ISSN 1041-2743 (Online)

International Journal of Finance

Volume 37
Issue 1
2025

Dilip K. Ghosh (Editor-in-Chief)
Alan Wing-Keung Wong (Co-Editor-in-Chief and Managing Editor)
Vincent Shin-Hung Pan (Managing Editor)

SCIENTIFIC & BUSINESS WORLD

Published by Scientific and Business World

Enhancing Portfolio Allocation with Machine Learning-Based Return Predictions: Does Frequency Matter

Wong Zhi Zhe

School of Management, Universiti Sains Malaysia

Hooy Chee Wooi

School of Management, Universiti Sains Malaysia

Corresponding author **Email**: cwhooy@usm.my

Abstract

Research aims: This study explores the integration of machine learning and optimization techniques (both classical & heuristic) in portfolio allocation, focusing on how data frequency influences predictive accuracy and investment performance.

Design/Methodology/Approach: Using five major technology stocks (AAPL, MSFT, GOOG, AMZN, META), six regression models—Ridge, Linear, Random Forest, XGBoost, Multilayer Perceptron (MLP) and Support Vector Regression (SVR)—were employed to predict asset returns based on historical data at daily, weekly, and monthly frequencies. Predicted returns were then optimized using three methods: Sequential Least Squares Programming (SLSQP), Particle Swarm Optimization (PSO), and Differential Evolution (DE), under realistic weight constraints (0.05 to 0.8 per asset, no short-selling).

Research findings: While machine learning (ML) models achieve higher predictive accuracy with higher-frequency (daily) data, this does not necessarily lead to better portfolio performance. PSO consistently outperformed other optimizers across all frequencies. Notably, portfolios optimized using PSO with weekly and monthly predictions delivered higher Sharpe ratios than the equal-weighted benchmark, while daily-frequency portfolios underperformed.

Theoretical contribution/Originality: This study offers valuable insights into the integration of machine learning in portfolio optimization, highlighting the effect of data frequency on the performance of the ML-embedded optimized portfolio.

Practitioner/Policy implication: Moderate to lower data frequencies may provide more robust signals for ML-embedded portfolio optimization, offering a better balance between prediction quality and investment returns.

Keywords: Portfolio Optimization, Machine Learning, Data Frequency, Particle Swarm Optimization, Predictive Accuracy, Investment Performance

JEL classification: C45, C61, G11, G17

1 Introduction

Portfolio allocation plays a critical role in investment decision-making, where individuals seek a balance between potential returns and risk through the optimal distribution of capital across a wide range of assets within the portfolio, such as the mix of stocks, bonds, and cash (Croome, 2025). According to Capital Market Theory and Modern Portfolio Theory (MPT), the market portfolio is the optimal portfolio. It is theoretically the most efficient portfolio for investors (Indeed Editorial Team, 2025). Market portfolio is the diversified portfolio that includes every investable asset in the world, with each asset weighted in proportion to its total presence in the market (Chen, 2020). However, practically, it is infeasible to construct a true market portfolio. Investors will face several limitations that can block them from constructing an optimal market portfolio, including a lack of access to all global assets, concerns about estimation accuracy, and involvement of transaction costs, making the market portfolio exist theoretically and is practically impossible (Chen, 2020). Therefore, investors often turn to traditional investment strategies such as equal-weighting and Markowitz's Mean-Variance Optimization (MVO) model, which have guided many investors for a long time (Chen, 2021; Tütüncü, 2011). The traditional portfolio allocation strategies, including equal-weight portfolios and mean-variance optimization, have been widely adopted due to their simplicity and historical success (Chen, 2021). However, these methods are mostly backward-looking, solely relying on historical returns and covariances. Besides, these models often work based on unrealistic assumptions of normality, linearity, and market efficiency (Fahmy, 2019). According to past empirical evidence, the financial returns are frequently non-normal (Abtahi, 2023). As a result, these classical allocation strategies may not capture the market dynamics and nonlinear nature of financial markets adequately, especially in today's increasingly volatile and data-rich environments.

Nowadays, the advancement of machine learning (ML) offers another promising alternative to investors. ML algorithms are capable of learning complex and nonlinear patterns from historical data and forecasting asset returns in a forward-looking manner (França et al., 2021). The forecasting results can then be used to guide portfolio construction, allowing more responsive allocation strategies that can capture the dynamic capital market better (Chen et al., 2020). By incorporating ML regression models to forecast future asset returns with allocation strategies, investors can make more informed decisions and potentially enhance their portfolio performance (Pinelis & Ruppert, 2021). For example, they can integrate the predictions of ML models into the optimization algorithm to generate insights into asset weight adjustments in their portfolio. Moreover, the choice of data frequency, such as daily, weekly, or monthly dimensions, may also affect the performance of both predictions and portfolio optimization. Generally, data with higher frequency (daily dimension) may better reflect the time-to-time market changes, suggesting higher responsiveness (Liu & Liu, 2024). On the other hand, the lower-frequency data (weekly or monthly dimensions) is not too sensitive to the immediate fluctuation but may reduce the noise and tends to improve long-term model stability (Glova & Barrios, 2024).

Although Machine Learning (ML) provides a compelling alternative, the effectiveness of ML-based return forecasting for portfolio optimization is yet to be widely explored, especially when taking the frequency of the input data (daily, weekly, monthly) into consideration, which might affect ML's prediction accuracy and the performance of optimized portfolios. Although ML models have shown their potential in predictive finance in previous studies, there is limited literature comprehensively assessing how daily, weekly, and monthly dimensions of data can influence the optimization outcomes, and whether this ML-optimized portfolio can outperform the simple but robust equal-weighted portfolio in a real-world setting (Gupta et al., 2025).

This research aims to explore the potential of ML-enhanced optimization by using ML regression models to forecast asset returns at different time frequencies (daily, weekly, and monthly dimensions) and using these predictions to optimize asset weights with the goal of maximizing the Sharpe ratio. By evaluating the effectiveness of ML in forecasting future asset returns across different data frequencies (daily, weekly, and monthly), and applying the forecasts in optimization approaches, this research can offer practical insights for enhancing portfolio performance in a risk-adjusted manner. It offers valuable insights into how frequency-aware ML models can be leveraged for improved portfolio construction, identifies the most suitable data frequency for effective ML-driven strategies, and provides guidance for practitioners on optimization algorithm selection for real-world applications. Fellow stakeholders, including asset management firms, financial advisors, and FinTech platforms, may benefit from the findings by understanding which data frequencies and ML models offer the best trade-off between predictive accuracy and portfolio returns. The study also provides valuable insights regarding the selection of optimization algorithms for real-world use. All of these can help the practitioners and investors construct a more efficient portfolio to effectively achieve higher returns while taking on a certain level of risk.

In the following sections, Section 2 reviews relevant academic and industry literature related to traditional portfolio theories, machine learning applications in finance, asset return prediction, and optimization algorithms. It identifies gaps in existing research that this study aims to address. Section 3 details the methodological approach, including data collection, preprocessing, model development, portfolio construction, and performance evaluation techniques. It also explains the tools and algorithms employed for forecasting and optimization. Sections 4 and 5 present the findings from the predictive modeling and portfolio optimization process, which will include comparative analysis across data frequencies and models and discuss the performance of ML-driven portfolios relative to the equal-weighted benchmark. Besides, the recommendations for practitioners and future researchers will also be provided. Limitations of the study are also discussed. The last section summarizes key insights from the research and highlights contributions to the field.

2 Portfolio Optimization with Machine Learning Prediction

Few studies integrate both machine learning-based predictions and heuristic optimization techniques in a comprehensive framework. Although both machine learning in return prediction and machine learning in portfolio optimization have been substantially studied, they are being analyzed separately. There is now a growing interest in integrating these two areas to enhance portfolio management strategies (Bhasin et al., 2024). However, this area has not been explored extensively yet. The potential of machine learning in enhancing portfolio optimization is still under wraps. This study explores how advanced optimization algorithms can be coupled with predictive models to better maximize risk-adjusted returns

2.1 Machine Learning Prediction

Machine Learning (ML) is part of computer science and a subset of Artificial Intelligence (AI), where it refers to the computer (ML algorithms) that can study the patterns of complex and high-dimensional historical data and make forward-looking predictions (França et al., 2021). In the context of predicting future returns, the primary focus is on regression-based supervised ML models as they can produce continuous data. Besides linear regression, ML models include Ridge Regression, Support Vector Regression (SVR), Random Forest, XGBoost, and Multilayer Perceptron (MLP) can be used in financial return forecasting. These models incorporate a wide variety of features, including momentum indicators, volatility measures, and lagged returns, allowing them to capture nonlinear or dynamic relationships often present in financial markets.

Ridge Regression is an enhanced version of linear regression. It is also known as L2 regularization, which can resolve the overfitting problem effectively by penalizing large coefficients (Miller et al., 2021). Besides, it is also useful when dealing with multicollinearity issues among the features or predictors. McDonald et al. (2014) have incorporated both linear regression and ridge regression to forecast a number of financial time-series data.

Support Vector Regression (SVR) is a more complex ML model compared to linear regression. It is a kernel-based method that aims to find a function that stays within a certain margin of error (called epsilon), focusing more on making accurate predictions for new data rather than just fitting the training data perfectly. According to Comito and Pizzuti (2022), Support Vector Regression is an extension of SVM, which introduces a region, named a tube, around the function to optimize with the aim of finding the tube that best approximates the continuous-valued function, while minimizing the prediction error at the same time. SVR is also well known for its capabilities in handling nonlinear relationships through its kernel functions, making it suitable for financial return forecasting. He (2024) has shown that SVR can outperform traditional models like ARIMA in stock return forecasting, indicating that it is effective in capturing the complex financial market.

Ensemble ML models are strong learner models built by combining multiple weak learners, such as a decision tree model. Random Forest Regression is an ensemble ML. The key concept is constructing multiple decision trees during training and averaging their outputs to improve predictive accuracy and prevent overfitting. Random Forests can capture complex interactions and nonlinearities effectively. It is splitting the data by mainly using metrics such as information gain and Gini impurity (Bulagang et al., 2020). On the other hand, XGBoost is another ensemble model, which is a powerful gradient boosting model that is built by combining multiple decision trees in a sequential manner, unlike the parallel manner of Random Forest (Mesut et al., 2023). XGBoost is aiming to optimize residual errors at each iteration. XGBoost is well known for its high performance, scalability, and ability to handle sparse data, making it a popular choice for financial time series forecasting. Recent research has highlighted the superior performance of both Random Forest and XGBoost against traditional models in predicting asset prices and returns (Qian & Zhang, 2025).

Multilayer Perceptron (MLP) is a commonly used neural network. According to Chan et al. (2023), MLP is composed of multiple layers, including an input layer, hidden layers, and an output layer, where each layer contains a set of perception elements known as neurons. MLPs can model complex nonlinear functions and interactions by learning hierarchical feature representations. However, the overfitting issue might happen and affect the performance of the model if it lacks sufficient data for training. It has been applied successfully in asset pricing and stock return forecasting, demonstrating its applicability in return predictions (Abolmakarem et al., 2024).

2.2 Heuristic Portfolio Optimization Techniques

For optimization of the predicted portfolio, 3 methods are utilized. They include one traditional algorithm, which is Sequential Least Squares Programming (SLSQP). This gradient-based optimization model will be implemented via scipy.optimize.minimize function in Python. The other two algorithms are heuristic optimization models, which are DE (Differential Evolution) and Particle Swarm Optimization (PSO). Past empirical studies have shown their success and effectiveness in portfolio optimization, indicating that these models are appropriate choices that align with the research objectives and are able to construct a reliable portfolio that serves the objective function: optimizing the Sharpe Ratio.

SLSQP was developed by Kraft (1988) to solve nonlinear optimization problems (Ma et al., 2024). SLSQP has gained popularity in financial engineering and portfolio optimization due to its ability to work with practical constraint settings, such as weight bounds and full investment (total weight equal to 1) constraints (Garg et al., 2025). In the context of portfolio optimization, a recent study by Garg et al. (2025) has leveraged SLSQP to find the optimal allocation of assets that minimizes portfolio volatility or maximizes risk-adjusted return (Sharpe ratio). This paper shows SLSQP as one of the critical methods that can achieve maximum portfolio diversity.

The second optimization method, DE, is a powerful evolutionary algorithm commonly used for solving complex optimization problems, including portfolio allocation. DE can be implemented using DEAP (Distributed Evolutionary Algorithms in Python), a flexible evolutionary computation framework. DE evolves a population of candidate solutions using operations like selection, crossover, and mutation. In portfolio optimization, DEAP provides the ability for practitioners to customize the model by themselves to handle complex, non-convex objective functions such as Sharpe ratio maximization, which can better tailor their objectives. DEAP's modular structure makes it ideal for experimenting with different evolutionary strategies and constraints (Fortin et al., 2012). In fact, DE is particularly effective in asset allocation where traditional gradient-based methods (e.g., SLSQP) may struggle due to non-convexity or over-complicated constraints. In past literature work, DE has been applied to optimize portfolios, achieving high expected returns and low investment risks when considering cardinality and bounding constraints (Kao & Chen, 2013).

Last but not least, we also include PSO, which was introduced by Kennedy and Eberhart (1995). PSO is a population-based stochastic optimization technique inspired by the social behavior of birds and fish. It has gained attention in the context of portfolio optimization since it can explore large and complex solution spaces efficiently (Chen et al., 2021b). Besides, PSO has been widely applied to solve non-convex and constrained. Furthermore, PSO has good flexibility, which can handle various objective functions, such as maximizing returns, minimizing risks, and balancing risk and return (Díez-Fernández et al., 2012). In 2021, PSO was used to implement the Markowitz mean-variance model to optimize the trade-off between risk and return, and this is considered a cornerstone of modern portfolio theory (MPT) (Mercangöz, 2021). Other empirical studies have also shown that PSO can often outperform genetic algorithms (GA) in terms of speed and sensitivity to initial conditions, making it a more reliable choice for portfolio optimization (Yourdkhani, 2014).

2.3 The Role of Time Frequency in Portfolio Optimization

Data frequency plays a critical role in shaping the performance of both predictive models and portfolio strategies. Generally, high-frequency data, such as daily or even minutes, micro-second data, can significantly enhance the accuracy of financial predictions. For instance, models that incorporate daily financial data have shown improved forecast accuracy for economic indicators like GDP growth and inflation compared to those using only monthly data (Andreou et al., 2013). Additionally, high-frequency data allows for real-time updates, which is crucial for timely decision-making.

On the other hand, the lower frequency data: weekly data offers a balance between more detail and noise reduction. It smooths the data by not including so much volatility that can appear in daily data while still being responsive to medium-term patterns. Rossini et al. (2019) have shown that weekly data can help reduce the noise in daily dimension data, which can be particularly beneficial in financial forecasting, where high-frequency data can be overly noisy. Monthly data emphasizes long-term trends and structural changes in the market. It minimizes the effects of short-term noise and is often used for strategic asset allocation or economic modeling, such as macroeconomic events (Galvão & Owyang, 2022). However, its lower frequency comes with fewer data points, which may affect the training quality of the ML models. Besides, the low-frequency data, like monthly data, cannot capture the details of market movement and will react more slowly to the fast-changing market conditions.

In the context of portfolio optimization, ML-based return forecasts derived from different data frequencies also impact the resulting portfolio composition and performance. High-frequency predictions may lead to more dynamic but riskier portfolios, while low-frequency forecasts support more stable, long-term allocations. Identifying the frequency that aligns best with a given investment strategy is key to effective ML-driven portfolio construction. However, many prior studies have applied ML models to forecast asset returns but have not systematically compared the impact of different data frequencies (daily, weekly, and monthly) on model performance and portfolio outcomes. This study can fill this gap by evaluating how prediction frequency influences both the accuracy of ML-based return forecasts and the performance of subsequent ML-optimized portfolio construction.

3 Research Data and Methodology

3.1 Data Source

The dataset comprises daily closing prices of five selected technology sector stocks: Apple Inc. (AAPL), Microsoft Corporation (MSFT), Alphabet Inc. (GOOG), Amazon.com, Inc. (AMZN), and Meta Platforms, Inc. (META). These assets were chosen randomly without strictly adhering to the principle of diversification in portfolio management, mainly to illustrate the application of machine learning models in portfolio construction. The historical price data was retrieved via the Yahoo Finance API, covering a period of five years from January 1, 2020, to December 31, 2024. The dataset includes daily closing prices, resulting in a data frame of 6 columns (Date plus the five assets' closing prices) and 1,256 rows.

The data analysis comprises 3 stages. In the first stage, the data are fitted to a set of ML regression models, which include a linear regression, Ridge Regression, Support Vector Regression (SVR), Random Forest, XGBoost, and Multi-Layer Perceptron (MLP). The ML models are then evaluated based on their accuracy in forecasting the asset returns using different input frequencies. The second stage is portfolio optimization, where the best ML regression model for each data frequency will be integrated into three portfolio optimization algorithms. Those algorithms are a classical approach called SciPy's Sequential Least Squares Programming (SLSQP), and two heuristic approaches: Differential Evolution (DE), and Particle Swarm Optimization (PSO). These models will provide their best recommended portfolio allocation decision to optimize the Sharpe Ratio. Lastly, all the recommended portfolios will be simulated using the actual historical returns, which is also known as back testing, and their performance will be benchmarked against an equal-weighted portfolio across all frequencies. This comparison can help assess whether ML adds value and which data frequency results in the most efficient allocation.

3.2 Data Processing

The original data has been restructured with the following steps, where k below represents the day, week, or month frequency:

- Closing prices are converted into returns by calculating the percentage change from the previous one to the next.
- Rolling momentum features are computed as the moving averages over 5-k and 10-k windows to capture short- and medium-term trends.
- Volatility features are extracted by calculating the rolling standard deviation over 5-k and 10-k windows, serving as a measure of recent price fluctuations.

After feature engineering, three datasets corresponding to daily, weekly, and monthly frequencies are created, each containing returns and rolling statistical features for the selected assets. The blank values in each data set are then removed to ensure data integrity. Finally, all features are scaled using standardization techniques to normalize the data, ensuring better stability and performance during machine learning model training.

The descriptive statistics and correlation table of the dataset are reported accordingly. From Table 1, we can see that all assets exhibit relatively low average daily returns, ranging from 0.000931 (AMZN) to 0.001234 (META), which is typical for high-frequency financial data. In terms of risk, as measured by standard deviation, we can observe that META is the most volatile asset (2.83%), while MSFT is the least volatile (1.92%). The minimum and maximum return values indicate that each asset has experienced significant short-term fluctuations. META shows the widest range, including a minimum return of -26.39% and a maximum of +23.28%. The skewness and kurtosis values further reveal deviations from normality in the return distributions. Most assets exhibit mild skewness, with AAPL and AMZN showing slight positive skewness, while GOOG and META are slightly negatively skewed. Notably, META and MSFT display extremely high kurtosis values (18.08 and 7.12, respectively), indicating heavy tails and the presence of outliers. These findings suggest that the return distributions are not normal, which is quite common in a complex real-world financial market setting. However, the portfolio optimization or risk management approach should account for these statistical features.

Table 1: Descriptive Statistics

Asset	Mean Return (%)	Std. Dev. (%)	Min (%)	Max (%)	Skewness	Kurtosis
AAPL	0.001189	0.019962	-0.128647	0.119808	0.104675	5.273339
AMZN	0.000931	0.022663	-0.140494	0.135359	0.069483	3.991863
GOOG	0.001036	0.020406	-0.111008	0.099652	-0.065325	3.557899
META	0.001234	0.028294	-0.263901	0.232824	-0.297335	18.082593
MSFT	0.000995	0.019217	-0.147391	0.142169	-0.017843	7.115708

3.3 The Evaluation Criteria for ML Prediction and Portfolio Optimization

The dataset is split into training and testing subsets to evaluate model performance reliably and prevent overfitting. In this study, the 80:20 train-test split is applied, where the ML models are trained on 80% of the data and evaluated on the remaining 20%. This step is conducted for all three return datasets (daily, weekly, monthly). Two metrics have been used to measure the accuracy performance of ML models.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right|$$

Mean Absolute Error (MAE) is the metric used to measure the average absolute difference between predicted and actual returns, where Y_i is the actual return at that point and $\hat{Y}_i$ is the predicted return of that point, reflecting the magnitude of prediction errors (Xia et al., 2023)

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Mean Squared Error (MSE) is another metric used to measure the average squared difference between predicted and actual returns, penalizing larger errors more heavily than MAE. The lower the MAE and MSE, the more accurate the models. These metrics provide complementary insights into model accuracy and error distribution (Zhao et al., 2023).

The primary objective of this portfolio optimization is to maximize the Sharpe Ratio, which represents the optimal trade-off point between returns and risk. The Sharpe Ratio is calculated by using the expected portfolio return to subtract the risk-free rate, then dividing by the portfolio's standard deviation. In this study, the risk-free rate will be assumed as zero to maintain the simplicity of the implementation and to provide a clearer and straightforward perspective. The best machine learning model for each frequency will be integrated into the optimization algorithms, and their predicted returns will serve as the foundation for the optimization algorithm to decide the allocation of the asset weights that seek the highest risk-adjusted performance.

To avoid overconcentration in any single asset, the following constraints are imposed on all optimization algorithms:

- No short-selling allowed: Asset weights must be non-negative.
- Weight bounds: Each asset's weight must be between 0.05 (5%) and 0.8 (80%) to encourage diversification and prevent extreme allocations.
- Fully invested portfolio: The sum of all asset weights must equal 1, ensuring that the entire capital is allocated.

By applying these constraints, the optimization process aims to produce realistic and diversified portfolios suitable for practical investment.

To establish a baseline for portfolio performance comparison, a naïve equal-weighted portfolio is constructed using the same assets. The equal-weighted portfolio assigns an identical weight to each asset without any predictive optimization, representing a simple and commonly used investment strategy. Performance results of the machine learning-optimized portfolios will be benchmarked against this equal-weighted portfolio in later back testing to assess the effectiveness and improvement brought from ML to portfolio optimization across different data frequencies (daily, weekly, and monthly).

3.4 Back Testing

Back testing is conducted using historical walk-forward validation (Chen, 2024). In other words, the optimized portfolio weights generated based on predicted data are applied to actual historical returns to simulate real-world investment performance over time. Notes that:

- A total of nine optimized portfolios (three per frequency × three optimization algorithms) are constructed by applying the optimized weights to historical returns.

- Additionally, three equal-weighted portfolios are created for the same periods and assets.
- The optimized portfolios and benchmarks are compared across the portfolio evaluation metrics to assess performance differences.

This back-testing approach allows the researcher to evaluate how the optimized portfolio weights can perform in real-world settings, providing insights into the effectiveness of ML-integrated optimization under different prediction frequencies.

4 Results and Discussion

4.1 Results on Return Prediction

This section presents the evaluation results of the machine learning (ML) regression models on predicting the asset returns using daily, weekly, and monthly data frequencies. The models were evaluated using Mean Absolute Error (MAE) and Mean Squared Error (MSE) metrics. The objective is to identify the best model for each frequency to be used later in portfolio optimization.

As we can observe in Table 2, basically, Ridge regression has the best performance when predicting returns using daily frequency data. On the other hand, SVR is outperforming other ML models in both weekly and monthly data predictions. Therefore, Ridge regression and SVR were chosen to integrate into the optimization algorithms; their predictions will serve as inputs to the portfolio optimization process, under the assumption that improved return predictions can enhance portfolio performance. The consistently poor performance of the MLP Regressor across all frequencies suggests that it is unsuitable for this application with the current data and features, or it might require further hyperparameter-tuning to better capture the pattern of the training data.

Table 2 Predictive model performance on daily return data.

	Daily Data		Weekly Data		Monthly Data	
Models	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**
Linear Regression	0.012769	0.000333	0.039248	0.002651	0.242139	0.094566
Ridge Regression	0.012643	0.000327	0.033506	0.001931	0.145989	0.031499
SVR	0.021621	0.000897	0.029193	0.001491	0.060695	0.006331
Random Forest	0.014745	0.000433	0.029927	0.001570	0.068412	0.007054
XGBoost	0.016876	0.000532	0.036660	0.002158	0.103248	0.016470
MLP Regressor	0.199514	0.051020	0.311055	0.110001	0.325317	0.137814

Overall, these findings emphasize that the choice of predictive models should be frequency dependent. Simpler linear methods, such as linear regression and ridge regression, are more suitable for predicting short-term returns. In contrast, those more complex, flexible non-linear models are preferred for longer-term returns. This strategic selection maximizes the forecasting accuracy, which is crucial for subsequent portfolio optimization. Additionally, the predictive model evaluation across different data frequencies reveals a clear pattern, where the lower the data frequency, the lower the accuracy of the predictions. This declining accuracy is observed consistently across all machine learning models based on both MAE and MSE metrics. This situation aligns with the past empirical works that suggest higher frequency data tends to increase the accuracy of prediction (Andreou et al., 2013).

4.2 Results on Portfolio Optimization

The optimized portfolio weights by each algorithm under each data frequency are shown. Besides, the performance metrics of each portfolio constructed using the recommended weights will also be presented. Notes that the performance metrics reported below are based on back testing using historical return data. For each frequency (daily, weekly, monthly), portfolios were optimized with three different algorithms: SLSQP, PSO, and DE. The results show how these algorithms allocate weights across assets and how the optimized portfolio performs over the historical period.

Table 4 presents the optimized weights for daily frequency portfolios. Both SLSQP and DE assign a dominant weight of 0.8 to Google (GOOG) and 0.05 to each of the other assets, whereas PSO allocates weights more evenly across the assets. Back testing shows that expected portfolio returns are similar across algorithms (0.0010), but PSO demonstrates slightly better risk management, where it achieves the lowest volatility (0.0188) and the highest Sharpe ratio (0.0543). For weekly data, SLSQP and DE portfolios both show a high concentration on META (weighted 0.8) and allocate 0.05 weight to other assets. PSO also allocated more weight (0.55452) to META, but it still distributes weights more broadly compared to SLSQP and DE. Back-tested results indicate PSO's portfolio performs the best with the highest expected return, lowest volatility, and yet a superior Sharpe ratio (0.1419). At the monthly frequency, SLSQP and DE still concentrate heavily on META (0.77689). Similar to the daily and monthly scenarios, PSO balances allocations more evenly. Other than the weight of 0.47654 assigned to META, PSO allocates significant weight to Google (0.27936) and Amazon (0.1384). PSO's portfolio back testing results yield the highest Sharpe ratio (0.3052), indicating better risk-adjusted performance.

Table 3 Optimized Weights

	AAPL	MSFT	GOOG	AMZN	META	Total weight
Daily Data						
SLSQP	0.05	0.05	0.8	0.05	0.05	1.0
PSO	0.08681	0. 34932	0. 4447	0. 05506	0. 06411	1.0
DE	0.05	0.05	0.8	0.05	0.05	1.0
Weekly Data						
SLSQP	0.05	0. 05	0. 05	0.05	0. 8	1.0
PSO	0. 21477	0. 08887	0. 09095	0. 05089	0. 55452	1.0
DE	0.05	0. 05	0. 05	0.05	0. 8	1.0
Monthly Data						
SLSQP	0. 05	0.05	0.05	0. 07311	0. 77689	1.0
PSO	0. 05178	0. 05392	0. 27936	0. 1384	0. 47654	1.0
DE	0. 05	0.05	0.05	0. 07311	0. 77689	1.0

Table 4 Optimized Portfolios' Evaluation

	Expected Return	Volatility	Sharpe Ratio
Daily Data			
SLSQP	0.0010	0.0195	0.0537
PSO	0.0010	0.0188	0.0543
DE	0.0010	0.0195	0.0537
Weekly Data			
SLSQP	0.0047	0.0353	0.1335
PSO	0.0049	0.0344	0.1419
DE	0.0047	0.0353	0.1335
Monthly Data			
SLSQP	0.0193	0.0648	0.2977
PSO	0.0201	0.0660	0.3052
DE	0.0193	0.0648	0.2977

The portfolio optimization results highlight the consistent superiority of the PSO algorithm in generating optimal portfolio weights that deliver better risk-adjusted returns compared to other algorithms. Among the three algorithms, PSO consistently outperformed DE and SLSQP across all frequencies when evaluated using real historical data. Specifically, PSO-optimized portfolios based on weekly and monthly frequency data outperform the equal-weighted portfolios, achieving higher Sharpe Ratios and lower volatilities. This highlights PSO's strength in effectively seeking better trade-offs between return and risk in a real-world setting.

Traditionally, high-frequency data is used to perform better in optimizing a portfolio (Alemany et al., 2020). However, this past result was tested without ML involvement. The involvement of machine learning deepens the algorithm in terms of complexity, and the result might be different in this context. Although predictive accuracy decreases at lower frequencies, the smoothing effect of less frequent data aggregation seems to contribute to more stable and reliable portfolio allocations. In contrast, the daily frequency optimized portfolios showed a Sharpe Ratio slightly below that of the equal-weighted portfolio, suggesting that more frequent rebalancing may introduce noise and instability despite better prediction accuracy.

Overall, the results suggest that PSO is a robust choice for portfolio optimization, especially when applied to weekly and monthly data frequencies. The involvement of ML models in optimization shows that more accurately predicted data might not translate to better portfolio performance. In contrast, lower frequency data with lower accuracy prediction results, surprisingly, provided more stable long-term perspectives and led to superior portfolio performance.

4.3 Benchmarking Equal-weighted Portfolio

Back testing was performed on equal-weighted portfolios constructed from the same set of assets and frequency periods to serve as benchmarks. Table 5 compares the performance metrics of PSO's optimized portfolios against these benchmarks since PSO's portfolio is performing the best in terms of Sharpe Ratio maximization across every data frequency. In the short-term perspective (daily), the equal-weighted portfolio has higher expected returns, lower volatility, and a higher Sharpe Ratio, outperforming PSO's portfolio in all metrics. In the medium-term (weekly) and long-term (monthly) perspectives, the equal-weighted portfolio is generating slightly higher expected returns. However, PSO's portfolio is way more stable in terms of portfolio volatility. Thus, PSO's portfolio has better overall performance against an equal-weighted portfolio, with high Sharpe Ratios. These results suggest that machine learning-enhanced portfolio optimization can provide meaningful improvements over naïve weighting schemes, particularly over longer time horizons.

Table 5 Benchmarking: PSO's portfolios vs Equal Weighted portfolios

Portfolio Type	Expected Return	Volatility	Sharpe Ratio
Equal-Weighted (Daily)	0.0011	0.0187	0.0576
Optimized (Daily)	0.0010	0.0188	0.0543
Equal-Weighted (Weekly)	0.0050	0.0361	0.1394
Optimized (Weekly)	0.0049	0.0344	0.1419
Equal-Weighted (Monthly)	0.0212	0.0714	0.2977
Optimized (Monthly)	0.0201	0.0660	0.3052

5 Conclusion and Implications

This study explored a machine learning-driven approach to portfolio optimization using historical stock data from five major technology companies—AAPL, MSFT, GOOG, AMZN, and META. The research was structured around two key stages: return prediction using various machine learning regression models, and subsequent portfolio optimization through three different techniques—SLSQP, Particle Swarm Optimization (PSO), and Differential Evolution Algorithm (DE). The results revealed that the performance of forecasting models varied across data frequencies. Notably, as data frequency decreased, all models experienced a decline in prediction accuracy, highlighting the trade-off between granularity and forecast reliability. However, the increase in prediction accuracy did not necessarily translate into superior portfolio outcomes. Among the portfolio optimization methods evaluated, all subjects to consistent weight constraints (minimum 0.05, maximum 0.8) to ensure comparability, Particle Swarm Optimization (PSO) emerged as the consistently best-performing technique, delivering the highest Sharpe ratios across all data frequencies. Back testing results demonstrated that portfolios optimized using weekly and monthly frequency data outperformed the equal-weighted benchmark in terms of risk-adjusted returns. In contrast, portfolios optimized on daily frequency data underperformed despite benefiting from more accurate predictive models. These findings suggest that while high-frequency data improves the precision of machine learning forecasts, portfolio performance depends more critically on the stability and strength of predictive signals, which tend to be more pronounced at moderate (weekly) or lower (monthly) frequencies. Consequently, the study concludes that weekly or monthly horizons strike a more effective balance between predictive accuracy and investment performance, offering a meaningful improvement over a naïve equal-weighted approach. Overall, this research underscores the value of integrating machine learning with heuristic optimization techniques in portfolio construction, while emphasizing the importance of carefully selecting data frequency to optimize real-world investment outcomes.

The findings of this study offer practical insights for investors, asset managers, and financial analysts interested in enhancing portfolio performance using machine learning. One key implication is that machine learning-generated return forecasts, particularly when integrated with Particle Swarm Optimization (PSO), can provide an effective foundation for constructing risk-adjusted portfolios, especially in medium- and lower-frequency settings such as weekly and monthly horizons. This suggests that ML-enhanced models may be more suitable for those medium-term investors or institutional strategies where daily adjustments are less feasible due to transaction costs or operational constraints. Moreover, the superior performance of PSO across all frequencies underscores its robustness in navigating complex search spaces and balancing return and risk effectively. Portfolio managers may consider incorporating PSO or similar population-based optimizers, such as other heuristic algorithms, into their allocation frameworks, especially when seeking to avoid local minima in traditional gradient-based methods. The underperformance of optimized portfolios at the daily level highlights the risk of overfitting and the potential unreliability of high-frequency return predictions in practical use when doing ML-embedded portfolio optimization. This points to the need for caution when applying ML models in short-term trading strategies without additional safeguards or ensemble mechanisms.

Future researchers can expand asset coverage beyond technology stocks to stimulate a more realistic and practical portfolio. The current back testing relied on a relatively short historical window. Extending the back testing horizon and implementing walk-forward validation would help assess the stability and reliability of model performance across different market regimes. Although weekly and monthly PSO-optimized portfolios outperformed the equal-weighted benchmark, it is unclear whether this advantage is structural or random. Future studies should statistically test the significance of frequency-dependent outperformance using larger samples and different time periods to confirm whether lower-frequency rebalancing genuinely enhances performance. Also, daily rebalancing may seem attractive in theory but becomes less viable when transaction costs and market frictions are accounted for. Incorporating these factors into optimization frameworks would yield more realistic and implementable strategies. Last but not least, while Sharpe ratio and error metrics (MAE/MSE) were useful in this study, future work should consider drawdown, Value-at-Risk (VaR), Sortino ratio, or tail risk measures to better capture risk dimensions that matter to practitioners.

References

Abolmakarem, S., Abdi, F., Khalili-Damghani, K., & Didehkhani, H. (2024). A multi-stage machine learning approach for stock price prediction: Engineered and derivative indices. *Intelligent Systems With Applications*, *24*, 200449. https://doi.org/10.1016/j.iswa.2024.200449

Abtahi, S. H. (2023). Uncertain random portfolio optimization based on skew chance distribution. *International Journal of Fuzzy Logic and Intelligent Systems*, *23*(1), 44–55. https://doi.org/10.5391/ijfis.2023.23.1.44

Alemany, N., Aragó, V., & Salvador, E. (2020). Optimal beats naive diversification: Asset allocation using high-frequency data. *The Journal of Portfolio Management*, *47*(1), 51–74. https://doi.org/10.3905/jpm.2020.1.185

Andreou, E., Ghysels, E., & Kourtellos, A. (2013). Should macroeconomic forecasters use daily financial data and how? *Journal of Business and Economic Statistics*, *31*(2), 240–251. https://doi.org/10.1080/07350015.2013.767199

Bhasin, N. K., Kadyan, S., Santosh, K., Hp, R., Changala, R., & Bala, B. K. (2024). Enhancing quantum machine learning algorithms for optimized financial portfolio management. In *Proceedings of the 2024 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS 2024)*. IEEE. https://doi.org/10.1109/INCOS59338.2024.10527612

Bulagang, A. F., Weng, N. G., Mountstephens, J., & Teo, J. (2020). A review of recent approaches for emotion classification using electrocardiography and electrodermography signals. *Informatics in Medicine Unlocked*, *20*, 100363. https://doi.org/10.1016/j.imu.2020.100363

Chan, K. Y., Abu-Salih, B., Qaddoura, R., Al-Zoubi, A. M., Palade, V., Pham, D., Del Ser, J., & Muhammad, K. (2023). Deep neural networks in the cloud: Review, applications, challenges and research directions. *Neurocomputing*, *545*, 126327. https://doi.org/10.1016/j.neucom.2023.126327

Chen, J. (2020, November 25). *Market portfolio: Definition, theory, and examples*. Investopedia. https://www.investopedia.com/terms/m/market-portfolio.asp

Chen, J. (2021, September 16). *Equal weight investing: Performance and examples*. Investopedia. https://www.investopedia.com/terms/e/equalweight.asp

Chen, J. (2024, April 11). *Backtesting: Definition, how it works, and downsides*. Investopedia. https://www.investopedia.com/terms/b/backtesting.asp

Chen, R.-R., Huang, W. K., & Yeh, S.-K. (2021). Particle swarm optimization approach to portfolio construction. *Intelligent Systems in Accounting, Finance and Management*, *28*(3), 182–194. https://doi.org/10.1002/isaf.1498

Chen, W., Zhang, H., Mehlawat, M. K., & Jia, L. (2020). Mean–variance portfolio optimization using machine learning-based stock price prediction. *Applied Soft Computing*, *100*, 106943. https://doi.org/10.1016/j.asoc.2020.106943

Comito, C., & Pizzuti, C. (2022). Artificial intelligence for forecasting and diagnosing COVID-19 pandemic: A focused review. *Artificial Intelligence in Medicine*, *128*, 102286. https://doi.org/10.1016/j.artmed.2022.102286

Corporate Finance Institute. (2025). *Modern portfolio theory (MPT)*. https://corporatefinanceinstitute.com/resources/career-map/sell-side/capital-markets/modern-portfolio-theory-mpt/

Croome, S. (2025). *How to achieve optimal asset allocation*. Investopedia. https://www.investopedia.com/managing-wealth/achieve-optimal-asset-allocation/

Díez-Fernández, M., Teleña, S. A., & Gorse, D. (2012). Construction of emerging markets exchange traded funds using multiobjective particle swarm optimisation. In C. A. Coello Coello, V. Cutello, B. Derbel, S. Forrest, G. Nicosia, & M. Pavone (Eds.), *Lecture Notes in Computer Science: Vol. 7553. Parallel problem solving from nature – PPSN XII* (Part II, pp. 140–147). Springer. https://doi.org/10.1007/978-3-642-33266-1_18

Fahmy, H. (2019). Mean-variance-time: An extension of Markowitz's mean-variance portfolio theory. *Journal of Economics and Business*, *109*, 105888. https://doi.org/10.1016/j.jeconbus.2019.105888

Fortin, F.-A., De Rainville, F.-M., Gardner, M.-A., Parizeau, M., & Gagné, C. (2012). DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, *13*, 2171–2175. http://www.jmlr.org/papers/volume13/fortin12a/fortin12a.pdf

França, R. P., Monteiro, A. C. B., Arthur, R., & Iano, Y. (2021). An overview of deep learning in big data, image, and signal processing in the modern digital age. In *Trends in deep learning methodologies* (pp. 63–87). Elsevier. https://doi.org/10.1016/b978-0-12-822226-3.00003-9

Galvão, A. B., & Owyang, M. (2022). Forecasting low-frequency macroeconomic events with high-frequency data. *Journal of Applied Econometrics*, *37*(7), 1314–1333. https://doi.org/10.1002/jae.2931

Garg, A., Singh, A. P., Saraswat, A., Kumar, A., & Taluja, A. (2025). Optimisation techniques in investment risk management: Scipy, SLSQP and modern portfolio theory. In *Proceedings of the 2025 3rd International Conference on Intelligent Systems, Advanced Computing, and Communication (ISACC 2025)* (pp. 1329–1334). IEEE. https://doi.org/10.1109/ISACC65211.2025.10969208

Glova, A. M. G., & Barrios, E. B. (2024). Modelling mixed-frequency time series with structural change. *Computational Economics*. https://doi.org/10.1007/s10614-024-10672-8

Gupta, S., Nachappa, S., & Paramanandham, N. (2025). Stock market time series forecasting using comparative machine learning algorithms. *Procedia Computer Science*, *252*, 893–904. https://doi.org/10.1016/j.procs.2025.01.050

He, T. (2024). Stock returns prediction: Comparison between ARIMA and support vector regression. In *Proceedings of the 2024 International Conference on Applied Mathematics, Statistics and Computer Science* (pp. 371–376). CRC Press. https://doi.org/10.1201/9781003508816-56

Indeed Editorial Team. (2025). *Experimental research: Definition, types and examples*. Indeed Career Guide. https://www.indeed.com/career-advice/career-development/experimental-research

Kao, Y., & Chen, Y.-C. (2013). A differential evolution approach to portfolio optimization problems. In *Proceedings of the 19th International Conference on Soft Computing: Evolutionary Computation, Genetic Programming, Swarm Intelligence, Fuzzy Logic, Neural Networks, Fractals, Bayesian Methods (MENDEL 2013)* (pp. 99–104). Brno University of Technology.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95 – International Conference on Neural Networks* (Vol. 4, pp. 1942–1948). IEEE. https://doi.org/10.1109/ICNN.1995.488968

Kraft, D. (1988). *A software package for sequential quadratic programming* (DFVLR-FB 88-28). Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt.

Liu, M., & Liu, Z. (2024). Research on financial service data based on neural network. In *Smart innovation, systems and technologies* (pp. 609–618). Springer. https://doi.org/10.1007/978-981-97-3210-4_50

Ma, Y., Gao, X., Liu, C., & Li, J. (2024). Improved SQP and SLSQP algorithms for feasible path-based process optimisation. *Computers & Chemical Engineering*, *189*, 108751. https://doi.org/10.1016/j.compchemeng.2024.108751

McDonald, S., Coleman, S., McGinnity, T. M., Li, Y., & Belatreche, A. (2014). A comparison of forecasting approaches for capital markets. In *Proceedings of the 2014 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr)* (pp. 92–99). IEEE. https://doi.org/10.1109/CIFEr.2014.6924062

Mercangöz, B. A. (2021). Portfolio optimization. In B. A. Mercangöz (Ed.), *International series in operations research and management science: Vol. 306. Applying particle swarm optimization* (pp. 15–27). Springer. https://doi.org/10.1007/978-3-030-70281-6_2

Mesut, B., Başkor, A., & Aksu, N. B. (2023). Role of artificial intelligence in quality profiling and optimization of drug products. In *Artificial intelligence in pharmaceutical sciences* (pp. 35–54). Elsevier. https://doi.org/10.1016/b978-0-323-89925-3.00003-4

Miller, A., Panneerselvam, J., & Liu, L. (2021). A review of regression and classification techniques for analysis of common and rare variants and gene-environmental factors. *Neurocomputing*, *489*, 466–485. https://doi.org/10.1016/j.neucom.2021.08.150

Pinelis, M., & Ruppert, D. (2021). Machine learning portfolio allocation. *The Journal of Finance and Data Science*, *8*, 35–54. https://doi.org/10.1016/j.jfds.2021.12.001

Qian, Y., & Zhang, Y. (2025). Long-term forecasting in asset pricing: Machine learning models' sensitivity to macroeconomic shifts and firm-specific factors. *The North American Journal of Economics and Finance*, *77*, 102423. https://doi.org/10.1016/j.najef.2025.102423

Rossini, R., Poccia, S., Candan, K. S., & Sapino, M. L. (2019). CA-smooth: Content adaptive smoothing of time series leveraging locally salient temporal features. In *Proceedings of the 11th International Conference on Management of Digital EcoSystems (MEDES 2019)* (pp. 36–43). ACM. https://doi.org/10.1145/3297662.3365830

Tütüncü, R. H. (2011). Recent advances in portfolio optimization. In J. R. Birge & V. Linetsky (Eds.), *Handbooks in operations research and management science: Vol. 15. Financial engineering* (pp. 33–72). Elsevier. https://doi.org/10.1093/oxfordhb/9780199553433.013.0002

Xia, G., Bi, Y., & Wang, C. (2023). Optimization design of passive residual heat removal system based on improved genetic algorithm. *Annals of Nuclear Energy*, *189*, 109859. https://doi.org/10.1016/j.anucene.2023.109859

Yourdkhani, S. (2014). Portfolio management by using value at risk (VaR): A comparison between particle swarm optimization and genetic algorithms. *Life Science Journal*, *11*(1 Spec. Issue), 15–26. https://doi.org/10.7537/marslsj1101s14.03

Zhao, Z., Alzubaidi, L., Zhang, J., Duan, Y., & Gu, Y. (2023). A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations. *Expert Systems With Applications*, *242*, 122807. https://doi.org/10.1016/j.eswa.2023.122807