is particularly pertinent when analysing investment performances. This happens if the analysis takes into consideration only the active stocks that are currently participating in the market (those are the stocks of the companies that did not exit the market for any causes, including bankruptcy, merger, etc.). The tendency of selectively analysed data will give a misleading result, namely to overestimate the expected average returns and underestimate the potential risks, therefore distorting the real market or efficacy of the investment strategy.

Survivorship bias in this project: Survivorship bias plays a vital role in my project due to the way the evaluation is designed. More so due to not including the historical constituents of the base universe in the analysis. The base universe of 750 stocks are the constituents of the NIFTY TOTAL MARKET index. This index is reviewed semi-annually, meaning that twice a year, some stocks are removed from that list of 750 while others are added. The removal and addition process is based on a selection criteria devised by the National Stock Exchange (NSE) that manages the index. When certain stocks get removed from the list, they essentially fail to pass through the selection process, leaving only those stocks which got selected. Using the current members of the index for backtesting would lead to skewed results as the constituents of that index in 2024 would not be the same as in 2022, but when we provide the list of those “winner” stocks (which is the current list) to the strategy when backtesting in 2022, it would be only looking at the stocks that are going to be better in the future, hence creating a huge bias. In simple words, it is somewhat similar to knowing which stocks are going to perform well in the future and only choosing a portfolio from those stocks, which is impossible to execute in real life.

Hence, a strategy must be survivorship-bias-free through access to the data of the historical constituents of the index for a completely fair backtesting process. This would help reflect on cases where the strategy could have picked “loser” stocks that are going to be delisted, bankrupt, or just underperform substantially in the future, potentially allowing me to identify the causes of those stock picks and try and get rid of them to improve to strategy.

The primary challenge that led to this kind of bias was access to the historical constituents on the index and the details of the additions and removals to the index. Detailed records of the historical members of market indices are available, but only from specialised financial databases. However, these databases are often not available due to high costs or supplementary

with more data, which are critical in ensuring a rigorous analysis of inclusions and exclusions of companies from the index over time. The presence of this survivorship bias in the project underscores an important limitation in the analysis. Therefore, the strategies developed might not work as well in the present day as they worked in the past according to the backtesting, where historical constituents' inclusion would change risk assessments and potential return estimations. This gives a very important pedestal for dealing with and mitigating the effect of survivorship bias on financial analysis and strategies, hence putting even more emphasis on how the effects can be reduced for better robustness and general application.

4.4. Analysis of Results

Using the results of backtesting, I carefully designed an analysis approach that helps surface the best investment strategies that match varied investor risk profiles. For this, I use a variety of metrics and parameters. This comprehensive data analysis focuses on performance metrics and strategic parameters to understand to what extent any given strategy is superior. The sections below will synthesise metrics used, parameters considered, selected strategies, live-testing stage, and key observations that have emerged as optimal across different dimensions of performance and risk management.

Metrics:

- CAGR (4Y, 3Y, 2Y, 1Y): These metrics calculate the compound annual growth rate over four distinct timeframes: 4-year, 3-year, 2-year, and 1-year periods. Each provides insights into the strategy's growth potential and performance consistency over time.
- Variance of Returns (4Y, 3Y, 2Y, 1Y): This set of metrics measures the variance in returns for different periods (4-year, 3-year, 2-year, 1-year) within the strategies. The variance is calculated

based on the returns achieved in each period, be it monthly or quarterly, depending on the specific rebalancing strategy. A lower variance indicates more consistent performance, while a higher variance points to greater risk due to fluctuating returns.

- **Success Rate:** The success rate is determined by the percentage of periods that yielded a positive return upon rebalancing, out of the total number of periods tested during the 4-year backtesting phase. For strategies with monthly rebalancing, this is calculated over 48 periods (months), and for quarterly rebalancing strategies, over 16 periods (quarters). A higher success rate indicates a strategy's reliability in generating positive outcomes over its rebalancing intervals.

Parameters Considered:

- Stock Selection Methods (SSM)
- Weight Allocation Methods (WAM)
- Analysis Time-Frame Combinations (ATFC)
- Rebalancing Periods (RP)

All the parameters are compared using the metrics. For example, all the stock selection methods from S1 to S8 are compared based on the average of the CAGR of all the strategies where a particular stock selection method was used.

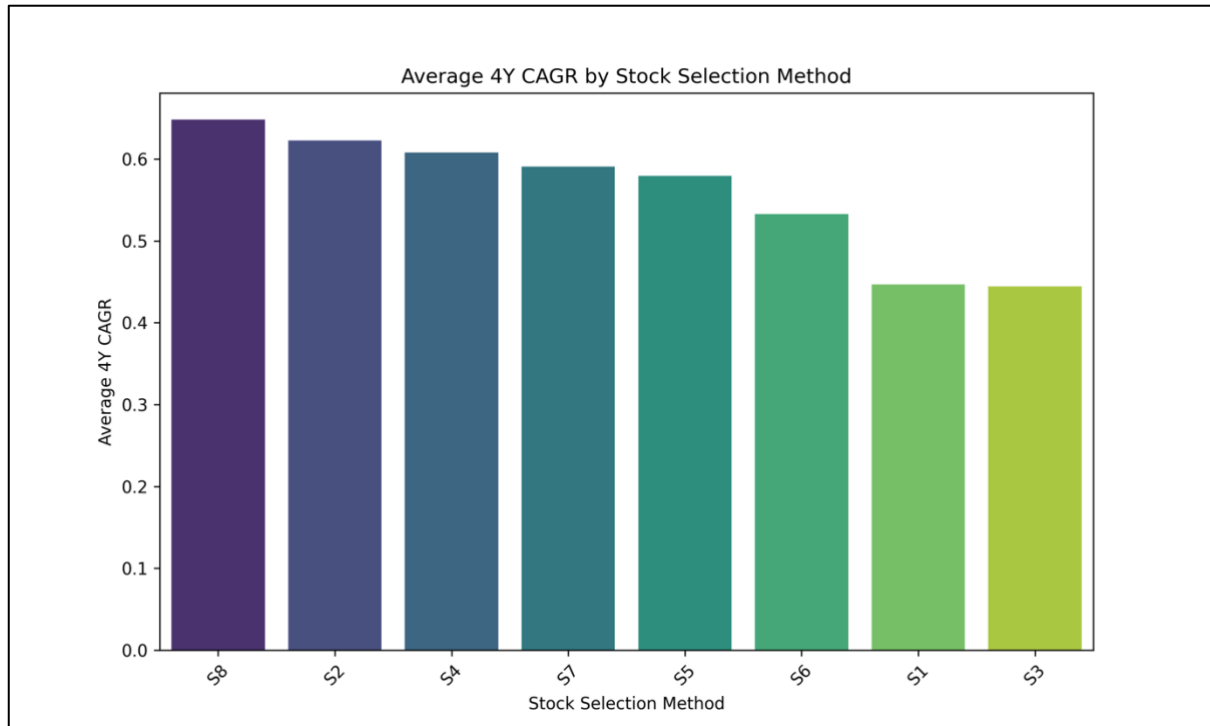


Figure 4.1: Average 4Y CAGR by Stock Selection Method

Similarly, the analysis of every combination of the metrics and the parameters has given some significant insights into the strategies devised during the project. All the other graphs used for analysis are provided in the appendix of the report.

Key Observations and Insights:

- The quarterly rebalancing of strategies often outperformed others in finding a balance between capturing market gains and minimising risk with periodic adjustments.
- The stock selection methods with fewer filters gave strategies that were more aggressive, taking long shots for the higher return but at the expense of the higher risk.
- The strategies with more filters in their stock selection method were important in protecting the portfolios from huge losses and enhancing the risk-adjusted returns.

- Lower limits on maximum funds per stock resulted in enhanced risk-adjusted performance that showed diversification is key.
- The objective of the weight allocation methods is minimising the variance correlated with a higher success rate, suggesting that stability often trumps volatility for long-term success.
- It is also observed that shorter time frames expose greater risk, while larger or diverse time frames in analysis make the strategies more flexible to the different market dynamics.
- The monthly rebalancing of strategies usually capitalises on short-term market moves but also introduces a higher risk because it forces more frequent adjustments in the portfolio.
- Higher allocation to high-performance stocks could amplify the returns but will also magnify the portfolio volatility.

4.5. Strategy Selection

Out of the 880 unique strategies formed by the combinations of the stock selection methods, weight allocation methods, and the analysis time-frame periods, 12 emerged as the best strategies on different parameters. This strategy selection is very specific, denoted by predefined criteria of characteristics based on success rates, compound annual growth rates (CAGR), and variance of returns. Herein, I provide an overview of these strategies and the rationale behind their selection:

Strategy 1 (S7-A7-T5): This is the strategy with the highest success rate of 88% over the backtest period, denoting long-term stability.

Strategy 2 (S4-A3-T10): This is the strategy with the highest 4-year CAGR of 154% over the backtested period, showing exemplary performance but at the cost of a higher risk.

Strategy 3 (S5-A6-T5): This is the strategy with the lowest 4-year variance in returns of 1.52% over the backtested period, proving to be the safest strategy of all, along with a trade-off in returns.

Strategy 4 (S1-A6-T7) & Strategy 5 (S4-A7-T5): These strategies turn out to be the best two strategies based on the lowest CAGR differential among the CAGR of different periods (4Y, 3Y, 2Y, and 1Y), showcasing consistent growth.

Strategy 6 (S5-A6-T7), & Strategy 7 (S8-A1-T4): These strategies turn out to be the best two strategies based on the lowest differential in the variance of returns among the variance of returns of different periods (4Y, 3Y, 2Y, and 1Y), showcasing consistent stability.

Strategy 8 (S3-A11-T3) & Strategy 9 (S3-A10-T3): These strategies turn out to be the best two strategies based on the highest average CAGR among the CAGR of different periods (4Y, 3Y, 2Y, and 1Y), showcasing exceptional growth potential and robust performance consistency.

Strategy 10 (S6-A1-T8), Strategy 11 (S3-A7-T5), & Strategy 12 (S1-A6-T9): These strategies turn out to be the best three strategies based on the lowest average variance of returns among the variance of returns of different time-periods (4Y, 3Y, 2Y, and 1Y), showcasing superior risk management and stability across various market conditions.

A table displaying some important statistics of these strategies is given below:

Sr. No.	Strategy Code	4Y CAGR	4Y Variance	Success Rate	Rebalancing Period
1	S7-A7-T5	76%	4.52%	88%	3m
2	S4-A3-T10	154%	17.96%	71%	1m
3	S5-A6-T5	28%	1.52%	75%	3m
4	S1-A6-T7	44%	2.36%	75%	3m
5	S4-A7-T5	58%	3.33%	60%	1m
6	S5-A6-T7	31%	1.57%	75%	3m
7	S8-A1-T4	47%	2.34%	71%	1m
8	S3-A11-T3	84%	4.08%	81%	3m
9	S3-A10-T3	87%	3.93%	81%	3m
10	S6-A1-T8	39%	2.53%	71%	1m
11	S3-A7-T5	41%	1.66%	75%	3m
12	S1-A6-T9	38%	1.58%	75%	3m

Table 4.1: 12 Selected Strategies

As discussed earlier, the base universe on which all of these strategies are built is the collection of 750 stocks that are currently a part of the NIFTY TOTAL MARKET index. As mentioned in the survivorship bias section of the report, this index is reviewed semi-annually, resulting in a bias in the results. Regardless, it does not hamper the comparative analysis and selection of the strategies. The index constituents change at the end of March, and at the end of September, which means that the list of stocks being used for the strategy development was the one reviewed at the end of September 2023. To try and get rid of some strategies that have shown more of an exceptional performance due to the bias, I decided to live-test these 12 strategies from 1st October 2023 to 28th December 2023, for every possible portfolio they recommend on all of the 59 days when the market was open. The reason to live-test it only till 28th December 2023 is that a quarterly rebalancing strategy would take 3 months to give a result to analyse, and today being the 28th March 2024, I couldn't live-test it beyond 28th December 2023. This supplementary scrutiny was carried out to assess the level of real-world applicability and resilience of the strategies beyond the theoretical constructs with an exclusive concentration on how they perform in the face of market realities, and to eliminate some more strategies from the 12 selected to give the investor a choice of choosing a strategy that suits their risk-profile while making sure that a large number of options does not confuse them.

When this was done, it was found that 7 out of those 12 strategies gave a success rate of 90% or above. What is meant here by success rate is the percentage of portfolios out of 59 (from October 2023 to December 2023) that gave a positive return, which essentially means that 7 out of the 12 selected strategies gave 54 such recommended portfolios out of 59 which gave a positive return. Out of those 7, 5 strategies were such that the average monthly returns of their 59 portfolios surpassed the average monthly return of NIFTY 50 (our benchmark index) for the same dates. These 5 strategies became the final choice as the best output of this project.

A table showing the average monthly returns and success rates of the 5 selected strategies is given below:

Sr. No.	Strategy Code	Average Monthly Returns	Success Rate
1	S7-A7-T5	6.39%	96.61%
2	S5-A6-T5	5.03%	100%
3	S5-A6-T7	5.03%	100%
4	S3-A11-T3	6.12%	100%
5	S3-A10-T3	6.98%	100%
	NIFTY 50	3.43%	

Table 4.2: Average Monthly Returns and Success Rates of the 5 Selected Strategies for the Live-Testing Evaluation from Oct 2023 to Dec 2023

It can be observed that although strategy S7-A7-T5 was the one that gave the highest success rate over 4 years of back-testing, a strategy that turned out to be better than that, strategy S3-A10-T3, has a higher success rate and a higher average monthly return (double of NIFTY 50's average monthly return) when looking at the live-testing results of these 4 months. While the other strategies are also comparable and have performed well, it is difficult to find the truly best strategy out of all of these without getting access to historical constituents of the NIFTY TOTAL MARKET index and re-simulating the backtesting process of those 4 years, or even a larger period than that to get a truly unbiased bigger picture of the performances of these strategies. Although it is safe to assume that at least these 5 strategies have been performing

great in the short-term and can be relied upon as far as a real investment is considered, also making sure that the future work

A density distribution plot of the monthly returns of the 59 portfolios generated by these 5 strategies is given below:

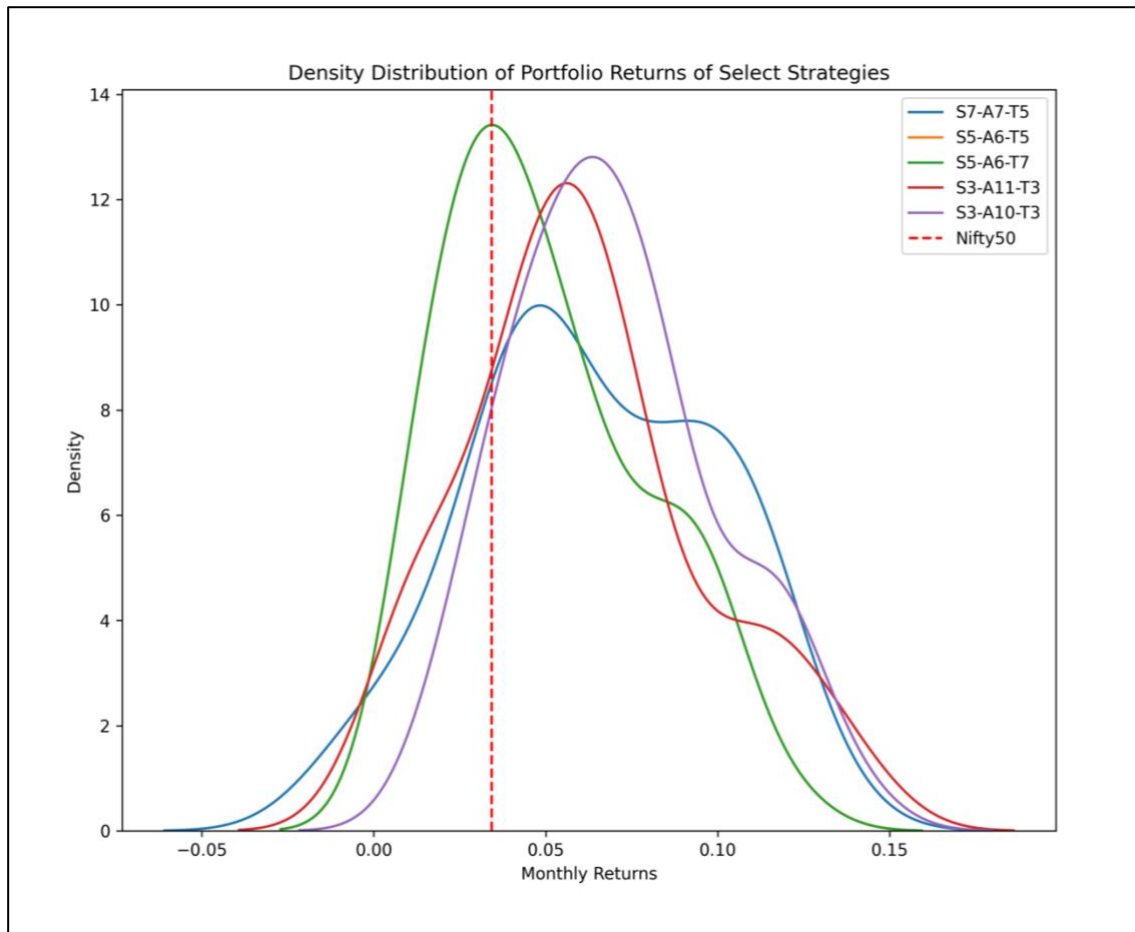


Figure 4.2: Density Distribution of Monthly Portfolio Returns of the 5 Selected Strategies for the Live-Testing Evaluation from Oct 2023 to Dec 2023

This selection shows a deep commitment to the use of historical data, empirical evidence, and performance efficacy, for integrating the investment strategies into the application, making sure that the investor uses a validated, accurate, and robust strategy.

The performance of these selected strategies for the entire duration of backtesting (from January 2020 to January 2024) can be observed in the graph below:

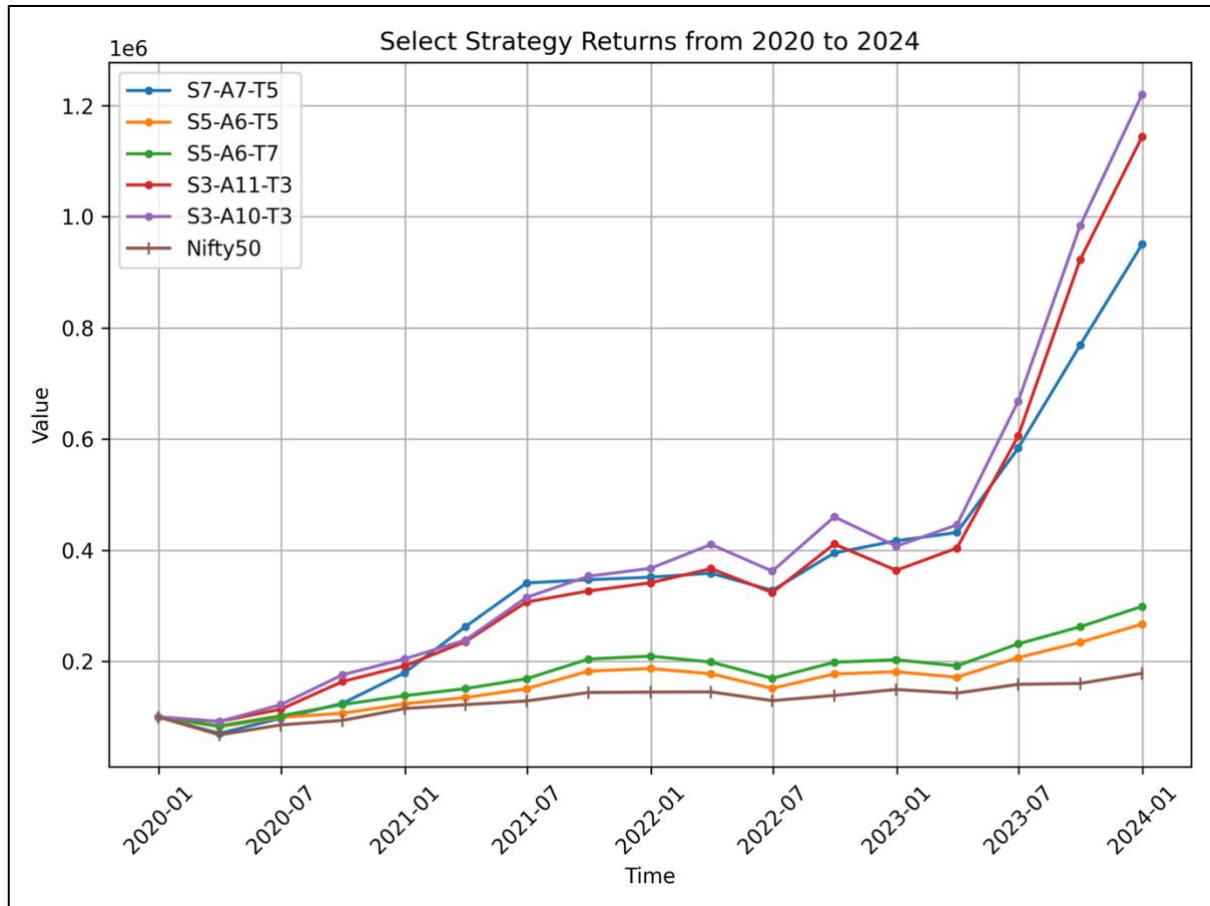


Figure 4.3: Comparative Performance of Selected Investment Strategies Versus Nifty 50 from January 2020 to January 2024

4.6. Position Sizing

Position sizing within an investment or trading strategy means the process through which the decision of the number of shares to buy or sell within the trade has been made. Position sizing is an important tool for risk management, as well as to assess the historical performance of the strategies. Most theories have an implied assumption that all stocks are infinitely divisible and any amount of money can be invested in them, which is not the case when the real market is considered. For this very reason, it becomes important to determine the exact number of shares that could be purchased of a given security, which is based on the results of the weight allocation method, to get an even more refined outlook on the performance of these strategies. To give an example, an investor could have an available fund of Rs.10,000. The strategy the investor chose might suggest they buy a portfolio of stock X (70%), stock Y (20%), and stock Z (10%). Now assume that 1 share of this stock Z is worth Rs.1,500, but 10% of the total fund available is only Rs.1,000. In this scenario, it is impossible to invest in company Z without increasing the total fund or changing the proportions of the fund distribution. To handle such situations I devised a position-sizing algorithm described below:

Determining Amount Per Stock: First, the function that calculates what amount of the total investment should go to the purchase of each stock in relation to its proportion suggested by the weight allocation method of the chosen strategy.

Checking Stock Prices: Next, it checks the current market price or the last traded price for every stock. This helps in knowing at what price level you can make purchases of the stocks.

Ensuring Affordability: Some stocks are so expensive that they might just be above what you can afford. The aim is to decide those that you can afford and possibly those you need to drop.

This is done by finding out the ratio of “Amount Per Stock” and “Last Traded Price”. If this ratio is below 1 for a particular security, we omit that from the portfolio.

Adjust Proportions: Next, with the affordable stocks filtered out, the algorithm recomputes how the investment is to be distributed between the remaining stocks and the filter. This is important to maintain the relative importance of each stock with respect to others in the portfolio and leave the proportional investment in line with the originally formulated strategy recommendation.

Calculating Quantities: Then we move on to calculate the number of shares that one can buy. This is a crucial step where the amounts derived in steps 1 and 2 are used to calculate the number of shares that could be bought for each security. However, since shares could only be in whole numbers, the values are always rounded down to ensure that we do not exceed the limited funds that the investor is willing to invest.

The gist of it is that this algorithm aids not just in the sensible purchase of shares according to planned proportions of investment but also ensures that when some stock is overpriced, the rest of the portfolio will adjust to stay in tune with your financial strategy and limitations. After the development of this algorithm, I tried experimenting with the investment amounts to see if it makes a difference, and if it does, then by how much, and what is the correct amount beyond which increasing the amount would not improve the performance by much. I tried simulating the live-tested 59 portfolios with position sizing with starting amounts Rs.1,000, Rs.10,000, Rs.1,00,000, and Rs.10,00,000.

The results show that the same portfolios if position-sized considering an initial investment amount of Rs.1,000 or Rs.10,000, gave a very poor performance, although relatively between

the two, the higher amount displayed a better performance. Similarly, when moving from Rs.10,000 to Rs.1,00,000, the performance improved significantly, which did not change much even after increasing the amount to Rs.10,00,000. Hence it would be safe to conclude that an amount around Rs.1,00,000 is more than sufficient to reach the maximum potential of the strategies devised during this project.

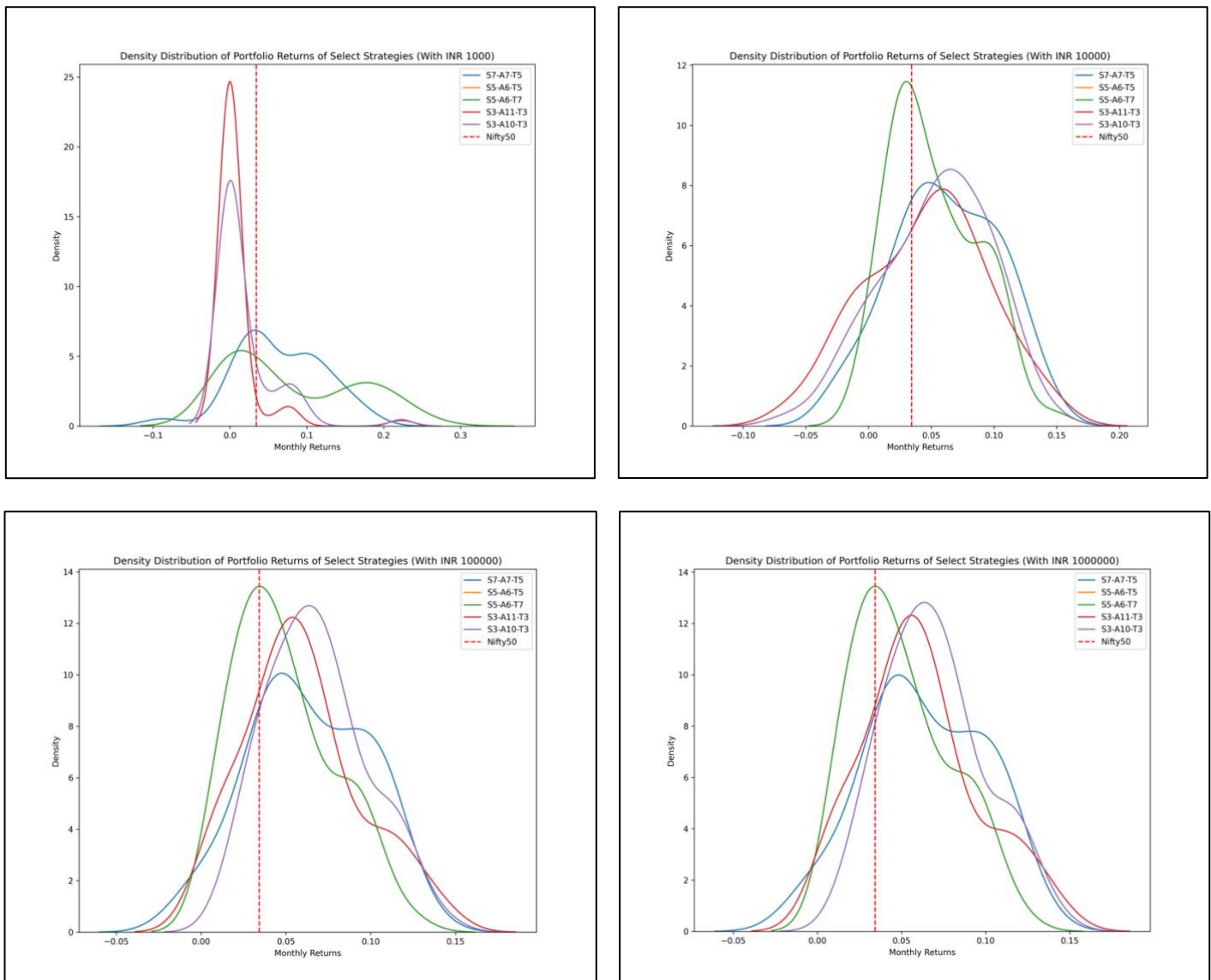


Figure 4.4: Density Distribution of Position Sized Monthly Portfolio Returns of the 5 Selected Strategies for the Live-Testing Evaluation from Oct 2023 to Dec 2023 with Investment Amounts being Rs.1,000, Rs.10,000, Rs.1,00,000, and Rs.10,00,000 respectively

It can be observed that as we moved from smaller amounts to bigger amounts, there is a shift in the curves towards the right, showing an increase in average monthly returns, but the shift of curves stops at the amount of Rs.1,00,000 and no further shift is observed with an amount of Rs.10,00,000.

As observed an investment amount of Rs.1,00,000 was sufficient to meet the potential of the strategies. I applied that to the portfolios that were suggested during the backtesting as well to see if it made a difference over time, the results of which are given below:

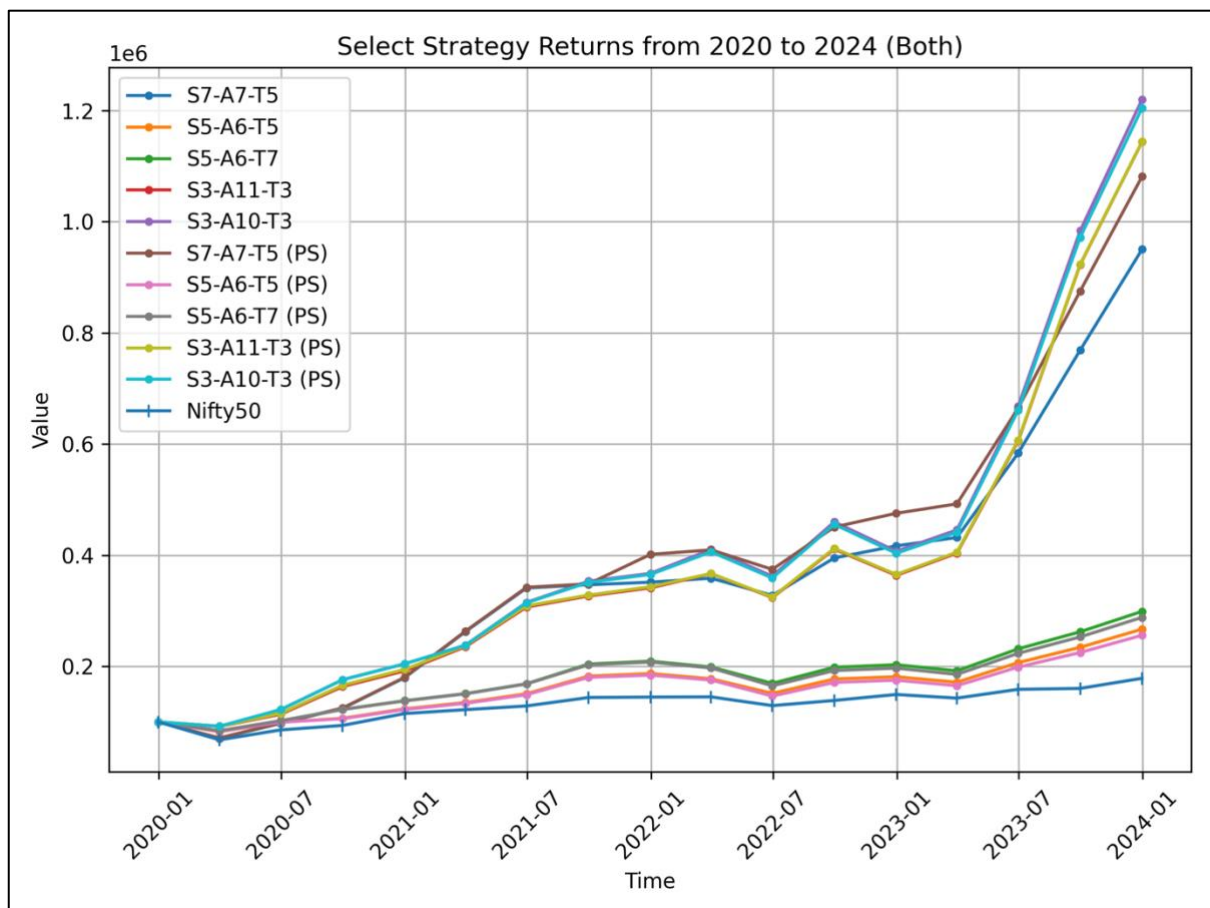


Figure 4.5: Selected Strategy Performance vs. Nifty 50 (2020-2024) - Normal and Rs.1,00,000 Position Size

It can be observed that most of the strategies did not show much of a deviation when backtested in a way where stocks are considered to be infinitely divisible versus when backtested with a position size of Rs.1,00,000. Some show a slight positive deviation, while some show a slightly negative deviation. Although, on average, the 5 strategies together showed a positive deviation of 0.95%, almost 1% between the two different cases.

This reaffirms our findings of the live-testing that if the initial investment amount is around as large as Rs.1,00,000, it is sufficient to utilise the maximum potential of the recommendation of the strategy and there won't be any significant improvement upon increasing the amount even more.

Chapter 5: Web Application Development

The purpose of the web application for this project is primarily as a prototype for this academic project and to showcase a use case for devising such portfolio management and time-based rebalancing strategies, bridging the gap between theoretical investing and practical investing. The idea was to make the application fairly straightforward despite the complex nature of the backend of the project itself. The development of the application was driven by the purpose of empowering individual investors by giving them a platform that not only gives them investment recommendations according to their risk profiles but also educates them about the reasoning behind the recommendations and lets them choose the strategy that they think is the best suited for their risk appetite.

The entire development of the web application is done through an external module available to use in *Python*, known as *Streamlit*. *Streamlit* allows programmers to create a platform to showcase their projects without having to know front-end programming languages like HTML, CSS, JavaScript, etc. It is an easily integrable module for projects based on *Python* and is mostly used for data science and machine learning projects. The web application is divided into three sections: *Project Description*, *Analysis and Results*, and *Get Recommendations*. Additionally, there is a message box through which users can directly and anonymously send their feedback or comments to me. The way to access the application is provided in Appendix A.

Project Description:

The project description section tells the user about the purpose of the project and the motivation behind it, along with the objective and the problem statement. Additionally, it includes a brief explanation of how the investment strategies are developed along with a user journey of how a

potential investor should use the application. Further, it includes a very important risk disclosure about the application being a student project for precautionary reasons.

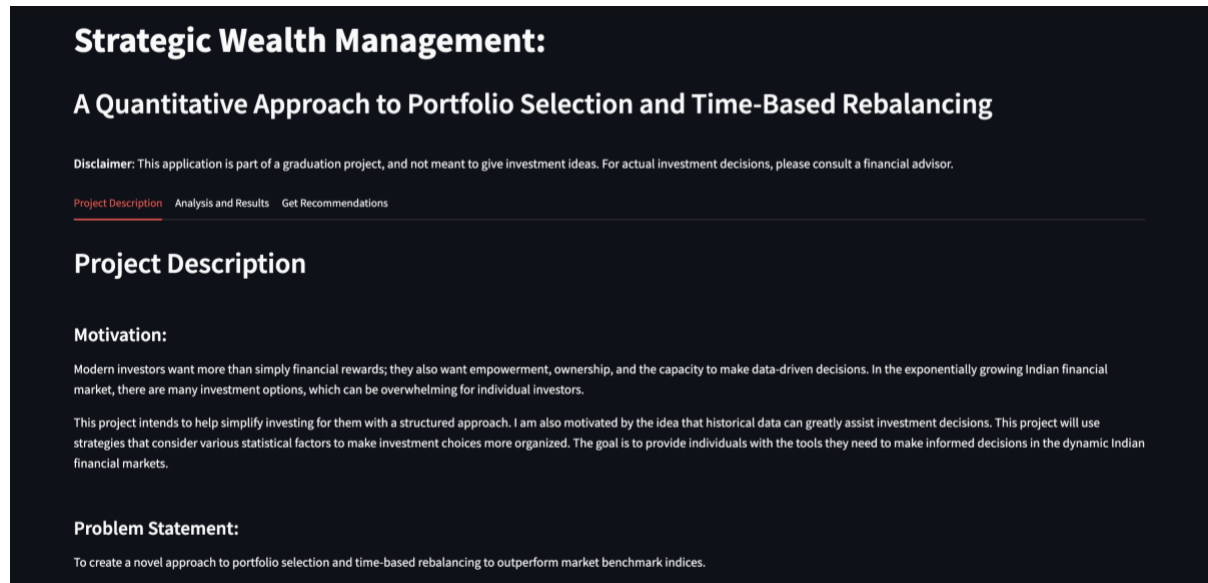


Figure 5.1: A screenshot of the 'Project Description' section of the application

Analysis and Results:

This section of the application helps a user to navigate through the results of the backtesting as well as live-testing of the 5 selected strategies. This also includes a brief description of these strategies along with important metrics of their performance in both phases of the evaluation.

Get Recommendations:

Coming to the last section of the application, this is where all the computations happen in the backend. Here, the user can choose a strategy of their liking, and the one that they think suits their risk appetite after they've gone through the *Analysis and Results* section. They can also choose a buying date for which they are seeking recommendations for a portfolio, which is recommended to be a closer date than a date in the future for more accurate results. Finally, they enter the amount that they want to invest in the recommended portfolio and click on the

“Get Recommendations” button to get a recommended portfolio along with a sector-wise distribution pie chart, the exact number of shares to buy from each recommended entity, and the last traded price (LTP) of those stocks.

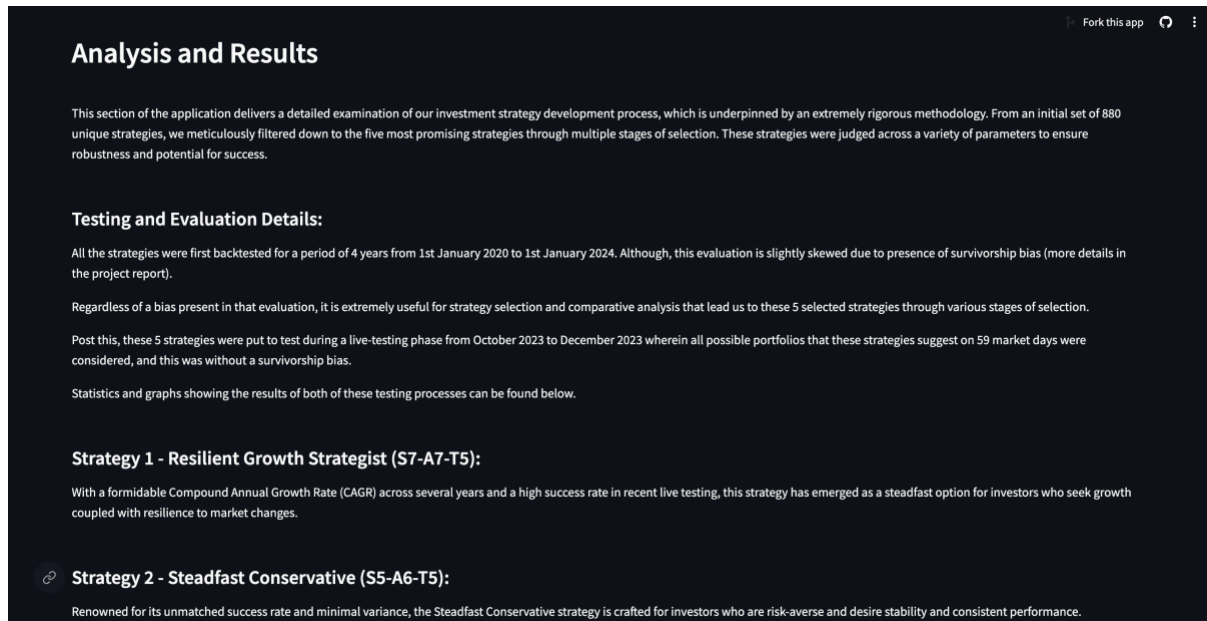


Figure 5.2: A screenshot of the ‘Analysis and Results’ section of the application

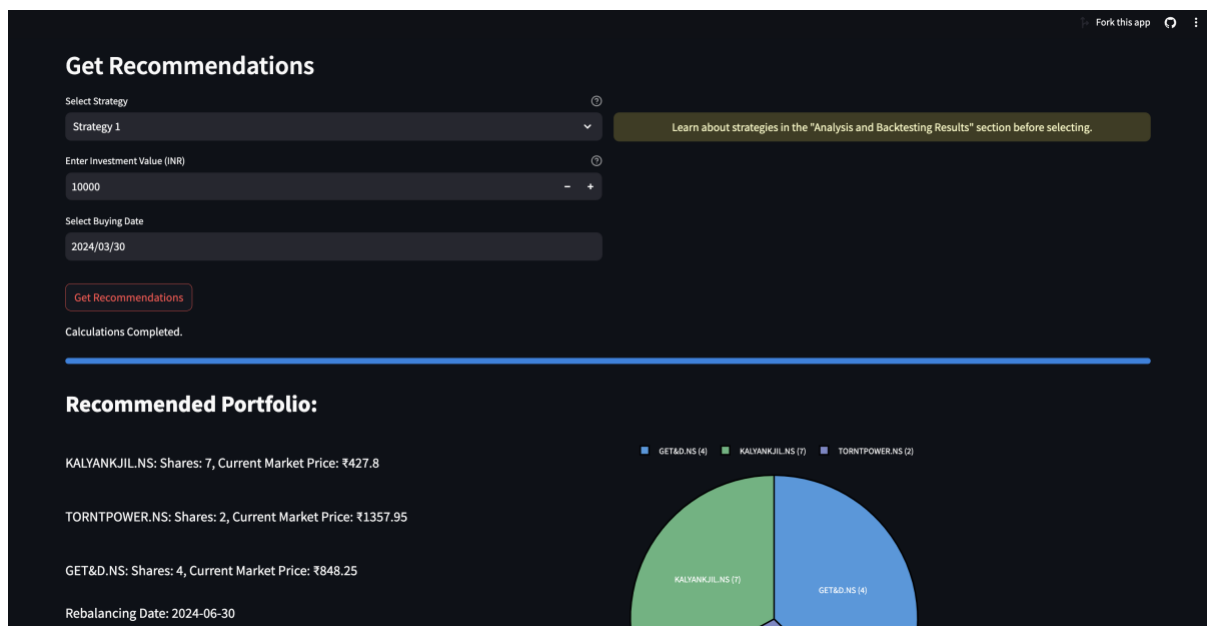


Figure 5.3: A screenshot of the ‘Get Recommendations’ section of the application

Chapter 6: Conclusion and Future Work

6.1. Conclusion & Summary

This project aimed to develop a quantitative investment strategy driven by data, capable of navigating through the complexities and dynamism presented by the Indian equity market.

Through an understanding of financial theories, computational finance, and market analysis, an innovative and dynamic approach to portfolio selection and rebalancing was developed, that responds to the changing market conditions. The project reflected a complete understanding of the complexity evident in the financial market, where a wide range of methodologies for the selection of stocks and weight allocation were devised, and rigorously tested against historical stock price data. The project's attempt to incorporate small-cap and mid-cap businesses in addition to large-cap companies, highlights a strategic diversification that seeks to leverage growth potential while mitigating market volatility. When stock selection techniques, weight allocation strategies, and analysis timelines were strategically combined and tested in a wide range of scenarios, valuable insights were produced.

Various degrees of success were exhibited by the strategies that arose from this rigorous analytical approach, providing insight into the fine balance between risk and reward. Within these, strategies which showed consistency in performance, risk factors that could be controlled, and also took advantage of market dynamics, exhibited the brightest promise. The live-testing phase, which is very critical to this project, helped support the various theoretical constructs and underscored the practicality of the strategies in real life. The success rates observed for this stage reinstated the potential for these strategies to guide investors towards making informed and strategic decisions regarding their investments.

The results based on backtesting need to be acknowledged; however, the analogies of the inherent limitations are especially the necessity of caution in interpretation due to survivorship bias and the access to historical index constituent data that is more comprehensive for refined analysis.

To sum up, this project contributes to a better understanding of the academic and practical perspectives of making quantitative analysis-based investments in the Indian stock market and sets a precedent for future research in this area. In return, it offers a flexible and scalable framework that suits different markets and goals of investment, hence fostering more innovations in the strategies of portfolio management. The work shows the promise that combining advanced analytics with portfolio management has the potential to positively affect data-driven decision-making for investing and trading.

6.2. Future Work & Recommendations

Survivorship Bias: As already discussed above, survivorship bias is the only considerable limitation to the project and removing survivorship bias would make the output of the project a lot better. This is one important area where I plan on doing future work in this project. The only challenge is finding a free yet reliable source for such historical data that helps me get rid of the bias.

Automated Rebalancing Updates: The way the app works currently is that a user can get recommendations, but they will have to keep track of their investment date and the rebalancing period of the strategy they chose and accordingly come back on the application to get another recommendation on the day of its scheduled rebalancing. This would be a very tedious process to follow for an investor. To tackle this issue, users could be asked for their contact or their

email address and the application could keep track of their investments, funds, and strategy choices, and accordingly send them updates about when to rebalance, what to sell, what to buy, how much to buy, etc. This is another interesting feature that would add value to the application in my opinion and I plan on working on it too.

Stop Loss: The strategies always suggest a portfolio, which rarely has all stocks that give positive returns, it's always a possibility that a few stocks out of the recommended portfolio give a loss, and the other stocks' profits make up for that loss. Although, if such cases are rigorously analysed, it is possible to find the reasons behind the inclusion of such stocks that result in a loss, and could potentially be avoided. This is a long-term goal of mine to add such features to the project. I believe that the inclusion of such work would need a lot more statistical and computational work in the background, but it would add immense value to the project and make it commercialisable.

References

- Barnes, Ryan. "Benchmark Your Returns with Indexes." Investopedia, January 31, 2022.
<https://www.investopedia.com/articles/basics/06/benchmark.asp>.
- Ceria, Sebastián, Anureet Saxena, and Robert A Stubbs. "Factor Alignment Problems and Quantitative Portfolio Management." *The Journal of Portfolio Management*, December 27, 2011, 111227035558003. <https://doi.org/10.3905/jpm.2012.2012.1.021>.
- Chen, James. "Annualized Total Return Formula and Calculation." Investopedia, February 29, 2024. <https://www.investopedia.com/terms/a/annualized-total-return.asp>.
- Chen, James. "Backtesting: Definition, How It Works, and Downsides." Investopedia, August 18, 2021. <https://www.investopedia.com/terms/b/backtesting.asp>.
- Chen, James. "Portfolio Weight: Meaning, Calculations, and Examples." Investopedia, March 30, 2020. <https://www.investopedia.com/terms/p/portfolio-weight.asp>.
- Chen, James. "Stock Pick: What It Is, How It Works, Example." Investopedia, May 11, 2022. <https://www.investopedia.com/terms/s/stockpick.asp>.
- Chen, James. "What Is Survivorship Bias? Definition and Use in Investing." Investopedia, October 31, 2021. <https://www.investopedia.com/terms/s/survivorshipbias.asp>.
- Fernando, Jason. "Market Capitalization: What It Means for Investors." Investopedia, March 5, 2024. <https://www.investopedia.com/terms/m/marketcapitalization.asp>.
- Fernando, Jason. "Sharpe Ratio: Definition, Formula, and Examples." Investopedia, January 30, 2024. <https://www.investopedia.com/terms/s/sharperatio.asp>.

Khanra, Sayantan, and Sanjay Dhir. “Creating Value in Small-Cap Firms by Mitigating Risks of Market Volatility.” *Vision: The Journal of Business Perspective* 21, no. 4 (October 25, 2017): 350–55. <https://doi.org/10.1177/0972262917733166>.

Kolm, Petter N., Reha Tütüncü, and Frank J. Fabozzi. “60 Years of Portfolio Optimization: Practical Challenges and Current Trends.” *European Journal of Operational Research* 234, no. 2 (April 2014): 356–71. <https://doi.org/10.1016/j.ejor.2013.10.060>.

Meade, N., J.E. Beasley, and C.J. Adcock. “Quantitative Portfolio Selection: Using Density Forecasting to Find Consistent Portfolios.” *European Journal of Operational Research* 288, no. 3 (February 2021): 1053–67. <https://doi.org/10.1016/j.ejor.2020.06.033>.

Pinkasovitch, Arthur. “Types of Rebalancing Strategies.” Investopedia, October 30, 2022. <https://www.investopedia.com/articles/stocks/11/rebalancing-strategies.asp>.

Tamplin, True. “Tangency Portfolio: Definition, Construction, Pros, & Cons.” Finance Strategists, January 23, 2024. <https://www.financestrategists.com/wealth-management/investment-management/tangency-portfolio/>.

Team, The Investopedia. “Modern Portfolio Theory: What MPT Is and How Investors Use It.” Investopedia, August 29, 2023. <https://www.investopedia.com/terms/m/modernportfoliotheory.asp>.

Appendix A: Project & Applications Resources

Application Link: <https://portfolio-project-fsp-oomrawat.streamlit.app/>

Application Demo Video Link:

<https://drive.google.com/drive/folders/1fTnt0RPjvHrPP5MFAIhRHmcfx9aDo2Xd?usp=sharing>

Presentation Link: <https://www.canva.com/design/DAF-3KXfMj0/e20kSLEiKOQT3mYY1C0QdA/edit>

GitHub Repository Link: <https://github.com/oomrawat/Portfolio-Project-FSP>

Appendix B: Code Resources

'streamlit_app.py': https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/app/streamlit_app.py

'recommendations.py': <https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/app/recommendations.py>

'functions.py': <https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/src/functions.py>

'weight_allocation_strategies.py': https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/src/weight_allocation_strategies.py

'optimization_functions.py': https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/src/optimization_functions.py

'main.ipynb': <https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/notebooks/main.ipynb>

'pipeline.ipynb': <https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/notebooks/pipeline.ipynb>

'backtesting_evaluation.ipynb': https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/notebooks/backtesting_evaluation.ipynb

'backtesting_results_visualisation.ipynb': https://github.com/oormrawat/Portfolio-Project-FSP/blob/main/notebooks/backtesting_results_visualisation.ipynb

Appendix C: Data Sources

Historical Stock Price Data:

- Link: https://github.com/oomrawat/Portfolio-Project-FSP/blob/main/data/all_stock_data.csv or <https://finance.yahoo.com/>
- Source: Yahoo Finance

Government Bond Price Data:

- Link: <https://github.com/oomrawat/Portfolio-Project-FSP/blob/main/data/India%2010-Year%20Bond%20Yield%20Historical%20Data.csv> or <https://in.investing.com/rates-bonds/india-10-year-bond-yield-historical-data>
- Source: Investing.com - India 10-Year Bond Yield

Index Constituent Data:

- Link: https://github.com/oomrawat/Portfolio-Project-FSP/blob/main/data/ind_niftytotalmarket_list.csv or <https://www.nseindia.com/products-services/indices-nifty-total-market-index>
- Source: National Stock Exchange (nseindia.com)

Appendix D: Additional Graphs and Visualisations

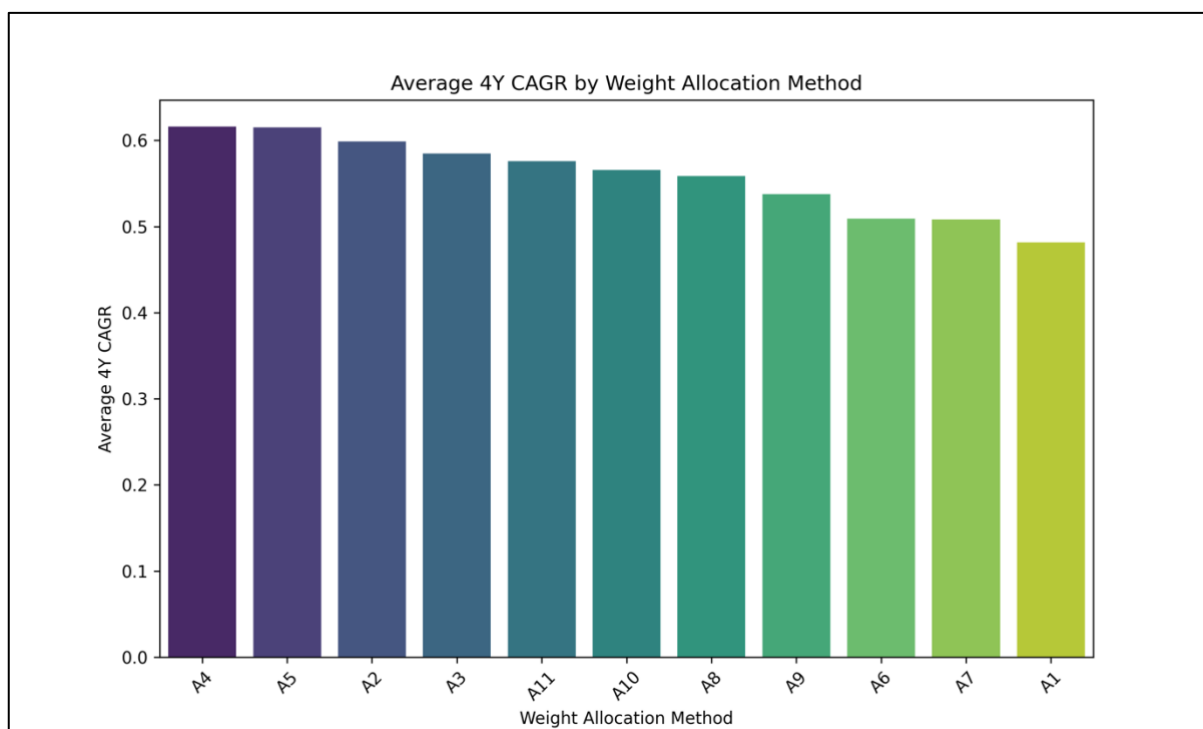


Figure D.1: Average 4Y CAGR by Weight Allocation Method

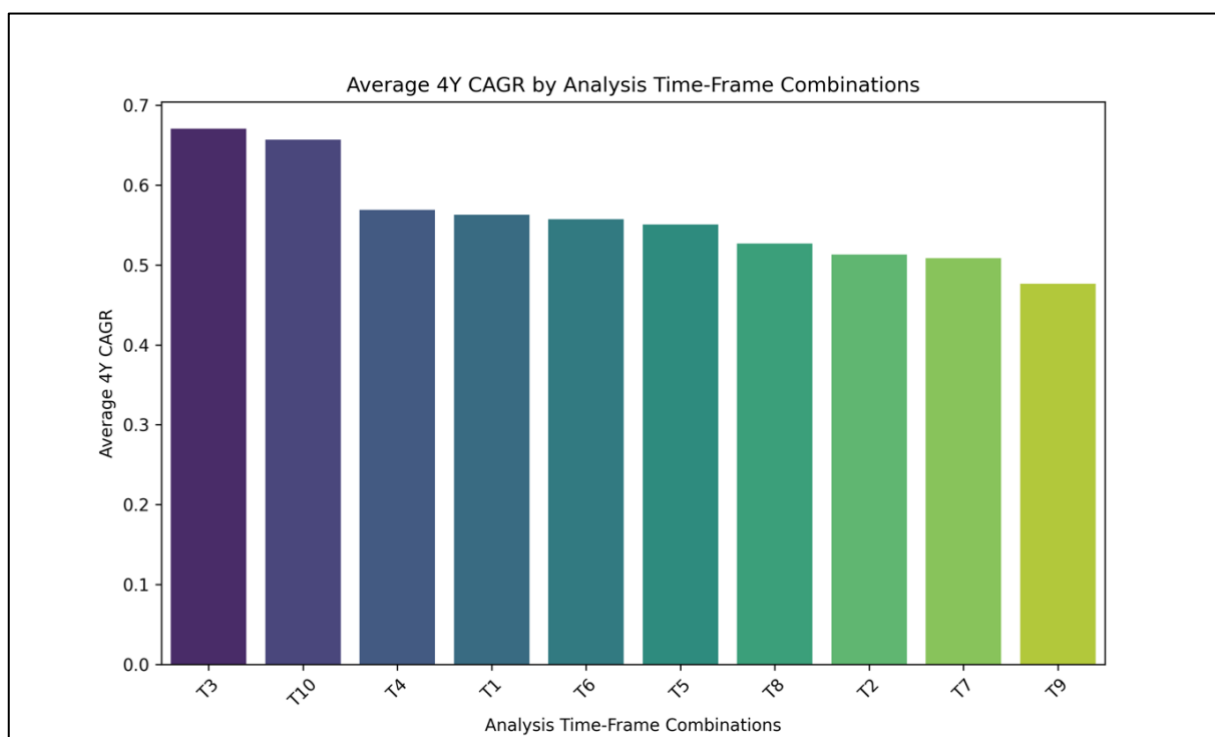


Figure D.2: Average 4Y CAGR by Analysis of Time-Frame Combinations

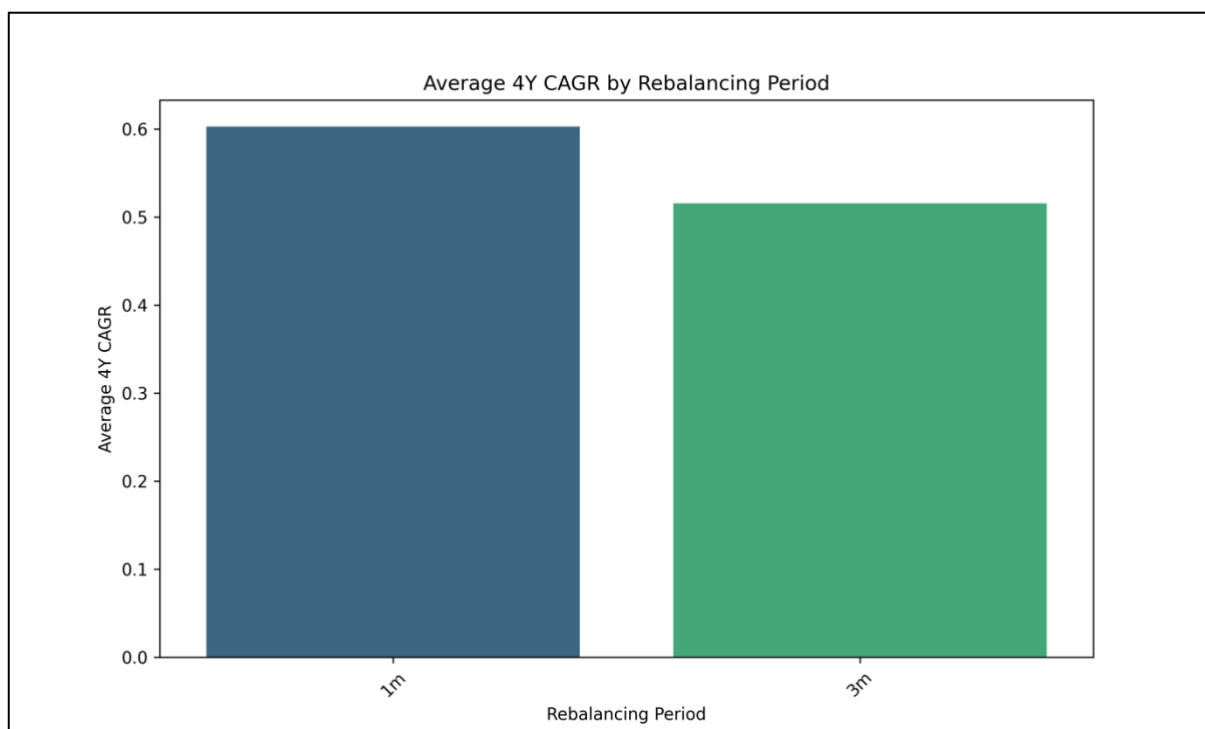


Figure D.3: Average 4Y CAGR by Rebalancing Period

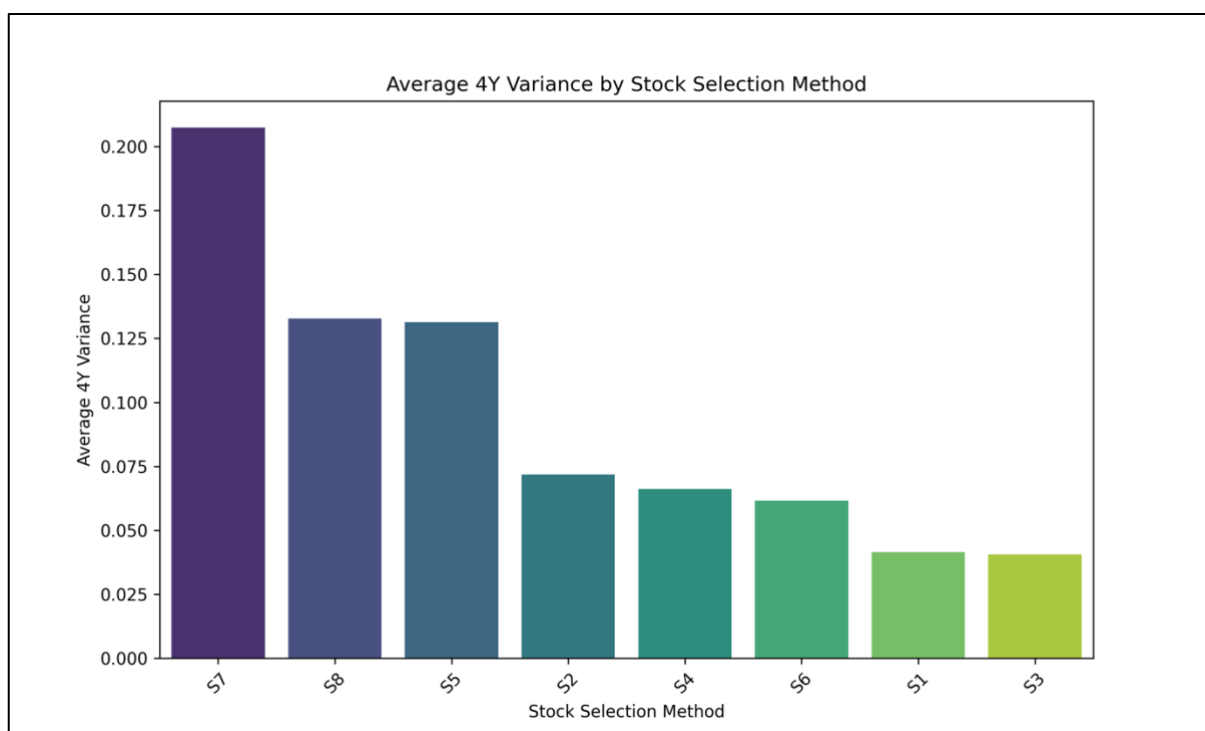


Figure D.4: Average 4Y Variance by Stock Selection Method

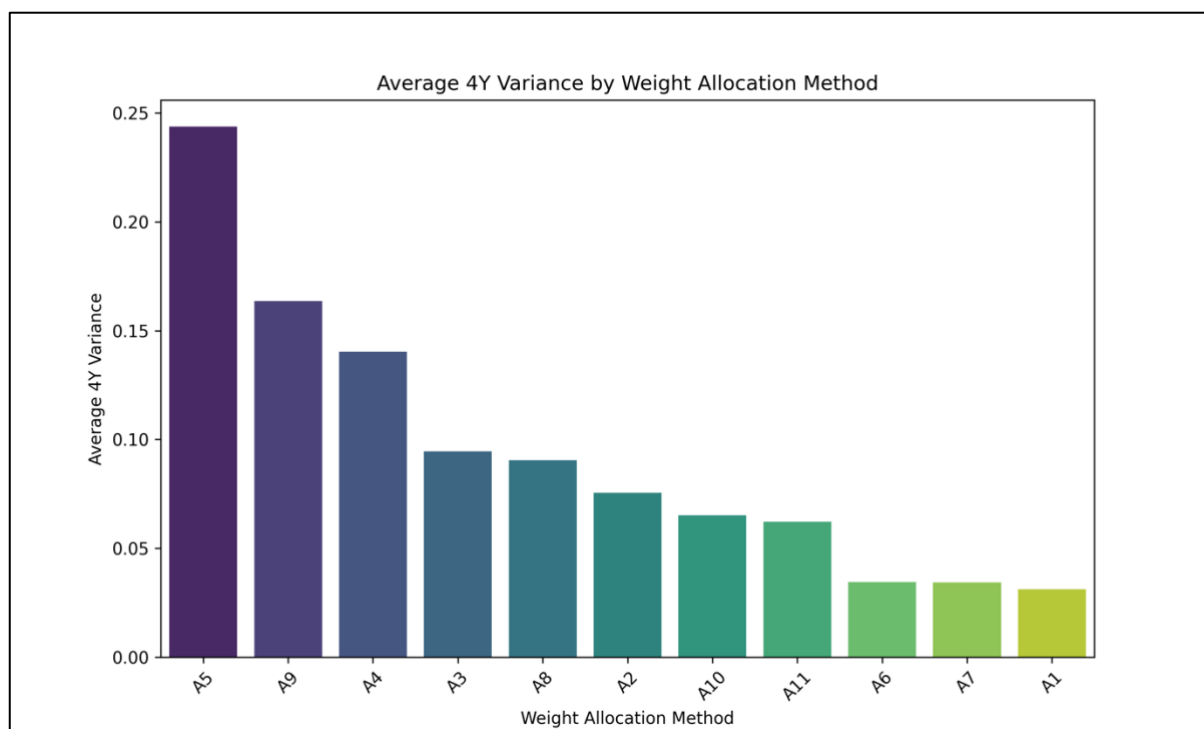


Figure D.5: Average 4Y Variance by Weight Allocation Method

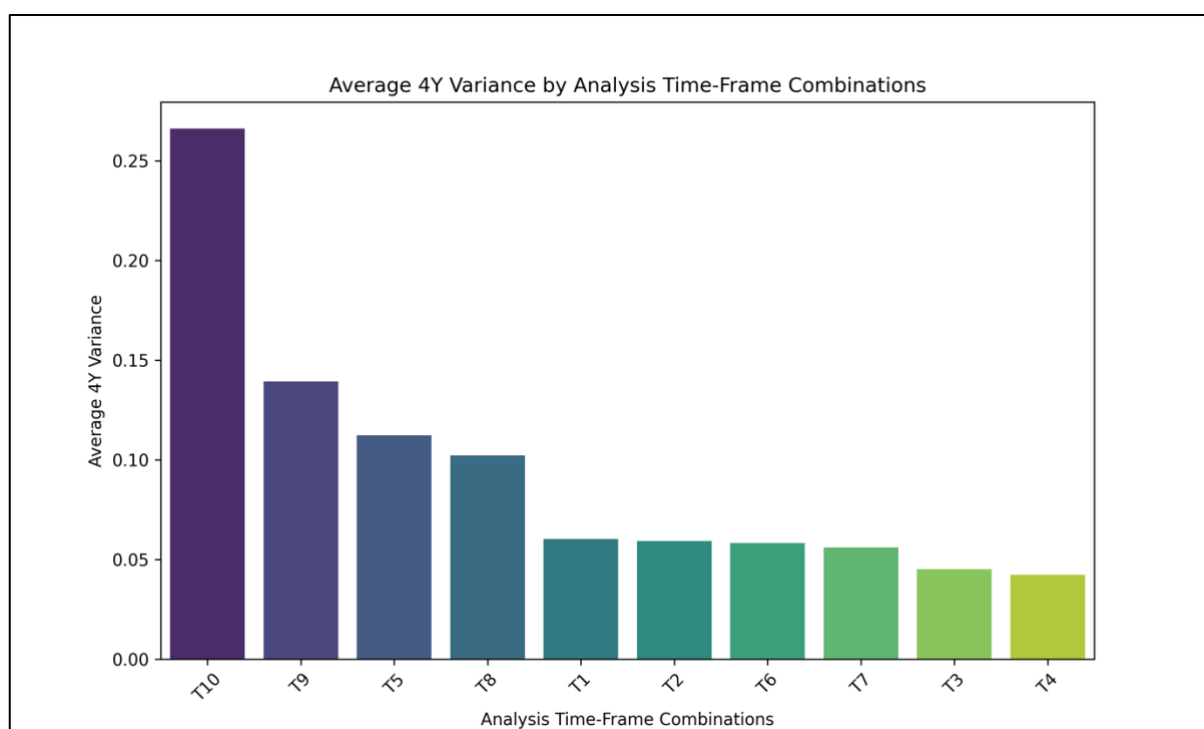


Figure D.6: Average 4Y Variance by Analysis Time-Frame Combinations

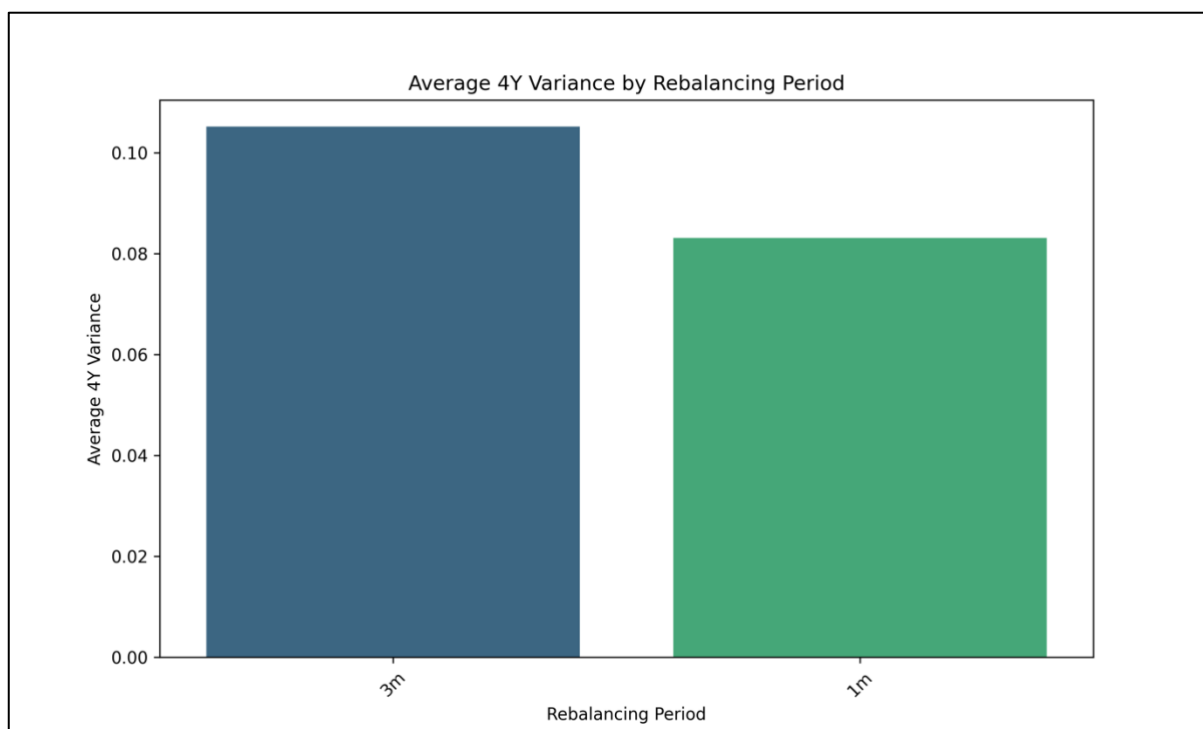


Figure D.7: Average 4Y Variance by Rebalancing Period

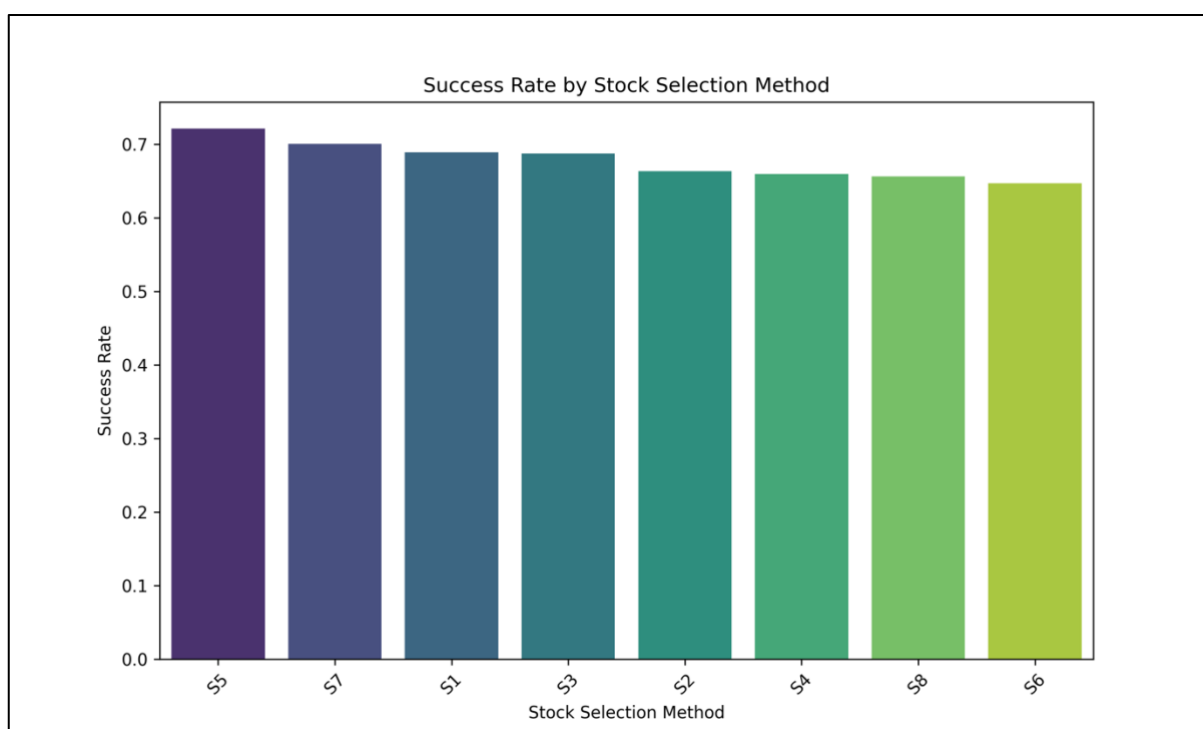


Figure D.8: Success Rate by Stock Selection Method

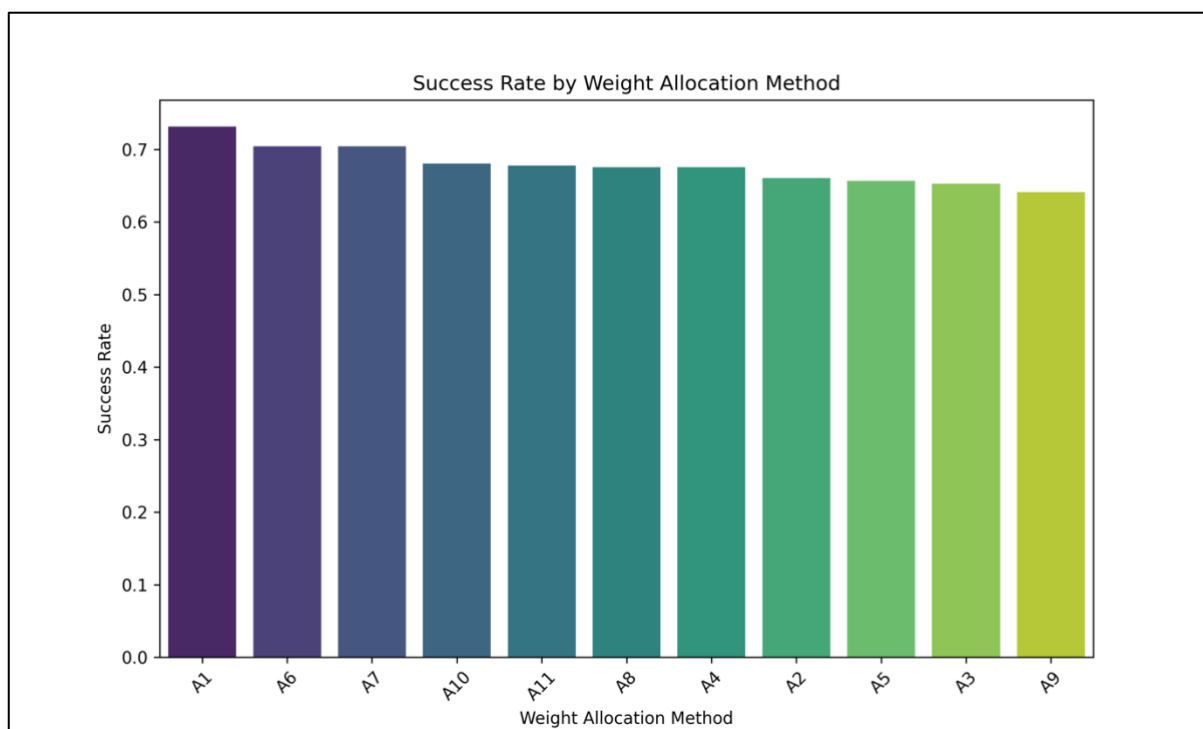


Figure D.9: Success Rate by Weight Allocation Method

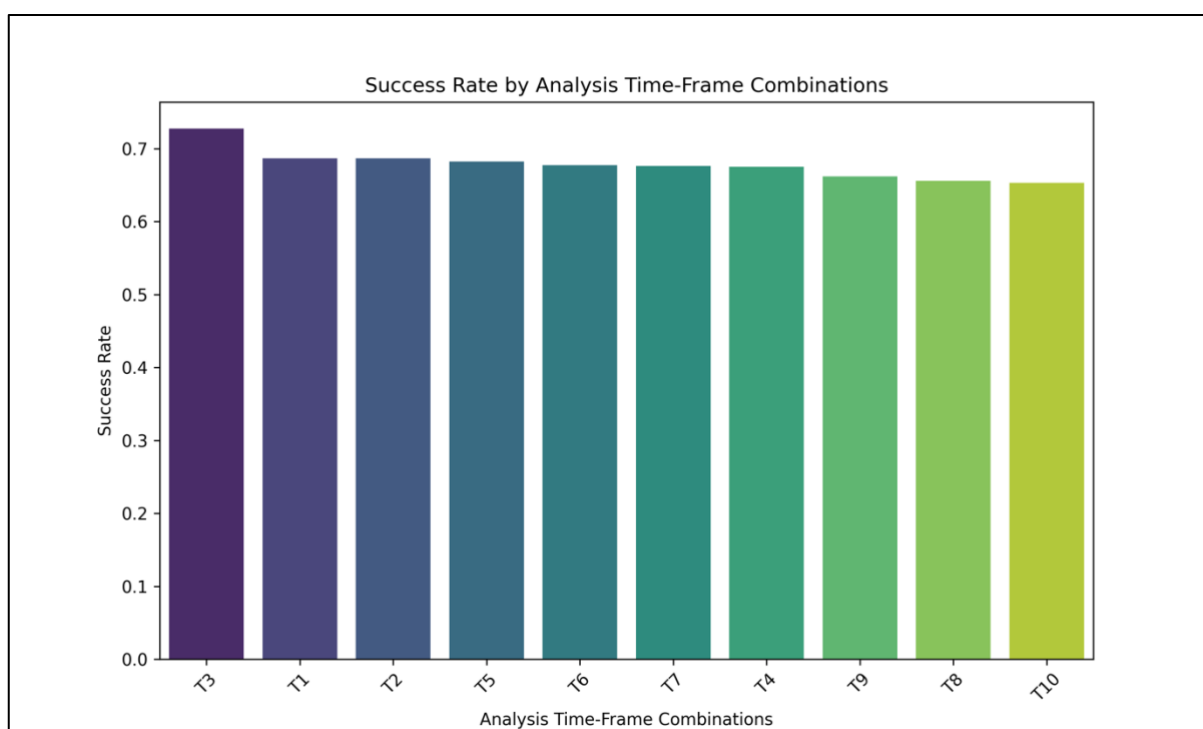


Figure D.10: Success Rate by Analysis Time-Frame Combinations

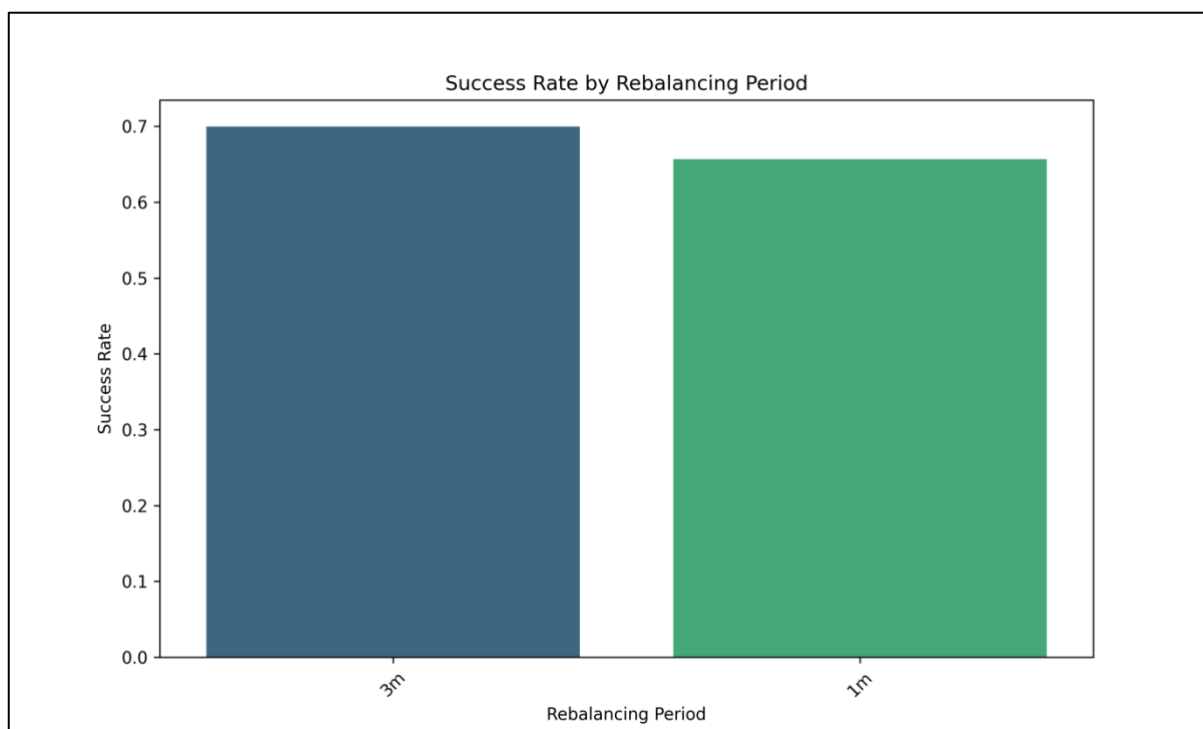


Figure D.11: Success Rate by Rebalancing Period

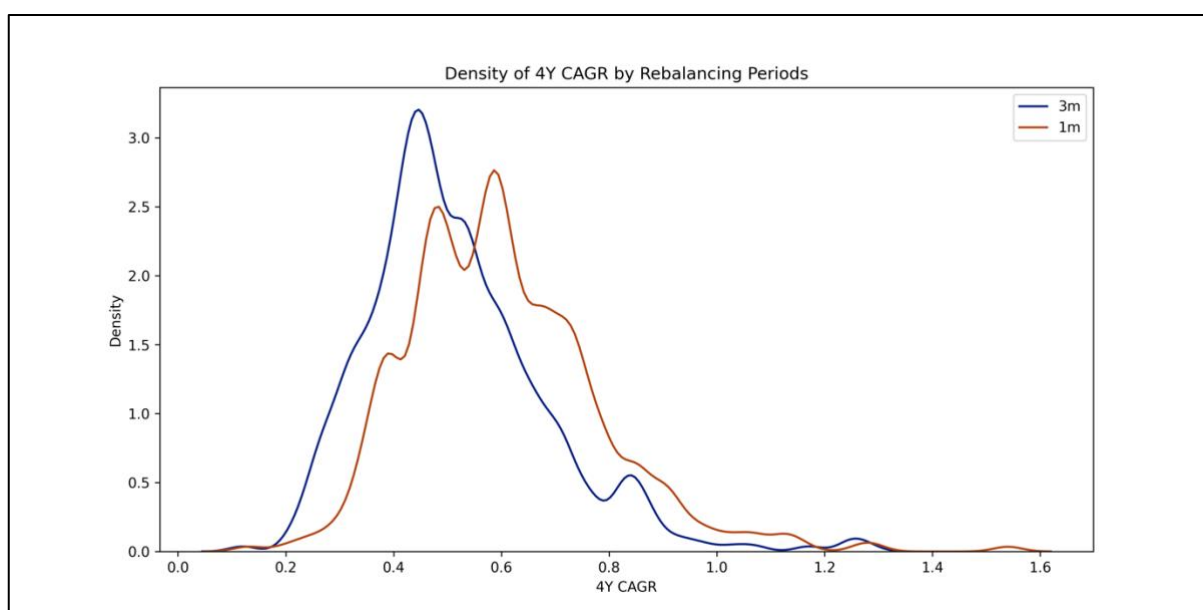


Figure D.12: Density Distribution of 4Y CAGR by Rebalancing Periods

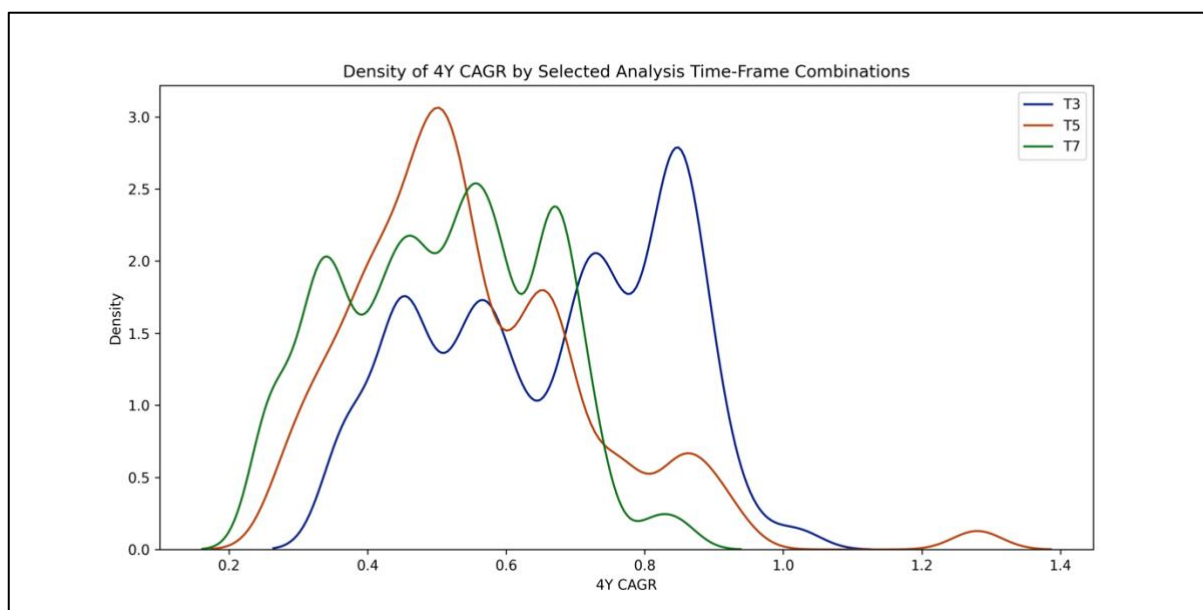


Figure D.13: Density Distribution of 4Y CAGR by Selected Analysis Time-Frame Combinations

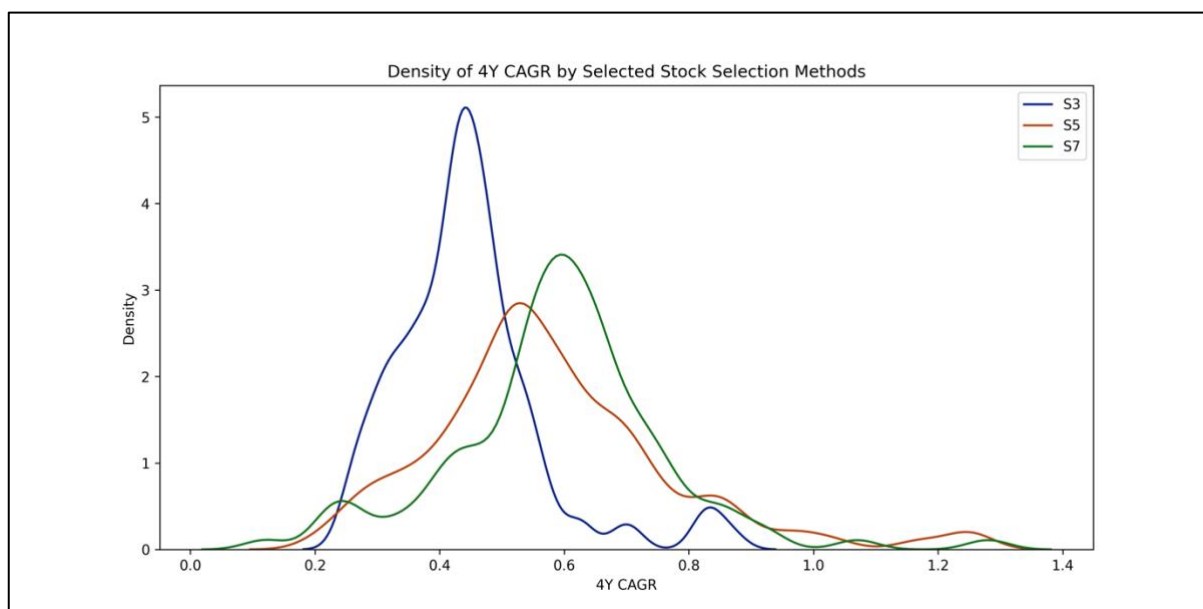


Figure D.14: Density Distribution of 4Y CAGR by Selected Stock Selection Methods

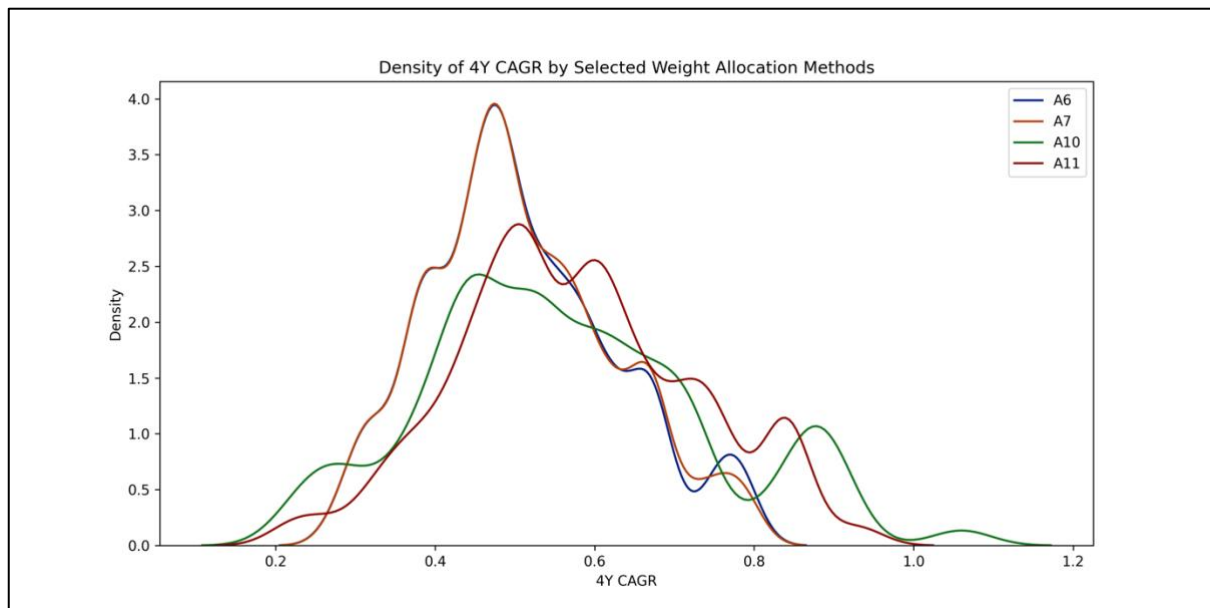


Figure D.15: Density Distribution of 4Y CAGR by Selected Weight Allocation Methods

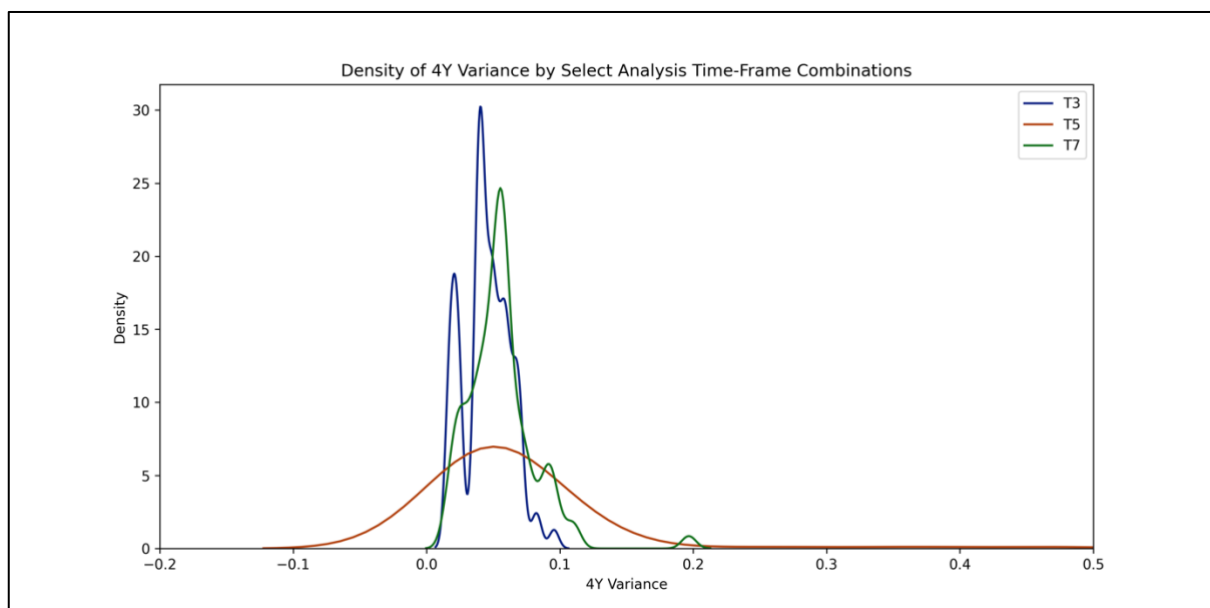


Figure D.16: Density Distribution of 4Y Variance by Selected Analysis Time-Frame Combinations

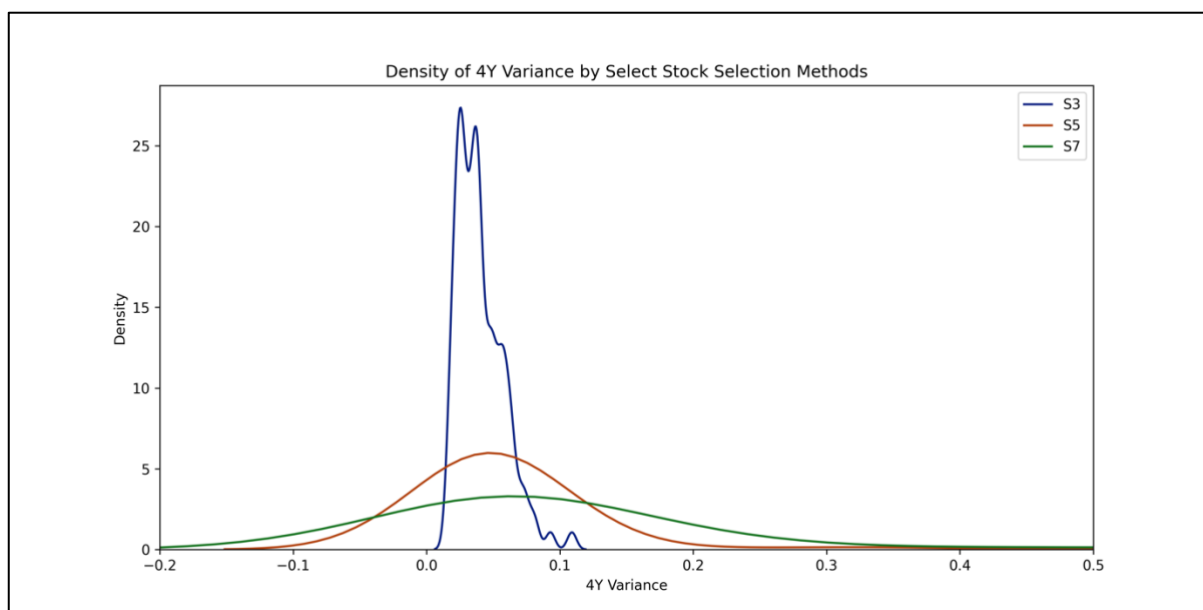


Figure D.17: Density Distribution of 4Y Variance by Selected Stock Selection Methods

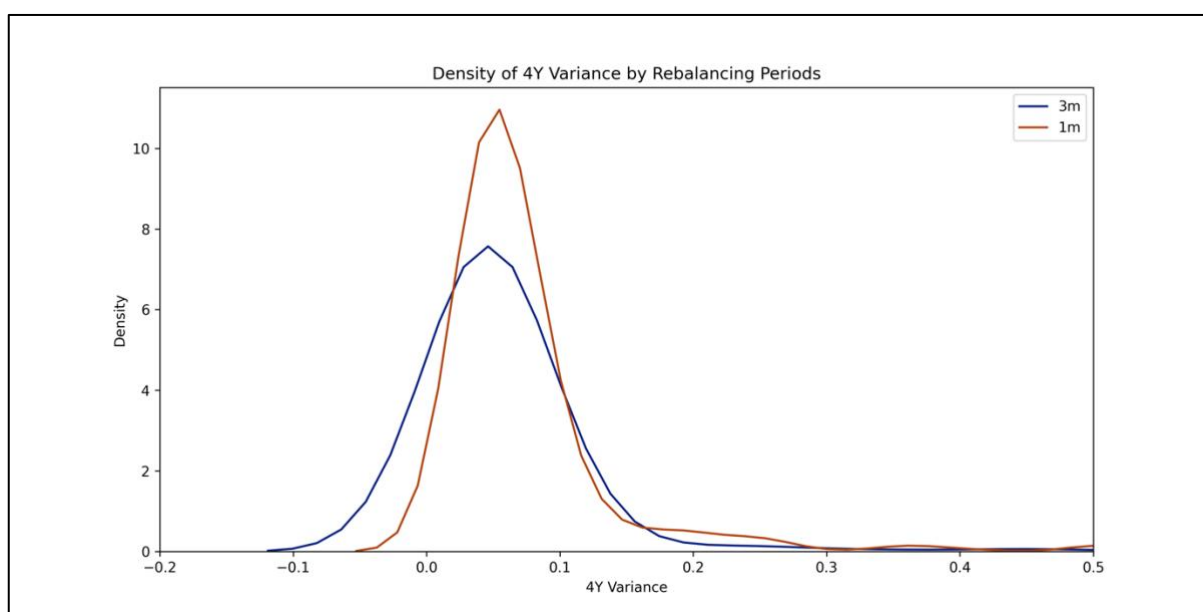


Figure D.18: Density Distribution of 4Y Variance by Rebalancing Periods

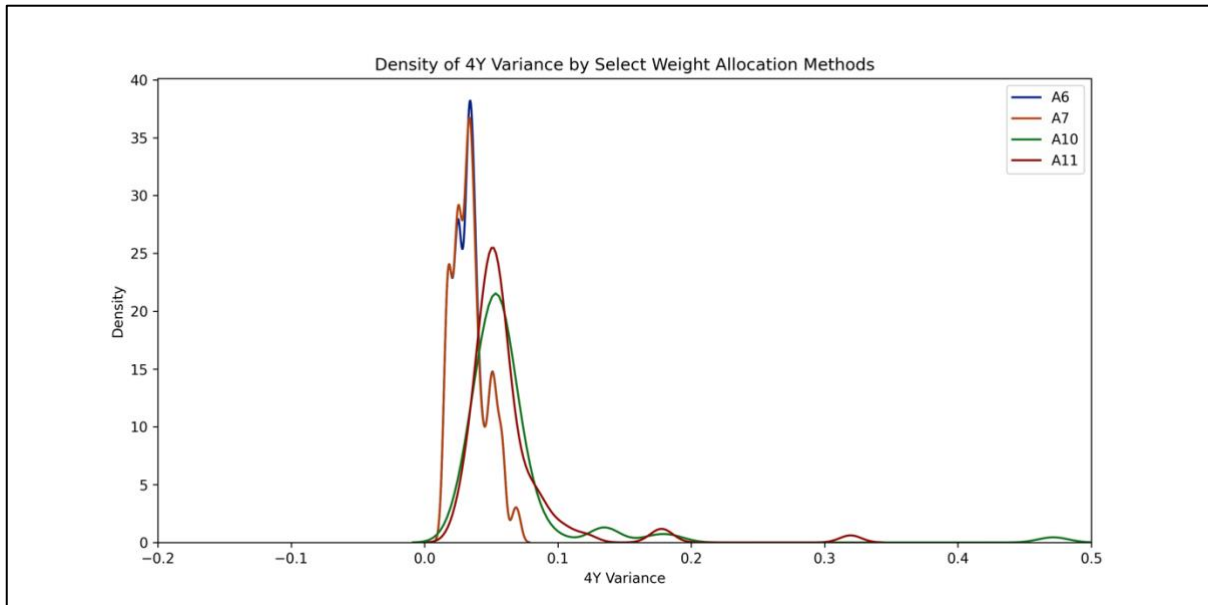


Figure D.19: Density Distribution of 4Y Variance by Selected Weight Allocation Methods

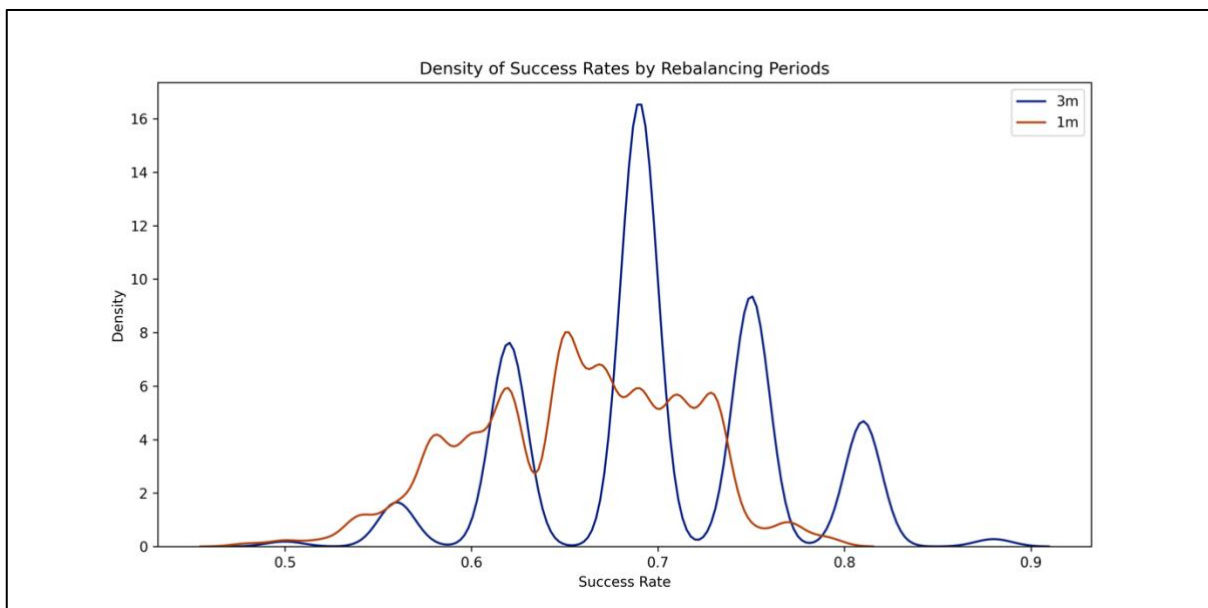


Figure D.20: Density Distribution of Success Rates by Rebalancing Periods

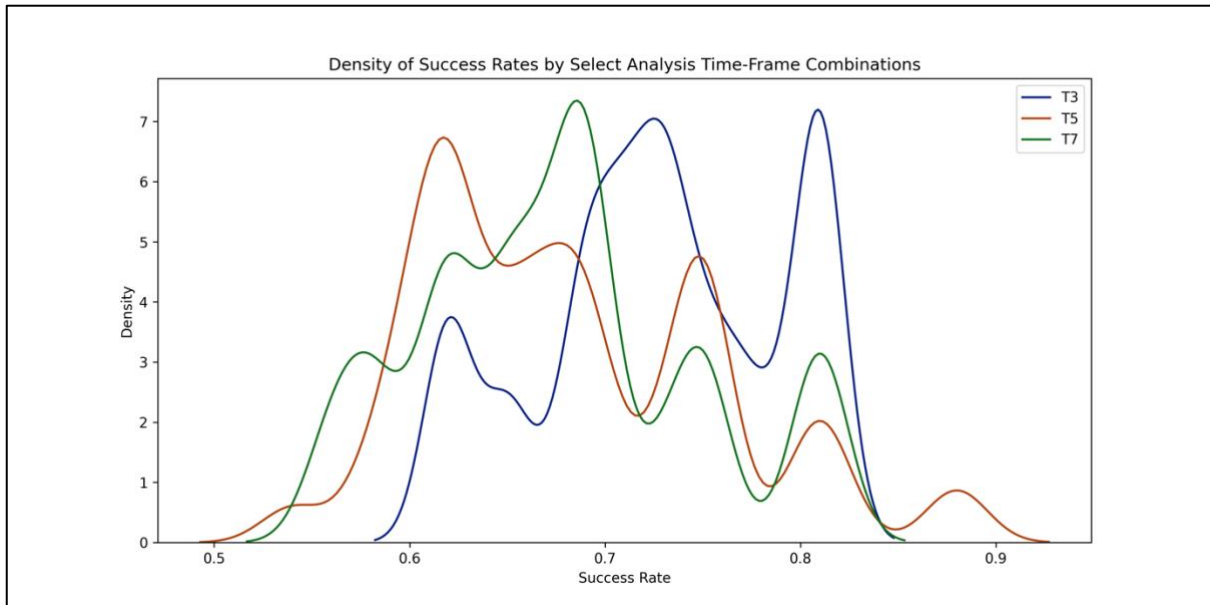


Figure D.21: Density Distribution of Success Rates by Analysis Time-Frame Combinations

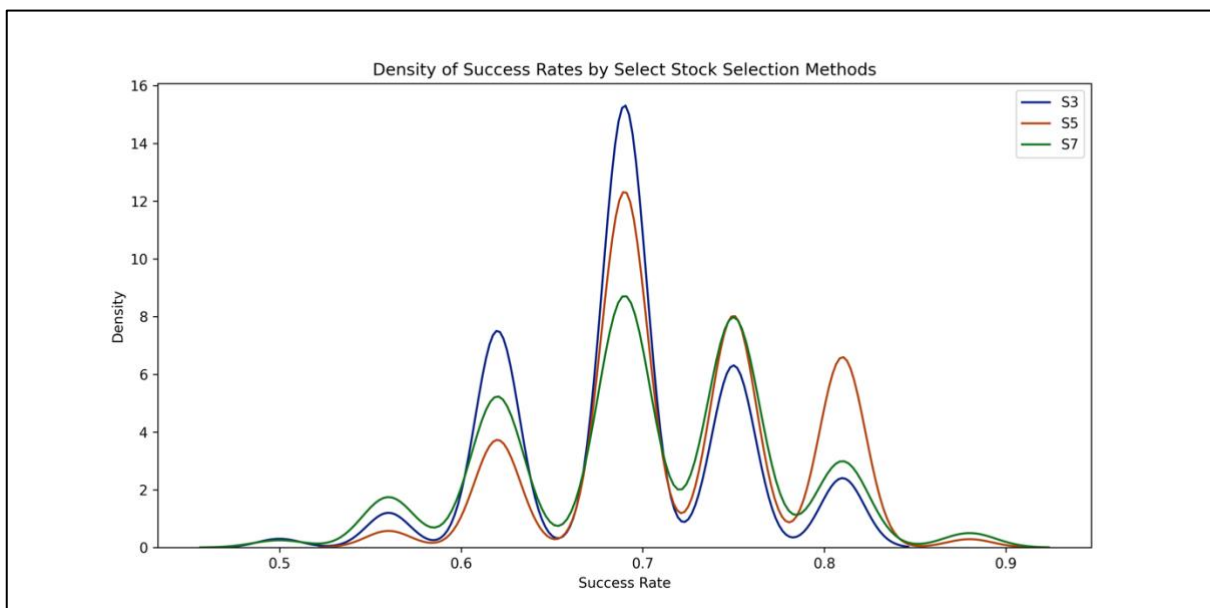


Figure D.22: Density Distribution of Success Rates by Stock Selection Methods

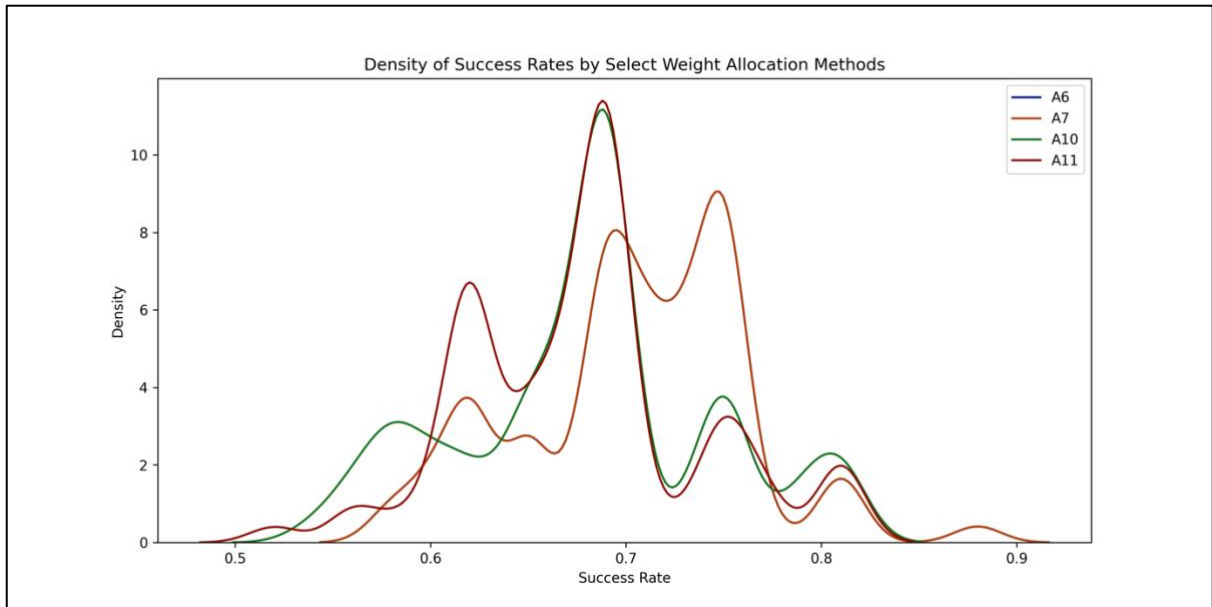


Figure D.23: Density Distribution of Success Rates by Weight Allocation Methods