ORIE 5370 Optimization Modeling in Finance

Course Project

An Analysis of LSTM Predictive Model with Traditional Optimization Models in Stock Portfolio optimization

May 2024

Songyan Xu (sx274), Yuxuan Wang(yw2592), Ruihan Dou (rd593)

Github: <https://github.com/Howl101/ORIE_5370_Project>

We extend our heartfelt thanks to Professor James Renegar and Teaching Assistant George Yu for their invaluable guidance and support throughout this project.

1. Introduction

Portfolio optimization is critical for maximizing returns while managing investment risk effectively. Traditionally, this optimization process has involved techniques such as the Mean-Variance optimization, which relies on balancing portfolios based on expected returns and the covariance of assets, as outlined in the ORIE 5370 class. Additionally, strategies like Maximum Diversification have been developed to enhance portfolio resilience by maximizing diversification across stocks.

Recently, advancements in computational capacity and artificial intelligence have ushered in a new era of portfolio management. Machine learning models, such as LSTM networks a specialized form of Recurrent Neural Networks designed to mitigate the Gradient Vanishing problem—have been employed to refine return predictions and optimize portfolio strategies.

The content of this report is structured into two main sections. The first section explores traditional portfolio optimization strategies, applying Mean-Variance and Maximum Diversification techniques directly to the portfolio. This approach follows a classical framework that prioritizes the structural properties of the portfolio without the influence of predictive models.

Following the examination of traditional methods, the report transitions to a modern approach involving an LSTM model. This model is leveraged to predict future stock returns and optimize the portfolio's Sharpe ratio. By predicting returns, the LSTM model facilitates the determination of optimal asset weights, aiming to maximize the efficiency of the portfolio under predicted future conditions.

Contrasting with the approach by Zhang, Zohren, and Roberts in *Deep Learning for Portfolio Optimization* (2020), which streamlines portfolio optimization by directly optimizing the Sharpe ratio of ETFs without a forecasting step, our methodology adopts a two-step process. This involves predictive modeling followed by optimization techniques, allowing for an in-depth evaluation of each methodology's effectiveness under varying market conditions. By examining these strategies both in isolation and in conjunction, the report aims to illustrate the potential advantages and disadvantages of integrating advanced predictive models with traditional financial portfolio management practices.

2. Data Overview

In this report, we used eight specific stocks from Nancy Pelosi's publicly disclosed portfolio. These stocks are Apple (AAPL), AllianceBernstein Holding LP (AB), Disney (DIS), Google (GOOG), Microsoft (MSFT), Nvidia (NVDA), PayPal (PYPL), and Tesla (TSLA). The data we used covers the daily adjusted price from July 8, 2015 (the date when eBay spun off PayPal into its own publicly traded company) to April 12, 2024 (the date when this report was initiated). All of the data in the report are downloaded from Yahoo Finance, which provides reliable information. The strategic selection of these stocks not only aligns the project with public interest but also allows an in-depth analysis of investment performance that is associated with a prominent public figure.

In choosing our portfolios, we used individual securities instead of the broad-based ETFs to capture the distinct volatility and growth patterns of the individual companies. This method challenges the robustness of our model because our portfolio includes stock with nuanced behaviors observed across different market sectors from technology to automobiles. Furthermore, to ensure accuracy in our return calculations, we used adjusted closing prices rather than closing prices. Adjusted prices account for external factors such as stock splits, and dividends. It provides a true reflection of stock value over time and enhances the reliability of our financial analysis. Finally, we calculated the return based on the adjusted close price and combined all data in a single data frame. The following figure (Figure 1. Full-sized Figures 1-20 can be found in the Appendix before References.) is a brief look at our dataset which has 2207 rows and 16 columns (8 columns for returns and 8 columns for prices).

Date	AAPL_Return	AB_Return	DIS_Return	GOOG_Return	MSFT_Return	NVDA_Return	PYPL_Return	TSLA_Return	CASH_Return	AAPL	AB	DIS	GOOG
2015-07-08	-0.024823	-0.015432	-0.016311	-0.015599	-0.001354	-0.007074	-0.052430	-0.048231	0.002	27.617405	13.559929	107.632240	25.841499
2015-07-09	-0.020396	0.000348	0.003559	0.007449	0.006329	-0.012214	-0.005764	0.011610	0.002	27.054110	13.564651	108.015335	26.034000
2015-07-10	0.026734	0.006616	0.007267	0.018149	0.002022	0.017517	0.005507	0.004769	0.002	27.777388	13.654389	108.800232	26.506500
2015-07-13	0.019305	-0.002421	0.013827	0.030973	0.020847	0.006582	0.060248	0.011615	0.002	28.313641	13.621328	110.304596	27.327499
2015-07-14	-0.000398	-0.021151	-0.001694	0.026622	0.001757	0.000503	0.004078	0.013312	0.002	28.302372	13.333221	110.117714	28.055000

Figure 1. Dataset Overview

Figure 2 below shows the change in price for those 8 stocks from 2015 to 2024. Most of the stocks perform quite well. We can explain some trends based on historical events. For example, Nvidia and Microsoft skyrocketed between 2023 and 2024 due to the rise of Artificial Intelligence and generative AI. Most of the stocks in the portfolio have a downturn between 2022 and 2023 due to the rising Interest Rate in the USA. The remaining part of the report will focus on how Machine Learning and Optimization will help with the portfolio weight allocation process.

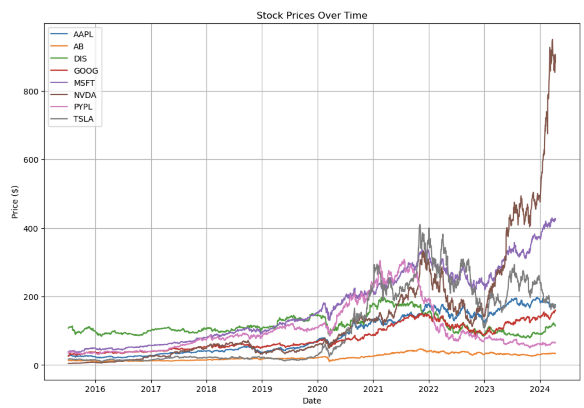


Figure 2. Stock Prices Movements, 2015-2024

3. The Mean-Variance Method

3.1 Assumption of Mean-Variance

Our first choice is the Markowitz Mean-Variance Portfolio Optimization. We want to clarify our model assumptions for the mean-variance portfolio before setting up the optimization problem:

- No short selling

Although short selling plays a significant role as a risk hedge, the objective function to optimize is more inclined to be non-convex when there are negative stock weights in the portfolio. Specifically, if the objective function is not a quadratic program, i.e., the objective function is not minimizing risk only, the objective function will be non-convex. The objective function of the mean-variance portfolio is minimizing risk only, but the objective function of the max diversification portfolio in the next part is not. Although it is feasible to transform the objective function or add regularization to deal with the non-convexity, this approach can be complicated and unstable. We make this assumption to make it easier to compare between these two strategies.

- 20-day (Monthly) rebalancing period

Considering the high transaction fees in real-world trading, it is inappropriate to re-balance the portfolio too frequently. On the other hand, the rebalancing period cannot be too lengthy to ensure the model is fully ex-posed to the latest market data. To achieve a balance, we set a 20-day rebalancing period.

- The 20% weight cap for a single stock

This can avoid putting all the money into one or two stocks, thereby impacting the diversification of the portfolio.

- We do not always require full exposure to the market

Normally, we want the portfolio to be fully invested in the market. However, when gett-ing a higher return than the target return (we set it as a positive parameter) is impossible, we want to hold cash to avoid significant loss instead of putting all the capital in the market.

- No bank.

In reality, when we hold cash, we can deposit money to earn risk-free int- erest from the bank. For simplicity, in this model, we assume there is no bank, and the risk-free rate is zero. In this situation, when it is impossible for the portfolio return to exceed the target return, we will hold cash until the next rebalancing period. Note that adding a bank or a positive risk-free rate will always improve the portfolio performance.

- No tax or brokerage fees

This is just for simplicity. When considering commission and capital gains tax, things get extremely complicated. However, it is worth investigating their influence if we want to execute our portfolio strategies in real-world scenarios.

- All securities are infinitely divisible

Since we want the portfolio to be fully in-vested in the market in normal situations, we require the total sum of all weights to equal 1. Thus, holding integer stock shares is not feasible. This assumption ensures that all available funds are fully utilized.

3.2 Objective Function

Based on the above six assumptions, we write the below optimization problem:

Goal: Minimize risk subject to achieving target return

$$\min_x x^T V x$$

$$s.t.\ u^T x \geq R$$

$$e^T x = 1$$

$$0 \leq x \leq 0.2$$

We want to clarify the notations here.

$x = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8)$ is the weight vector for eight stocks, V is the covariance matrix;

$\mu = (\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6, \mu_7, \mu_8)$ is the expected return of eight equities;

R is the target return;

$e = (1, 1, 1, 1, 1, 1, 1, 1)$.

The first constraint ensures the portfolio return exceeds the target return, and the second constraint ensures the sum of all weights equals 1.

But how do we define the expected return and the covariance matrix? Our initial idea was to reference the approach described in Mr. Renegar's course note. Suppose μ_i is the expected return of equity i, and σ_{ij} is the covariance of returns between assets i and j, where i = 1, 2, …, 8, j = 1, 2, …, 8. According to the course note, the simplest (and obviously not the best) choices for μ_i and σ_{ij} are based solely on historical data. Specifically, given historical returns r_{it} for equity i and all days $t = 1, \ldots, 20$ in one rebalancing period, we have

$\overline{r}_i = \frac{1}{20}\sum_{t=1}^{20} r_{it}$ as the average returns for eight equities. Thus, μ_i is defined as $\overline{r}_i$, and σ_{ij} is defined as $\frac{1}{20}\sum_{t=1}^{20} (r_{it} - \overline{r}_i)(r_{jt} - \overline{r}_j)$. In this way, the expected returns are defined as $\overline{r}_i$ and the (i, j) coefficient in the covariance matrix is σ_{ij}.

We also propose a slightly different way to define the expected returns and the covariance matrix. Instead of directly using the average returns for eight equities, we use the compound return within one rebalancing period as the expected returns. Moreover, instead of using the covariance among stock returns within one rebalancing period, the covariance matrix is defined as the covariance among stock returns from the start date of the dataset to the current rebalancing date. A covariance matrix derived from a longer time can offer a more stable covariance reflecting different market conditions. We are curious whether adopting different expected returns and a covariance matrix can impact the portfolio performance.

3.3 Mean Variance Result

Combining two different expected returns and two different covariance matrices in pairs, we obtained four results by minimizing the portfolio's variance ($x^T V x$) subject to achieving the target return. Starting from 2015-07-08, we use the next 20 days to determine the optimal weights. With these optimal weights, we allocate capital to eight stocks and hold them for the next 21 to 40 days. We update the portfolio value and find the optimal weights again in the next 40th day.

Following the above process, we have the four portfolio values in the end:

Portfolio	Additive Return + Covariance Covariance Matrix within the Rebalancing Period	Compound Return + Covariance Matrix within the Rebalancing Period	Additive Return + Covariance Covariance Matrix from the Start Date	Compound Return + Covariance Matrix from the Start Date
Portfolio Ending Value	$321,619.84	$366,519.53	$259,079.14	$299,618.86

We conclude that the second portfolio generated the highest portfolio value in the end, and considering the covariance of returns from the start date of the dataset will negatively impact the portfolio performance.

From Figure 3, we can see that no matter what definitions of expected returns and covariance matrices we use, a large drawdown exists from mid-2021 to 2023 in all four portfolios. This is a market event rather than the failure of our model. According to *Lauricella and Solberg (2022)*, investors focused on companies that could thrive in the lockdown during the pandemic, which led to the overvaluation of tech stocks. After the pandemic, the increased interest rate and the economic transition resulted in a price correction in tech stocks.

Although all four portfolios experienced this drawdown after the pandemic, the drawdown is less significant when using the compound returns as the expected returns. This is probably due to several reasons.

Firstly, the compound return reflects actual investment performance over time, because it captures the cumulative effect of gains and losses. On the other hand, the mean of returns may overestimate or underestimate the actual growth of the investment.

Secondly, the compound return is sensitive to the sequence of daily returns, which reflects the impact of market volatility, while the average return ignores this impact.

Figure 3. Portfolio Value Over Time in Four Different Mean-Variance Portfolios

4.5 Optimal Weight

In the end, we want to examine the optimal weights of eight stocks over time. We examine Case 2's optimal weights, as it generates the highest ending portfolio value. From Figure 4, we can see that for AAPL, AB, DIS, GOOG, and MSFT, their optimal weights at rebalancing dates are mostly either the upper limit (20%) or the lower limit (0%). This is worth further investigation as it seems the weight cap has negatively impacted the portfolio performance by limiting investing in winners. Note that the optimal weights in all four portfolios have this pattern. However, increasing the upper limit of the weight for individual stocks does not seem to improve the portfolio's performance, as the portfolio is more easily affected by individual stocks, thus going against the principle of diversification. Therefore, we must carefully choose the upper limit of weights to take a balance between diversification and portfolio performance.

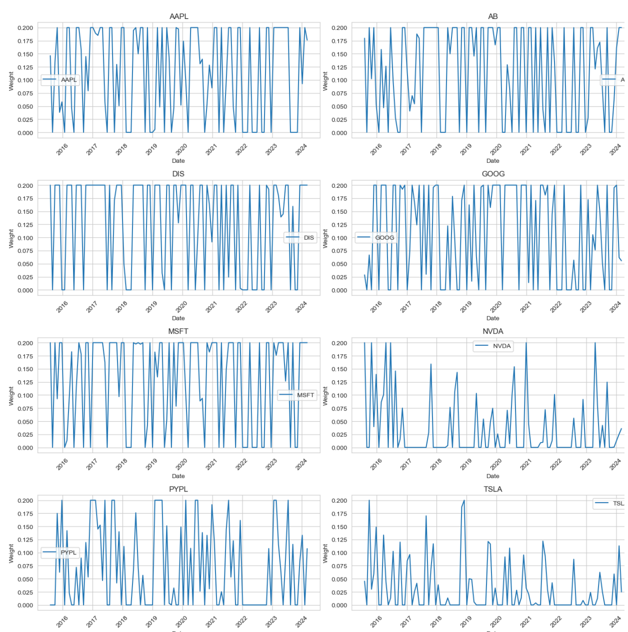


Figure 4. Optimal Weights for Eight Stocks Over Time

4. Max Diversification

4.1 Background of Max Diversification

Max Diversification Portfolio comes from the idea of maximizing the diversification ratio. As proposed by Choueifaty (2008), the diversification ratio is defined as $D = \frac{x^T\Sigma}{\sqrt{x^T V x}}$, where $\Sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6, \sigma_7, \sigma_8)$ is the vector of asset volatilities. Choueifaty (2008) suggested that the advantages of the max diversification portfolio have the following advantages over the mean-variance portfolio:

- The max diversification portfolio is better diversified across various risks.
- The max diversification portfolio minimizes idiosyncratic risk more effectively than the mean-variance optimization, which is particularly helpful in volatile markets.
- The success of mean-variance optimization heavily depends on the accurate estimation of expected returns, which can be challenging. However, the max-diversification optimization relies on the estimation of volatilities and covariance among assets, which are generally more stable than estimating expected returns.

4.2 Objection Function

Based on the seven assumptions listed in the previous part, we want to maximize the portfolio's diversification ratio. However, as the objective function is not a standard quadratic program, it is no longer convex, which makes optimization using the cvxpy package difficult. Therefore, we must transform the objective function and adjust our assumptions for this strategy.

We refer to the IEOR 4500 course notes at the University of South Carolina (n.d.) to address this problem. The course note describes a way to transform the Sharpe Ratio Maximization problem given a long-only portfolio:

Original Problem:

$$\text{maximize} \frac{\mu^T x - r_f}{\sqrt{x^T V x}}$$

$$\text{s.t. } e^T x = 1$$

$$Ax \geq b$$

$$x \geq 0$$

Transformed Problem:

$$\text{minimize } y^T V y$$

$$\text{s.t. } \hat{\mu}_t y = 1$$

$$\hat{A} y \geq 0$$

$$y \geq 0$$

In the transformed problem, we define

$\hat{\mu} = \mu^T x - r_f$, $\hat{A}$ be the matrix whose i, j-entry is $a_{ij} - b_i$ in the original problem.

Note that the critical assumption for this transformation is there exists an x satisfying all constraints in the original problem such that $\mu^T x - r_f > 0$.

According to the course note from the *University of South Carolina (n.d.)*, if $\bar{y}$ is the optimal solution for the transformed problem, then $\bar{x} = \frac{\bar{y}}{\sum_j \bar{y}_j}$ is the optimal solution for the original problem. By solving the transformed problem to get the optimal $\bar{y}$ we can derive the optimal solution $\bar{x}$ for the original problem. Setting $\hat{\mu} = \Sigma$ and disregarding the constraint $Ax \geq b$, we can see that maximizing the diversification ratio is essentially the above original problem. Following the above procedures, we have the optimal $\bar{y}$ and $\bar{x}$. Note that $\bar{x}$ is the optimal weights for eight stocks.

Regarding the assumptions we made, the only difference from the mean-variance portfolio is that we cannot set an upper limit on x. As the cvxpy package solves the transformed problem, we cannot constrain what values x can take directly. Luckily, it is still uncommon for a single stock to have over 50% of funds invested in it.

4.3 Maximum Diversification Result

Similar to the mean-variance optimization, we want to compare the impact of using different covariance matrices (max diversification does not involve the expected returns). Figure 5 shows that although two portfolios experienced similar drawdowns at the same level, the portfolio using a covariance matrix derived from a longer time generates a higher ending value.

Specifically, the first portfolio has an ending value of $566,542.64, and the second portfolio has an ending value of $701,607.80. We notice that, in general, the max diversification portfolios perform better than the mean-variance portfolios. This corroborates Choueifaty's view on the advantages of max diversification optimization.

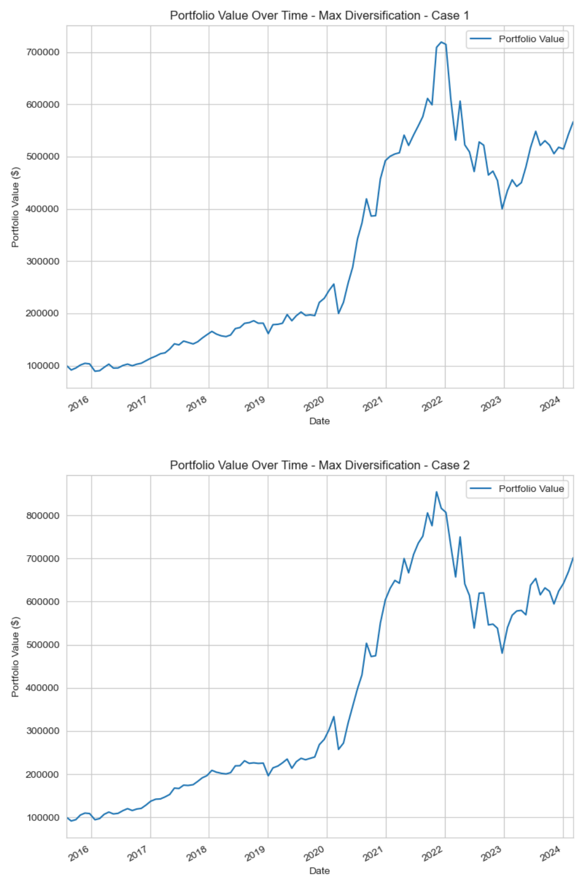


Figure 5. Portfolio Value Over Time in Four Different Max Diversification Portfolios

Besides, Figure 6 and Figure 7 below prove that using a covariance matrix derived from a longer period can provide more stability over time. The transition from Figure 6 to Figure 7 showcases a less volatile strategy. Moreover, extreme spikes in optimal weights in the first portfolio can indicate high turnover and associated costs in real-world trading. Therefore, we prefer the second portfolio (Case 2) in the max diversification.

Figure 6. Optimal Weights for Eight Stocks - Max Diversification - 20-day Covariance

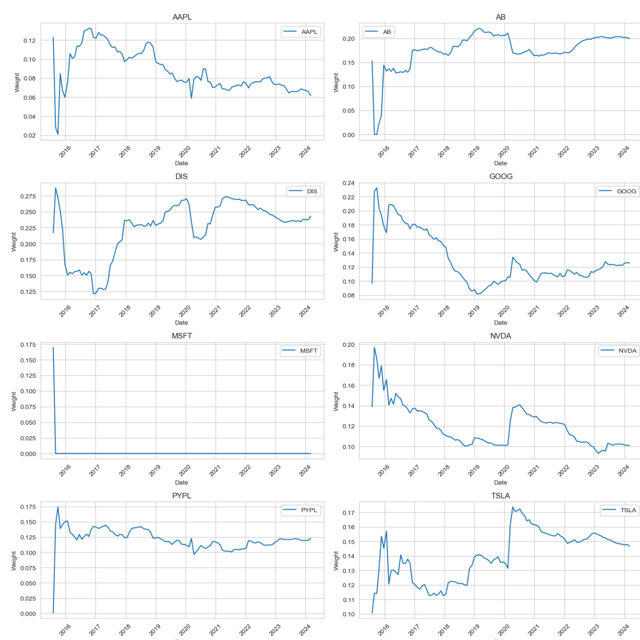


Figure 7. Optimal Weights for Eight Stocks - Max Diversification - All History Covariance

5. LSTM Model

5.1. Background of LSTM

Following the traditional MV and MD models, we want to further explore the use of LSTM (Long Short-Term Memory) in our project. In this section, we first use the LSTM model to predict future stock returns. Following this, we employ a Sharpe ratio optimization to determine optimal portfolio weights. This two-step approach is different from the approach in *Zhang, Z., Zohren, S., & Roberts, S. (2021) Deep Learning for Portfolio Optimization* where prediction and optimization are integrated. However, we believe separating prediction and optimization processes allows for more nuanced adjustments to each process and enhance the reliability of the model by preventing the optimization considerations from skewing the LSTM results.

LSTM is a specialized form of RNNs (Recurrent Neural Networks) specifically designed to address the Gradient Vanishing problem by using a series of gates that regulate information flow. The model has 3 total gates as shown in Figure 8 below:

Forget Gate: decide which information to discard from the cell state.

Input Gate: updates the cell state with new inputs.

Output Gate: determines what to transfer to the output.

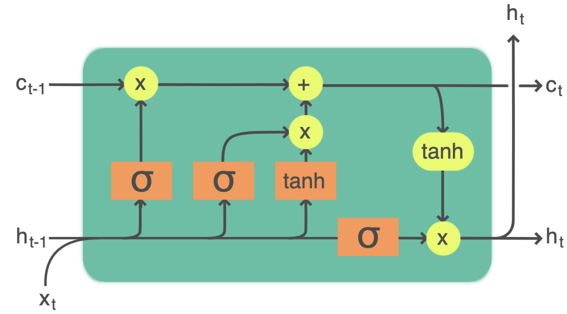


Figure 8. Model Gates

These gates allow the LSTM to selectively remember and forget information through a complex mechanism of weights and activation functions, making it highly effective for applications like language modeling and financial forecasting.

5.2 Data Preprocessing

Before proceeding with model training, it is necessary to split the dataset into training and testing sets. This step is essential to ensure that our model can generalize well to new, unseen data, thus providing a reliable measure of its performance. In our case, we have designated 80% of the data as the training set and the remaining 20% as the testing set. For the later LSTM model, this 20% data, starting July 2022 and ending March 2024, will be used as the observed period for result evaluation.

This method of splitting allows for a substantial amount of data to be used for training the model, ensuring that it learns comprehensive patterns and dynamics present in the dataset, while also retaining a significant portion of the data for testing its predictive capability.

Finally, we structure the input data into a format suitable for LSTM. We create a sequence consisting of 60 consecutive data points used as feature X, and the data point immediately following the sequence used as the target y. Since in our assumption, a month consists of 20 trading days, a season with three months will include 60 days. We believe the choice of 3 months allows the model to capture enough information on prices and stock movements, without over-fitting to the short-term fluctuations. This process is repeated through the entire data-set, sliding one position at a time, which ensures that the model learns from the entire history of 60 days to predict the next day's value. This input structure is consistent with the format required in LSTM and allows the model to recognize patterns over time.

With these preparations complete, our dataset is ready for the training phase, which follows immediately in the subsequent sections of the report. This structured approach to preparing our data ensures that our LSTM model has a strong foundation for learning and making accurate predictions.

5.3. Model architecture

Though return can be calculated from price, we use both return and price as our input to better capture different aspects of market dynamics. Additionally, we use a single-layer LSTM model to effectively balance computational efficiency and learning capacity. Following the LSTM layer, a dropout is incorporated to mitigate over-fitting. This layer randomly deactivated a proportion of the neurons during training, encouraging the model to develop more robust features and reliable predictions. The LSTM model concludes with a dense layer with 16 neurons, which is the same as the number of input dimensions and allows the effective translation of processed features into robust predictions.

5.4. Model Training

The loss function used in training the LSTM model is the mean squared error (MSE), which measures the average of the squares of the differences between predicted values and actual values. By minimizing MSE, the model is tuned to reduce prediction errors, which enhances the overall accuracy of the forecasts and the reliability of outcomes. Using a grid-search method, we eventually selected 90 neurons for the LSTM layer, and 0.2 for the dropout rate.

Additionally, an L2 regularization is incorporated into the model to help prevent overfitting. As we learned in class, L2 regularization works by adding a penalty (which is set to 0.01) on the size of the coefficients. This penalty term discourages the model from fitting the noise in the training data by keeping the model weights small, which promotes simpler model predictions.

The training process employs a batch size of 32 and runs for 30 epochs (where the loss function starts decreasing). A batch size of 32 allows for sufficient gradient estimates while maintaining a manageable computational load and training for 30 epochs ensures the model iteratively learns from the data, adjusting its weights to optimize the loss function without overfitting.

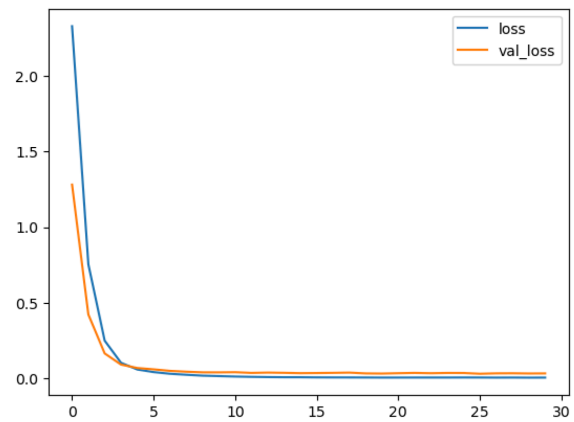


Figure 9. Loss Plot

5.5. LSTM Sharpe Optimization

Following the training and testing of our LSTM model, we proceed to employ the model's predictions for portfolio optimization. The primary objective here is to maximize the Sharpe ratio, a popular measure of the risk-adjusted return of a portfolio. The Sharpe ratio is calculated as the ratio of the portfolio's excess returns over the risk-free rate to its standard deviation. In our simulation, however, we simplify the calculation by assuming a risk-free rate of zero and focus solely on maximizing the returns relative to their volatility.

We initiate the optimization by defining the number of assets and setting initial weights evenly distributed across all assets. Constraints are also set to ensure all asset weights sum up to 100% of the portfolio. Asset bounds are defined to restrict the allocation in each asset, which is capped at 20%, which ensures the diversification of the portfolio. Portfolio rebalancing is conducted periodically every 20 days (a month) based on the model's predictions of future returns. The optimization process during each rebalancing period involves:

Prediction of Returns: The LSTM model predicts future returns for the period until the next rebalance.

Portfolio Rebalancing: Using these predictions, we apply the Sharpe ratio maximization technique to determine the optimal weights for the next period.

The optimization is carried out using the minimize function from *scipy.optimize*, specifically using the Sequential Least Squares Programming (SLSQP) method due to its suitability for constrained optimization problems.

The results from the optimization are stored and assessed at each rebalance date, with the model adjusting the portfolio according to the calculated optimal weights. This dynamic rebalancing helps in adapting to changes in market conditions and asset performance as predicted by the LSTM model.

At the end of our forecasting period, we consolidate the rebalance dates and the corresponding optimal weights into a *data frame*, which provides a clear and actionable output of how the portfolio should have been adjusted over time to maximize the Sharpe ratio.

5.6. LSTM Performances

This section presents an empirical evaluation of the portfolio optimized using an LSTM-based forecasting model. We conduct a backtest to simulate the performance of this optimized portfolio by using historical market data, thereby assessing how it would have performed under past market conditions. We then proposed several enhancements to add to the usability of the portfolio in practice.

5.6.1. Simple LSTM Model

We start with an initial capital, which is set to $100,000 for this simulation. The portfolio's initial value is based on the stock prices at the beginning of the test period.

At each rebalance date, the portfolio value is updated based on the new weights and current market prices. The calculation involves normalizing the portfolio's growth by its initial values to maintain proportionate scaling. The calculated value of the portfolio changes with respect to time are shown in Figure 10 below.

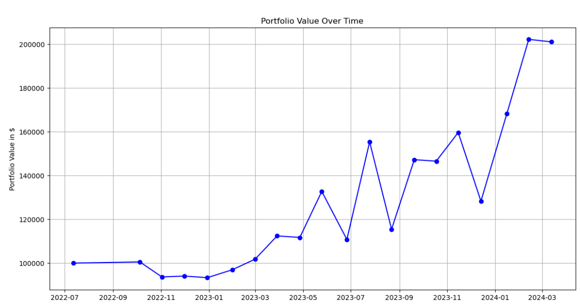


Figure 10. Simple LSTM - portfolio value over time

The backtest results indicate that the portfolio achieved a cumulative return of 100.94% over the observed period. Notably, despite experiencing significant fluctuations, the portfolio registered substantial gains across 18 months. The maximum drawdown recorded during this period was 25.67%. This high drawdown, which occurred primarily during the second to the fourth quarter of 2023, with the peak drawdown reaching 25.67% in August 2023, underscores a potential volatility concern that might influence fund managers' strategic decisions. Such a considerable draw-down may necessitate tactical adjustments to the investment approach, potentially leading to reducing or liquidating positions to avoid risks.

5.6.2. Equal-weighted Portfolio

To better evaluate the LSTM-based portfolio, we also established an equal-weighted portfolio as a benchmark to compare the performance of the two strategies. The equal-weighted portfolio distributes the initial capital equally across all 8 stocks, providing a straightforward benchmark in investment management. The initial capital was again set at $100,000, which is evenly distributed across all stocks. At each point in time, the value of the portfolio is just the latest market prices of the assets, proportionally adjusting for the changes in asset prices relative to their initial values.

The calculated value of the portfolio changes with respect to time are shown below in Figure 11.

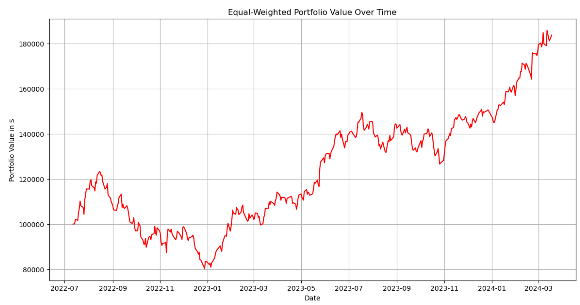


Figure 11. Equal-weighted - portfolio value over time

The equal-weighted portfolio yielded a cumulative return of 83.80%, with a maximum drawdown of 15.04%.

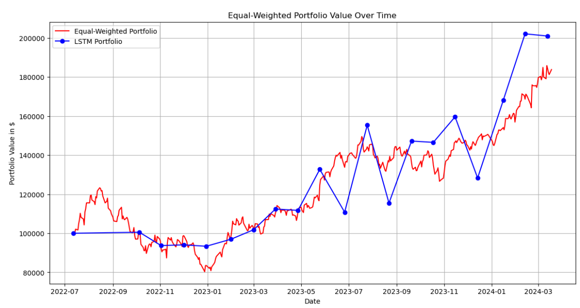


Figure 12. Comparison: Simple LSTM vs. Equal-weighted

Comparatively, as shown in Figure 12, the LSTM portfolio outperformed the equal-weighted portfolio by 17.14% in terms of cumulative return. Moreover, out of 18 rebalance points, the LSTM portfolio's value surpassed that of the equal-weighted portfolio 12 times, translating to a proportion of 66.7%. These results indicate a superior performance of the LSTM portfolio relative to the equal-weighted strategy.

However, it is necessary to acknowledge that the LSTM portfolio exhibited higher volatility, particularly during the turbulent quarters of 2023. This increased volatility, evidenced by the higher maximum drawdown, highlights the trade-offs between higher returns and potential risks associated with such advanced modeling techniques.

In response to the high volatility observed in the LSTM portfolio, a new mechanism involving the use of cash reserves will be introduced, which is intended to provide a buffer against market downturns and enhance the overall risk management of the portfolio.

5.6.3. LSTM Portfolio with cash

In an effort to address the volatility observed in the LSTM-based portfolio, our strategy incorporated an additional asset, cash, which is inherently less volatile. The cash asset is uniquely characterized by a constant price of one and a return equivalent to the Federal Interest Rate, adjusted to a monthly basis. Importantly, the allocation cap for cash was set at 100%, unlike other assets capped at 20%. This allows the portfolio to be fully liquidated into cash based on the model's recommendations, potentially safeguarding against adverse market conditions.

The portfolio was re-optimized using the same LSTM parameters previously used, but with the inclusion of cash as a new asset class. This approach aims to evaluate the impact of cash on the portfolio's risk-return profile, particularly focusing on periods of heightened market volatility.

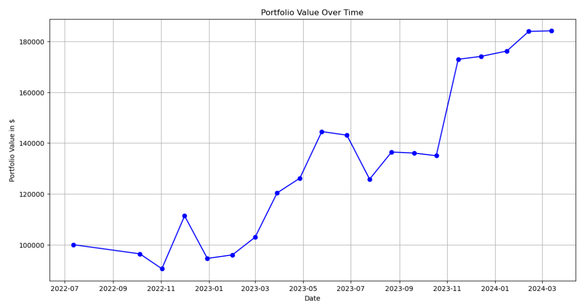


Figure 13. LSTM with cash - portfolio value over time

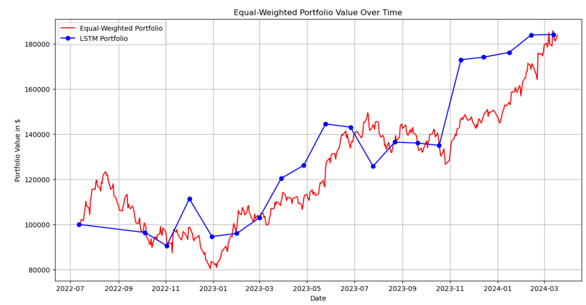


Figure 14. Comparison: LSTM with cash vs. Equal-weighted

As shown in Figure 13, the introduction of cash into the portfolio led to several notable outcomes. For the cumulative return, the portfolio achieved a cumulative return of 84.18% over the period. Although this represents a decrease of 16.76% from the original LSTM-optimized portfolio without cash, it still slightly surpasses the equal-weighted benchmark by 0.38%.

The maximum drawdown was substantially reduced to 15.03%, which is a 41% decrease from the original maximum drawdown of 25.67%. Notably, the maximum drawdown now slightly undercuts that of the equal-weighted portfolio by 0.1%, marking a significant improvement in risk management. Also, the volatility of the portfolio, particularly during the second to the fourth quarters of 2023, was markedly reduced. The stabilization is reflected in the reduced frequency and magnitude of value dips during this period, as illustrated in the accompanying comparison plot.

Figure 14 shows that the LSTM portfolio with cash not only remains consistently higher than the equal-weighted portfolio during most of the test period but also exhibits smoother fluctuations. This smoother performance curve underscores the effectiveness of cash in buffering against market downturns. Over the 18 rebalance points analyzed, the LSTM portfolio with cash outperformed the equal-weighted portfolio in 11 instances. This 61.1% superiority rate also shows the robustness of incorporating cash into the portfolio strategy.

5.6.3. Optimized for historically high

In an innovative approach to optimize portfolio performance, we integrated a strategy wherein the rebalancing mechanism is conditionally bypassed if the portfolio value reaches a new historical high. This method aims to sustain optimal asset allocations that have proven successful, thereby potentially enhancing returns by maintaining positions that are currently performing well.

At each scheduled rebalance point, the portfolio's current value is assessed against its historical peak values. If the current value equals or surpasses the previous highest value, the portfolio maintains its existing asset allocations instead of adjusting them. This strategy is predicated on the assumption that positions driving the portfolio to new heights should be retained as long as they continue to perform optimally.

5.6.4. Implementation with the Original LSTM Model

Initially, this strategy was employed using the simple LSTM model included only the eight stocks, shown below in Figure 15.

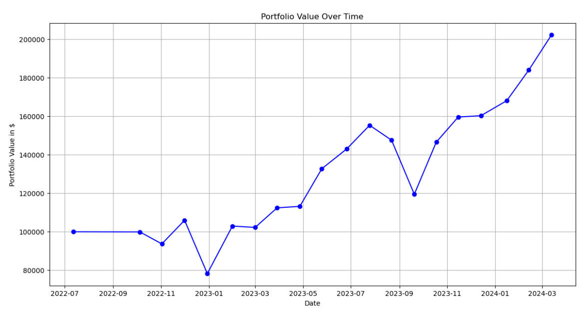


Figure 15. Enhanced LSTM - portfolio value over time

The portfolio achieved a cumulative return of 102.23%, indicating an improvement compared to the original return of 100.94%. The strategy resulted in a maximum drawdown of 26.14%, still occurring in August 2023, which is slightly higher than that of the original LSTM portfolio. Among the 18 rebalance points, the new LSTM portfolio outperformed the equal-weighted portfolio by 12 points, which is the same as the original portfolio. This shows that the strategy generally amplified the characteristics of the original portfolio, with slightly higher return and maximum drawdown, showing the tradeoff between return and risk.

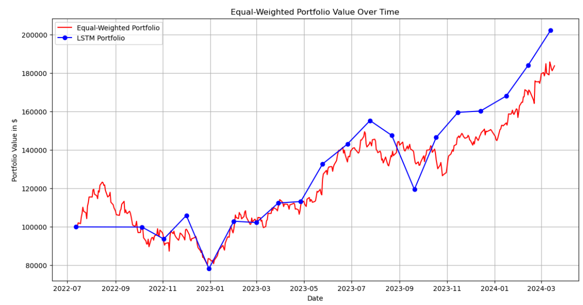


Figure 16. Comparison: Enhanced LSTM vs. Equal-weighted

5.6.5. Implementation with the Cash-added LSTM Model

Finally, we added this strategy to our cash-included model.

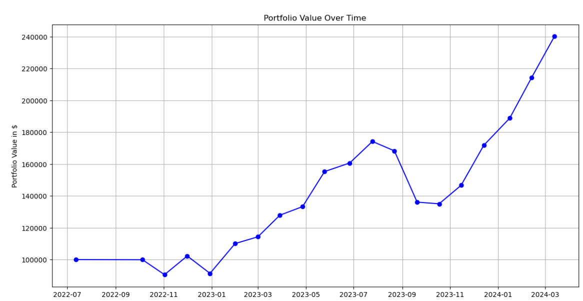


Figure 17. Enhanced LSTM with cash - portfolio value over time

The enhanced portfolio has markedly outperformed the equal-weighted and original cash-included LSTM portfolios in terms of cumulative return. Achieving a 140.12% return is a significant leap from the returns of the original models, indicating the successful exploitation of positive market trends and asset performance.

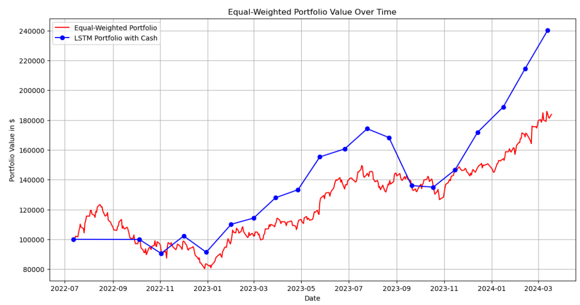


Figure 18. Comparison: Enhanced LSTM with cash vs. Equal-weighted

Although the maximum drawdown increased to 22.40%, we believe this is a reasonable increment in risk considering the substantial increase in return. This drawdown is significantly higher than the original models but reflects a more aggressive investment stance, which has paid off in terms of overall growth. Also, with 14 out of 18 rebalance points showing superior performance compared to the equal-weighted strategy, the new model not only offers higher returns but also more consistent outperformance. This suggests that the model's predictions and the strategy to hold at peaks are effectively aligned with market movements.

5.7. Performance Comparison

In the comparison of the performances of the four portfolios, we concluded that the enhanced LSTM portfolio with cash could be a more favorable one for several considerations.

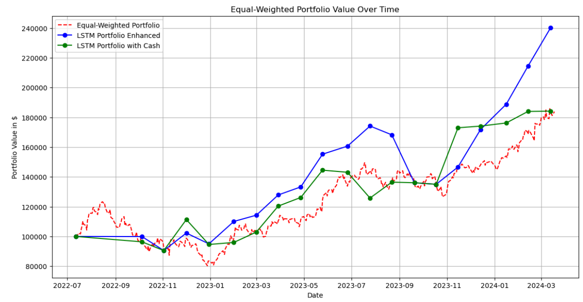


Figure 19. Comparison: LSTM with cash vs. Enhanced LSTM with cash vs. Equal-weighted

Below is a table comparing the results from the four LSTM portfolios, with C.R., M.D., and Prop. indicating the cumulative return, maximum drawdown, and proportion of the value at rebalancing points that outperforms the Rqual-weighted portfolio.

LSTM portfolios	C.R. (%)	M.D. (%)	Porp. (%)
Equal-weighted	83.80	15.04	-
Simple	100.94	25.67	66.7
+Cash	84.14	15.03	61.1%
Simple Enhanced	102.23	26.14	66.7
+ Cash Enhanced	140.12	22.40	77.8

LSTM portfolios	Performances
Simple	High return with high risk
+Cash	High stability with lower return
Simple Enhanced	Higher return with Higher risk
+ Cash Enhanced	Higher return with reasonable risk

- Optimized Returns

The strategy's ability to lock in gains at market highs contributes directly to its superior cumulative return. This suggests a nuanced understanding of market timing and asset valuation, which are crucial for maximizing investment gains.

- Acceptable Risk

While the maximum drawdown has increased, it remains within an acceptable range given the enhanced returns. The consistency of superior performances at rebalancing points also indicates that it is a stable and balanced strategy that seeks to maximize returns without exposing the portfolio to undue volatility.

- More Flexibility

The model's adaptability to market conditions—by using the cash positions and adjusting the stance at historical highs—ensures that it not only protects gains but also positions the portfolio advantageously for potential market upswings.

6. Conclusion

6.1. Some Comparison

We also compared and tested our Mean-Variance and Maximum diversification strategies on the tested period. The results are reflected below.

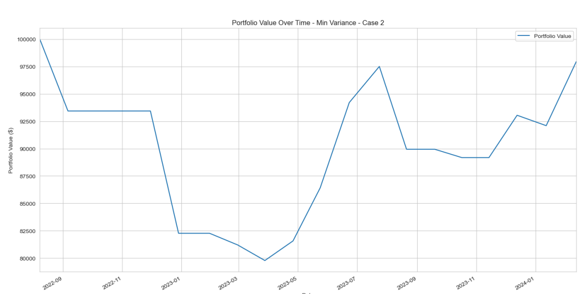


Figure 20. MV - portfolio value over the observed period

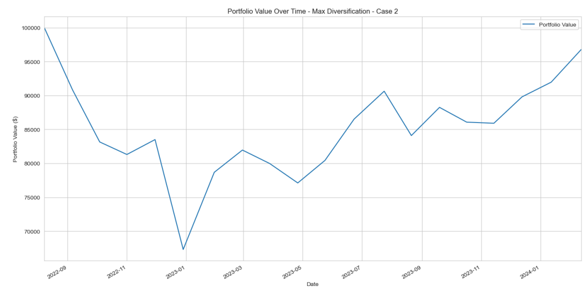


Figure 21. MD - portfolio value over the observed period

From Figure 20 and 21, we can see that both the mean-variance portfolio and the max diversification portfolio fail to generate a return in the test period. In fact, both portfolios performed poorly during this period. Compared to the LSTM or the equally-weighted portfolio, we can see that these two optimization methods are sensitive to market fluctuations, and failed to generate returns. However, we believe this is insufficient to say that the two strategies have failed, or are outperformed by the LSTM model. This is because we initially included extensive training on the LSTM model with abundant historical data.

6.2. Discussion

The essence of LSTM is a type of neural network. While it is widely recognized for its ability to uncover non-linear patterns in data, it is also criticized for its low interpretability. Some researchers even refer to neural networks as “black boxes”, as it is challenging to understand their decision-making process. Even with the adoption of an attention mechanism and training a single layer of LSTM, it is still difficult to interpret the results we obtained effectively.

However, this can be dangerous in real-world scenarios, especially when dealing with portfolio optimization tasks. When we perform optimization, we want to understand how the portfolio is formed deeply.

With this in mind, we can promptly modify the portfolio strategy to respond when the portfolio has unexpected fluctuations. Therefore, we have employed several traditional optimization methods as a comparison and included enhancements to the LSTM portfolio.

Given the inherent low interpretability of the LSTM model, it is prudent to explore alternative models that offer greater transparency and easier interpretation. By comparing the performance of these models with that of the LSTM, we can check whether the portfolio generated by LSTM exhibits fluctuations that align with those of portfolios constructed using traditional optimization techniques. If divergent events occur, we can leverage our understanding of traditional optimization methods to investigate the observed discrepancies. Furthermore, by integrating the merits of both LSTM and traditional optimization techniques, we may formulate innovative hybrid strategies that achieve a balance between the predictive power of deep learning and the transparency of classical methods, leading to a more reliable portfolio management practice.

6.3 Limitations and Future Studies

In our research, we did not directly compare the results of the Mean-Variance, Maximum Diversification, and LSTM portfolios. This is because the LSTM portfolio was trained and tested using only the last 20% of the data, which may not accurately reflect its overall performance. While the LSTM model performed well during this time period after extensive training and tuning, the performance of the Mean-Variance and Maximum Diversification portfolios during this specific period may not be indicative of their general performance. For a more comprehensive comparison, it would be necessary to backtest on multiple time spans and aggregate the results for a fairer conclusion.

Appendix

Date	AAPL_Return	AB_Return	DIS_Return	GOOG_Return	MSFT_Return	NVDA_Return	PYPL_Return	TSLA_Return	CASH_Return	AAPL	AB	DIS	GOOG
2015-07-08	-0.024823	-0.015432	-0.016311	-0.015599	-0.001354	-0.007074	-0.052430	-0.048231	0.002	27.617405	13.559929	107.632240	25.841499
2015-07-09	-0.020396	0.000348	0.003559	0.007449	0.006329	-0.012214	-0.005764	0.011610	0.002	27.054110	13.564651	108.015335	26.034000
2015-07-10	0.026734	0.006616	0.007267	0.018149	0.002022	0.017517	0.005507	0.004769	0.002	27.777388	13.654389	108.800232	26.506500
2015-07-13	0.019305	-0.002421	0.013827	0.030973	0.020847	0.006582	0.060248	0.011615	0.002	28.313641	13.621328	110.304596	27.327499
2015-07-14	-0.000398	-0.021151	-0.001694	0.026622	0.001757	0.000503	0.004078	0.013312	0.002	28.302372	13.333221	110.117714	28.055000

Figure 1. Dataset Overview

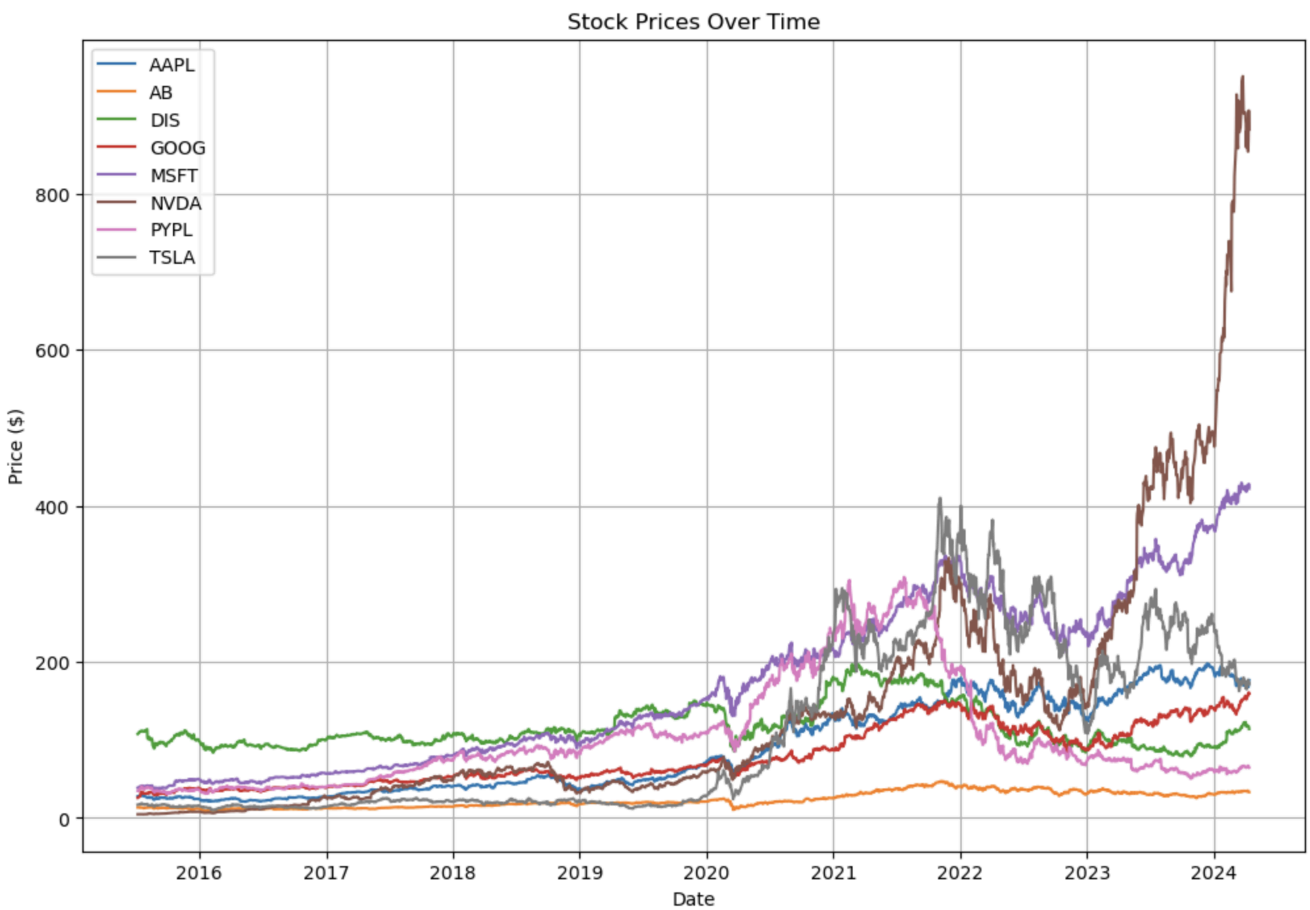


Figure 2. Stock Prices Movements, 2015-2024

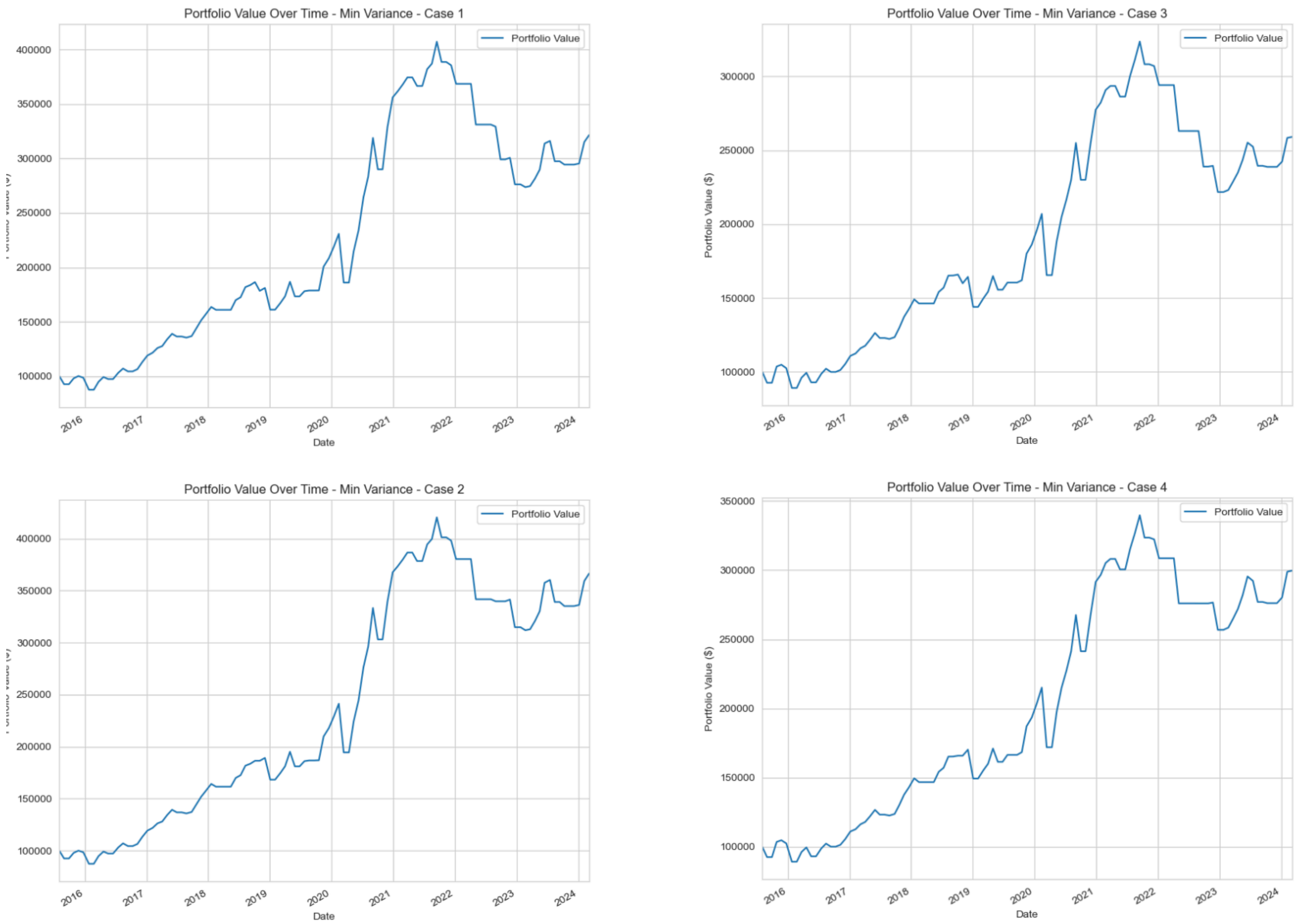


Figure 3. Portfolio Value Over Time in Four Different Mean-Variance Portfolios

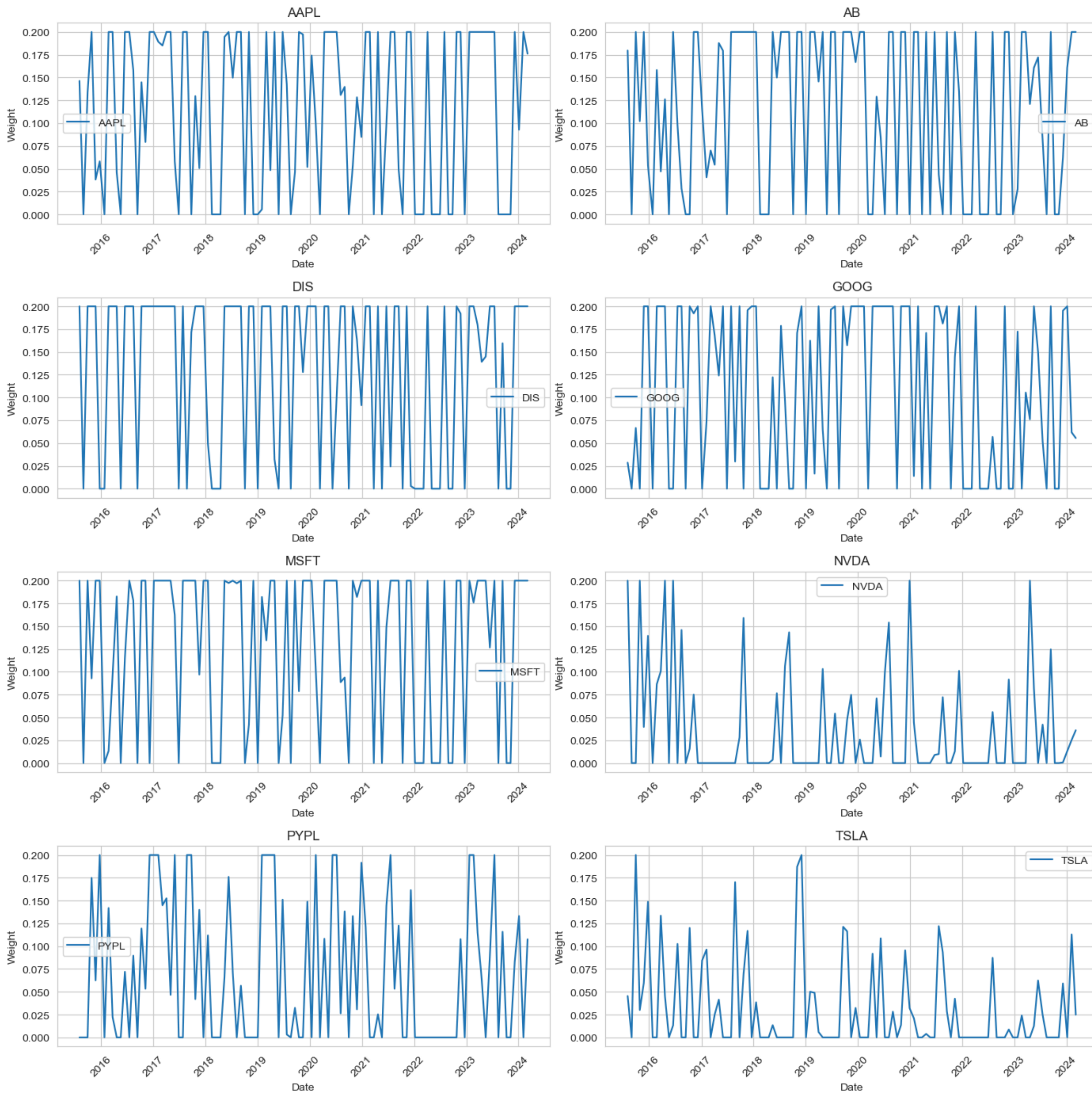
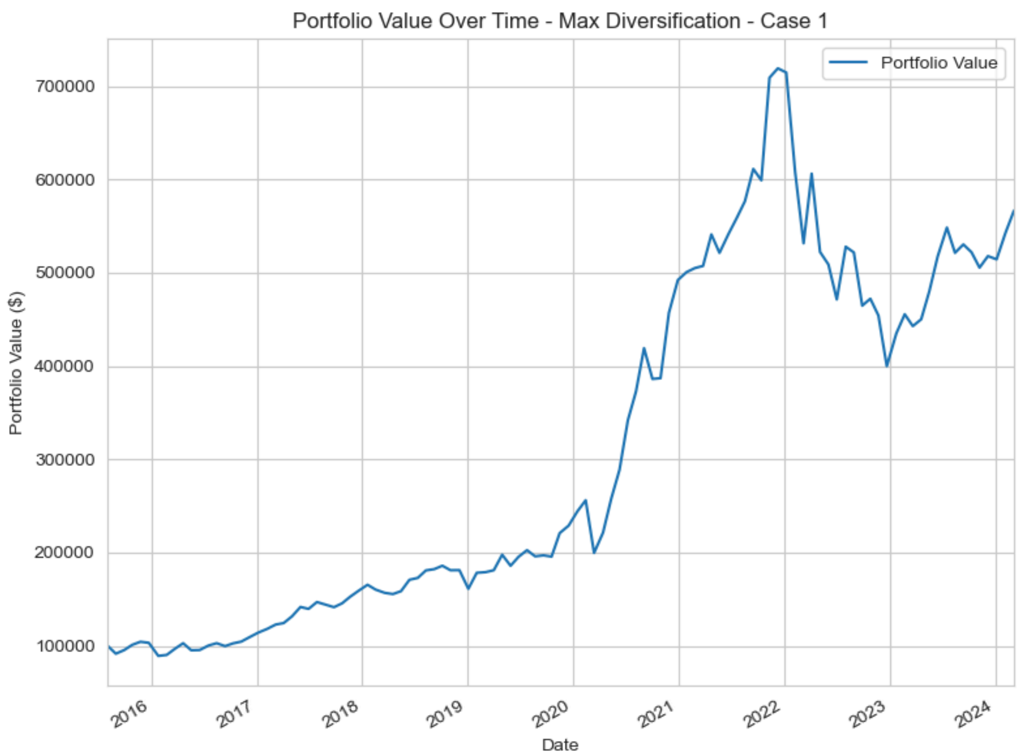


Figure 4. Optimal Weights for Eight Stocks Over Time



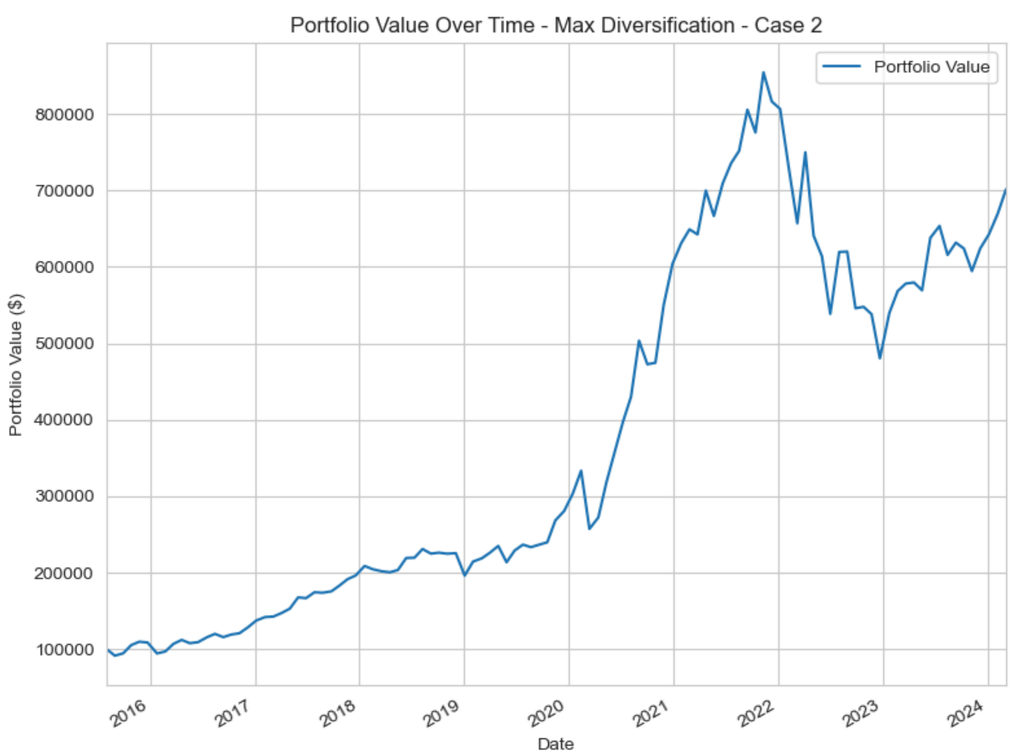


Figure 5. Portfolio Value Over Time in Four Different Max Diversification Portfolios

Figure 6. Optimal Weights for Eight Stocks - Max Diversification - 20-day Covariance

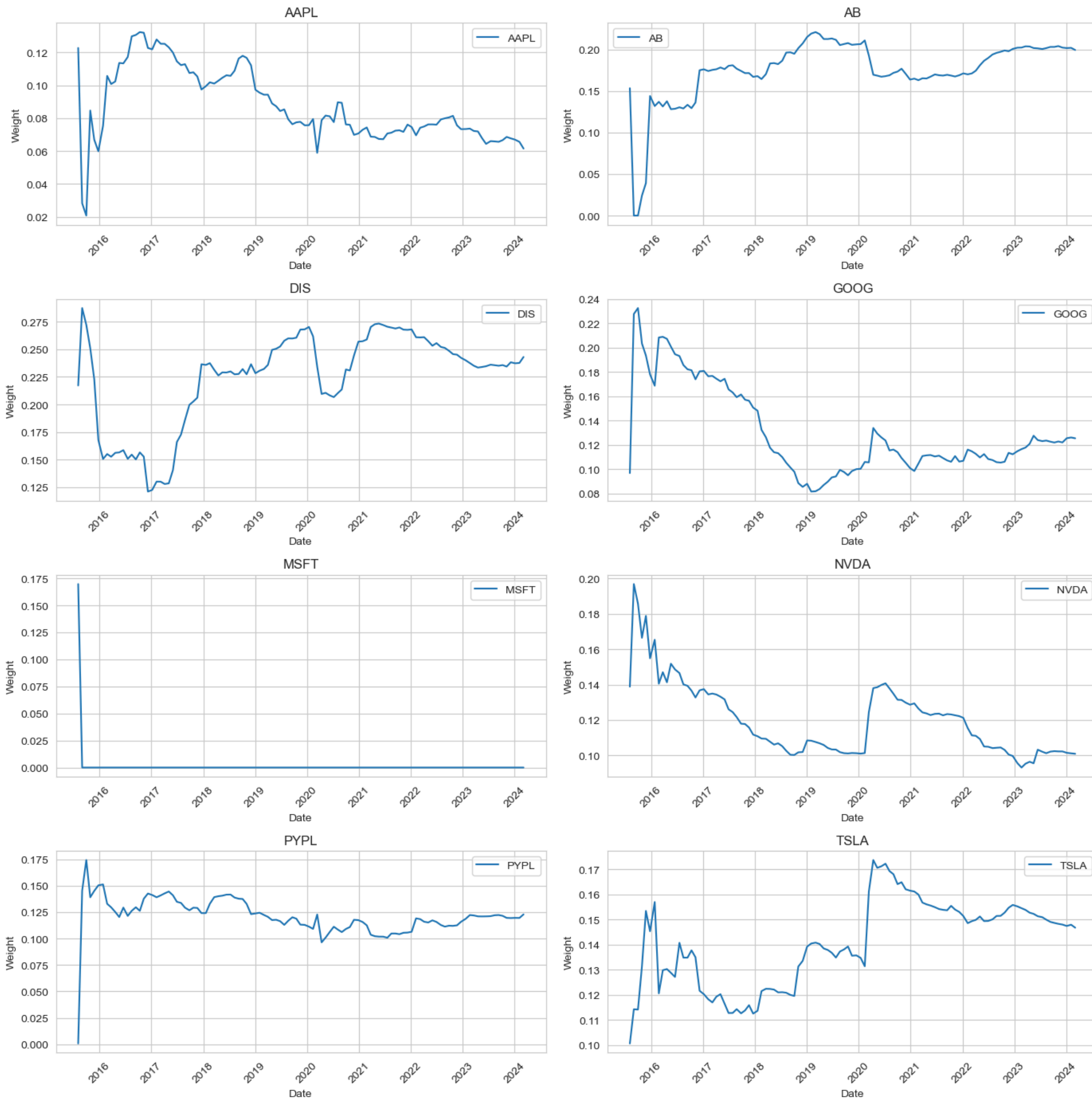


Figure 7. Optimal Weights for Eight Stocks - Max Diversification - All History Covariance

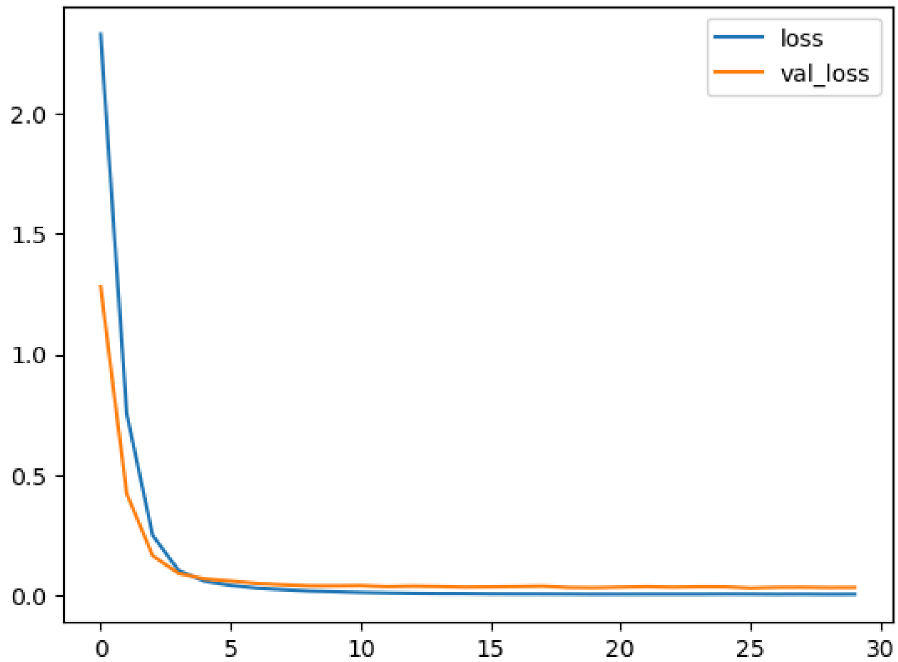


Figure 9. Loss Plot

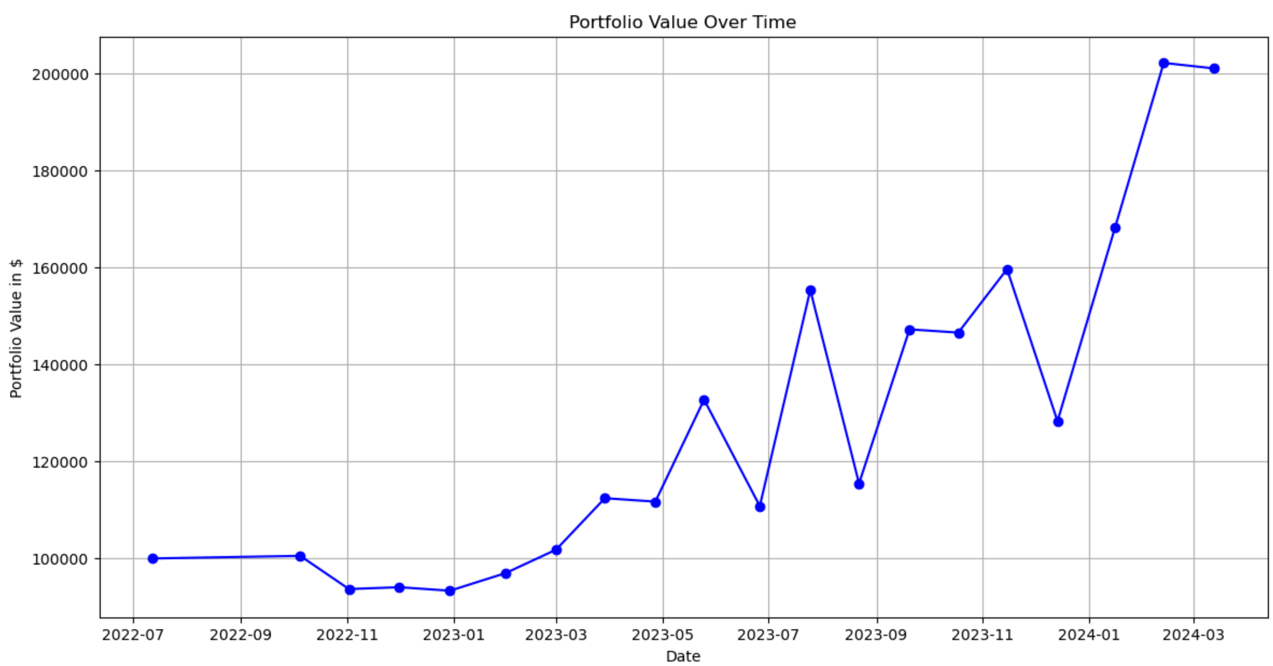


Figure 10. Simple LSTM - portfolio value over time

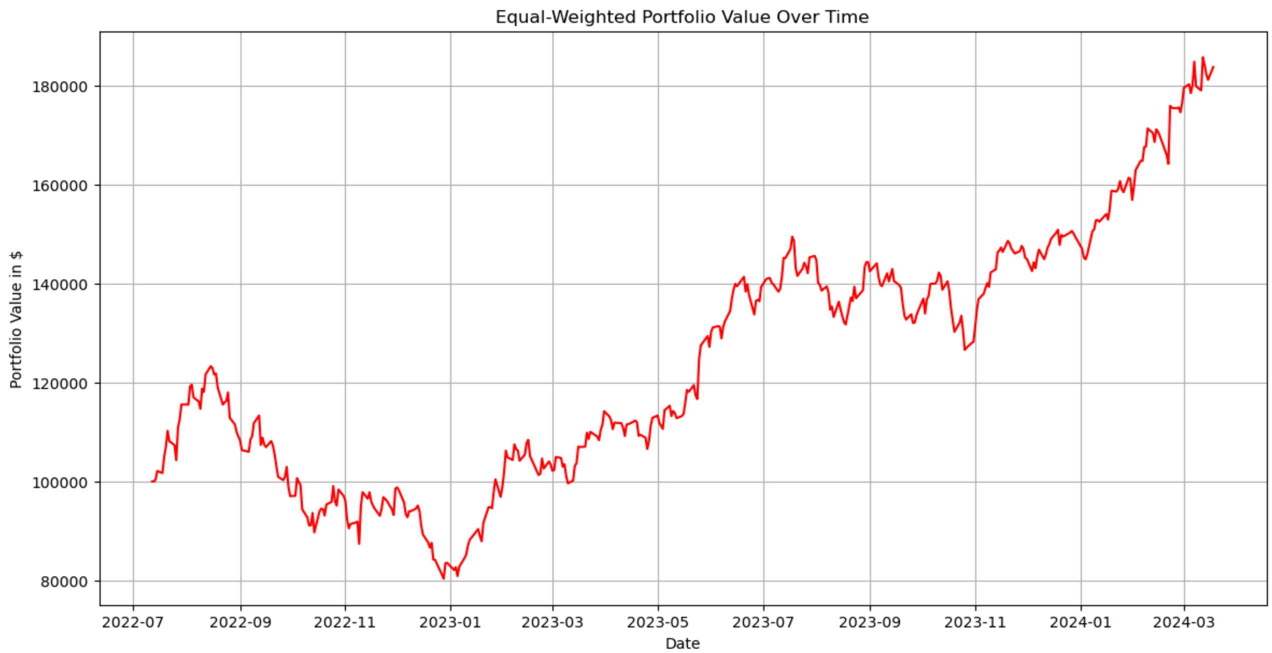


Figure 11. Equal-weighted - portfolio value over time

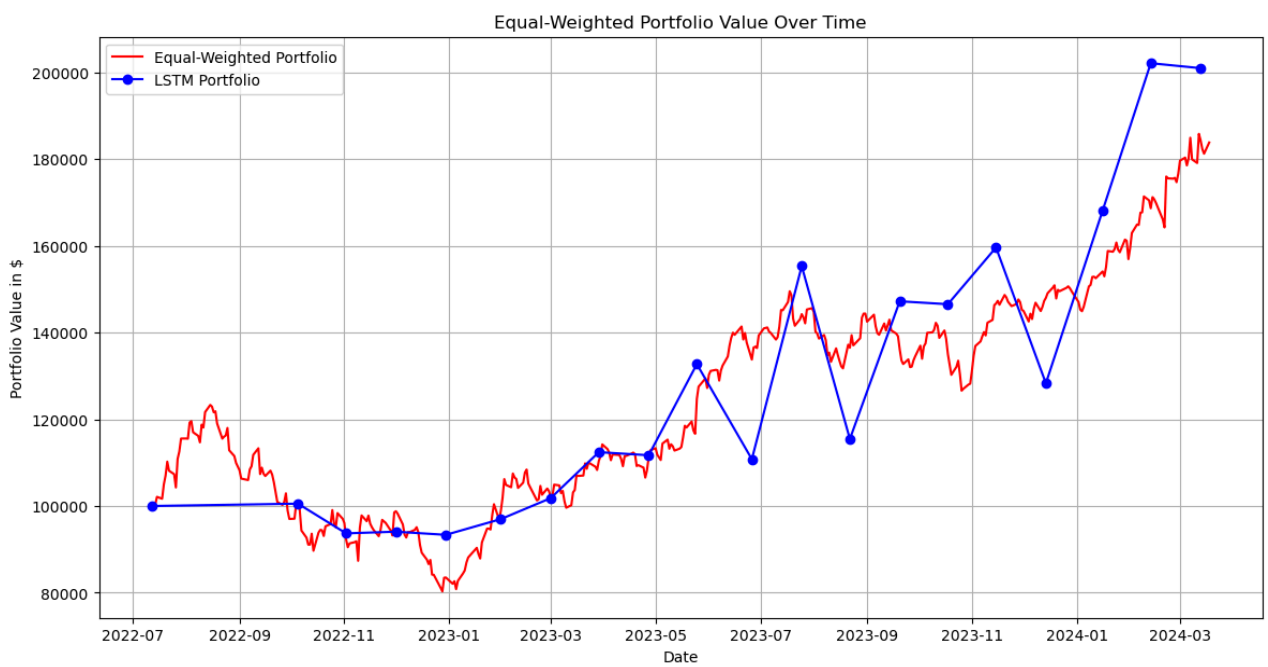


Figure 12. Comparison: Simple LSTM vs. Equal-weighted

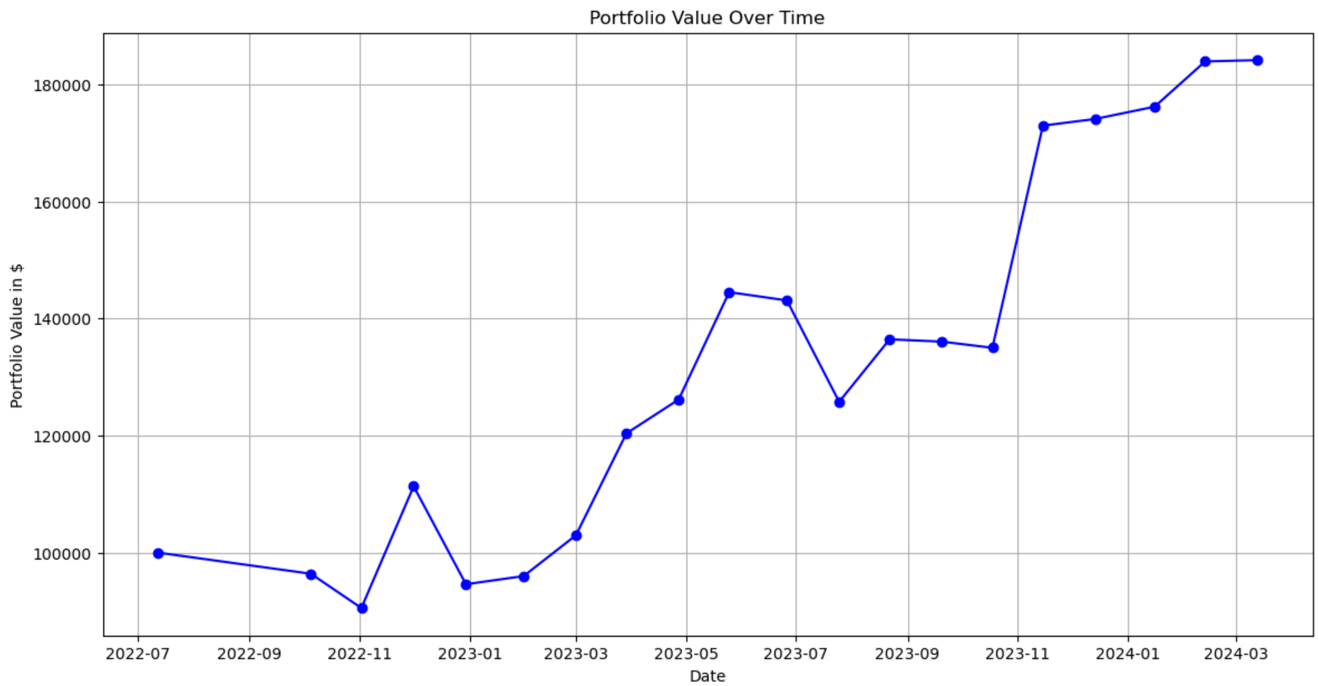


Figure 13. LSTM with cash - portfolio value over time

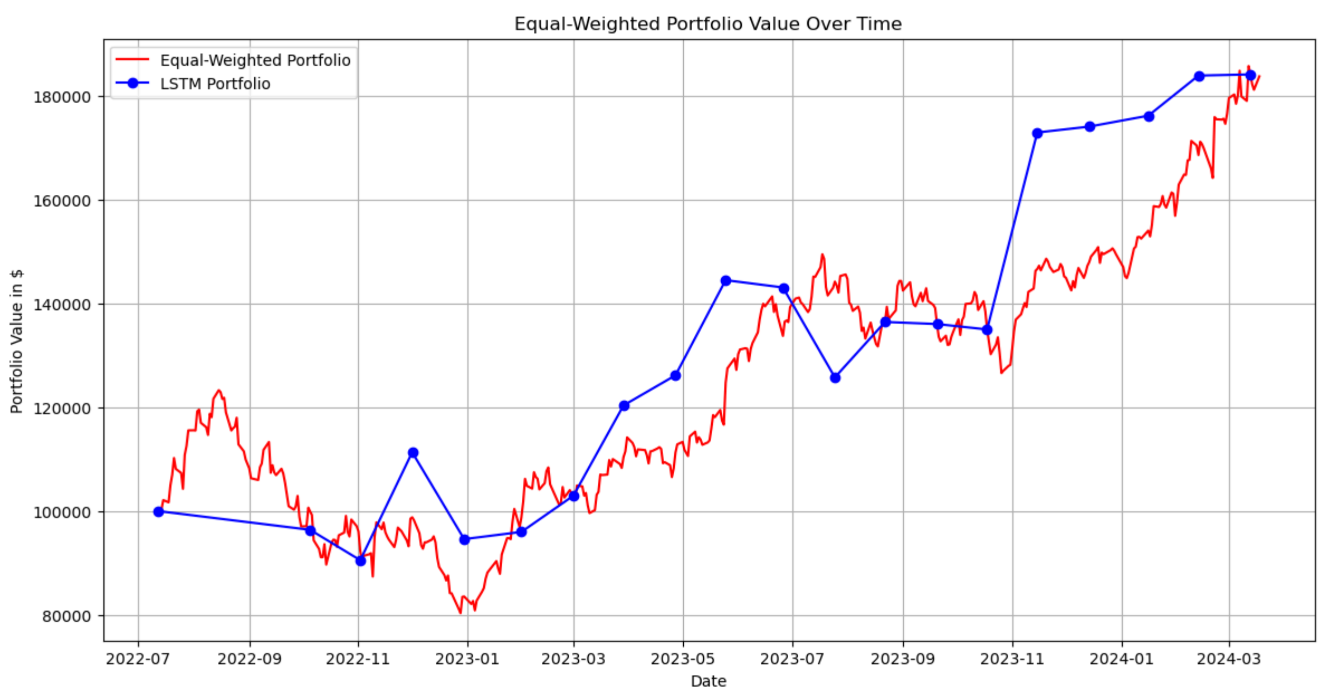


Figure 14. Comparison: LSTM with cash vs. Equal-weighted

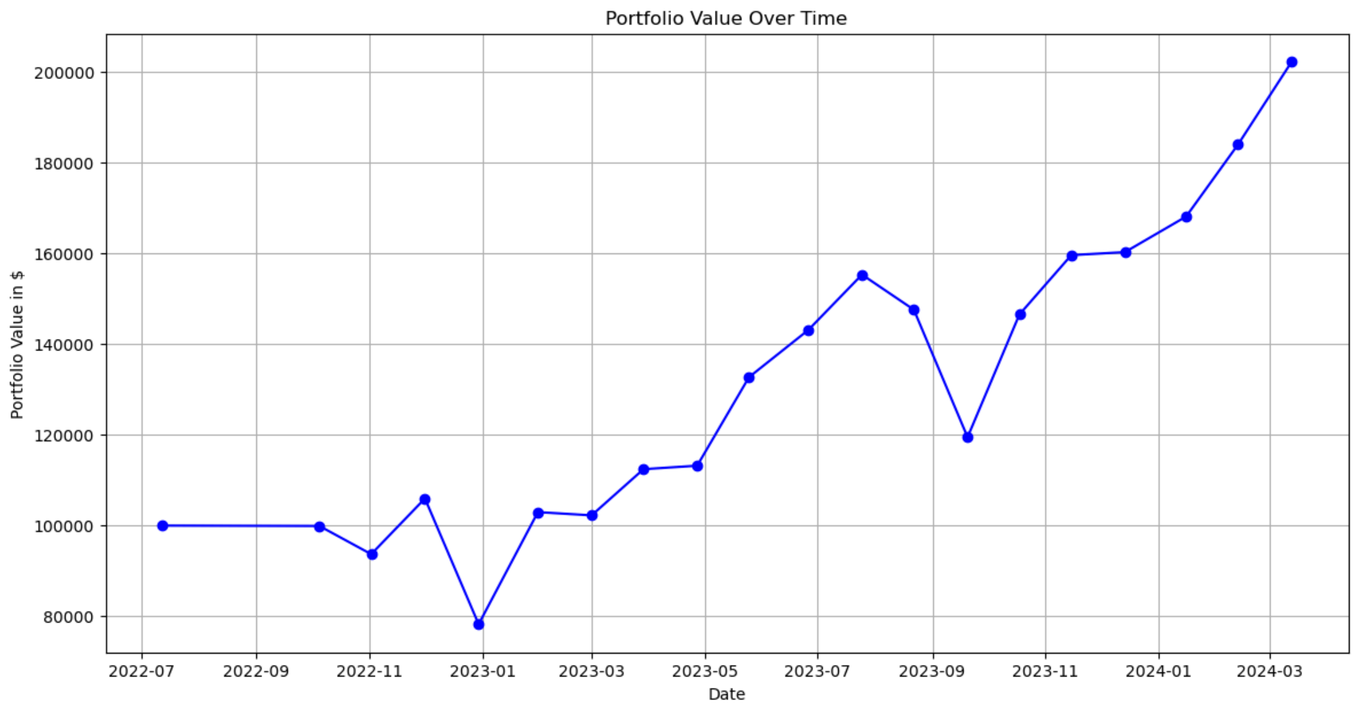


Figure 15. Enhanced LSTM - portfolio value over time

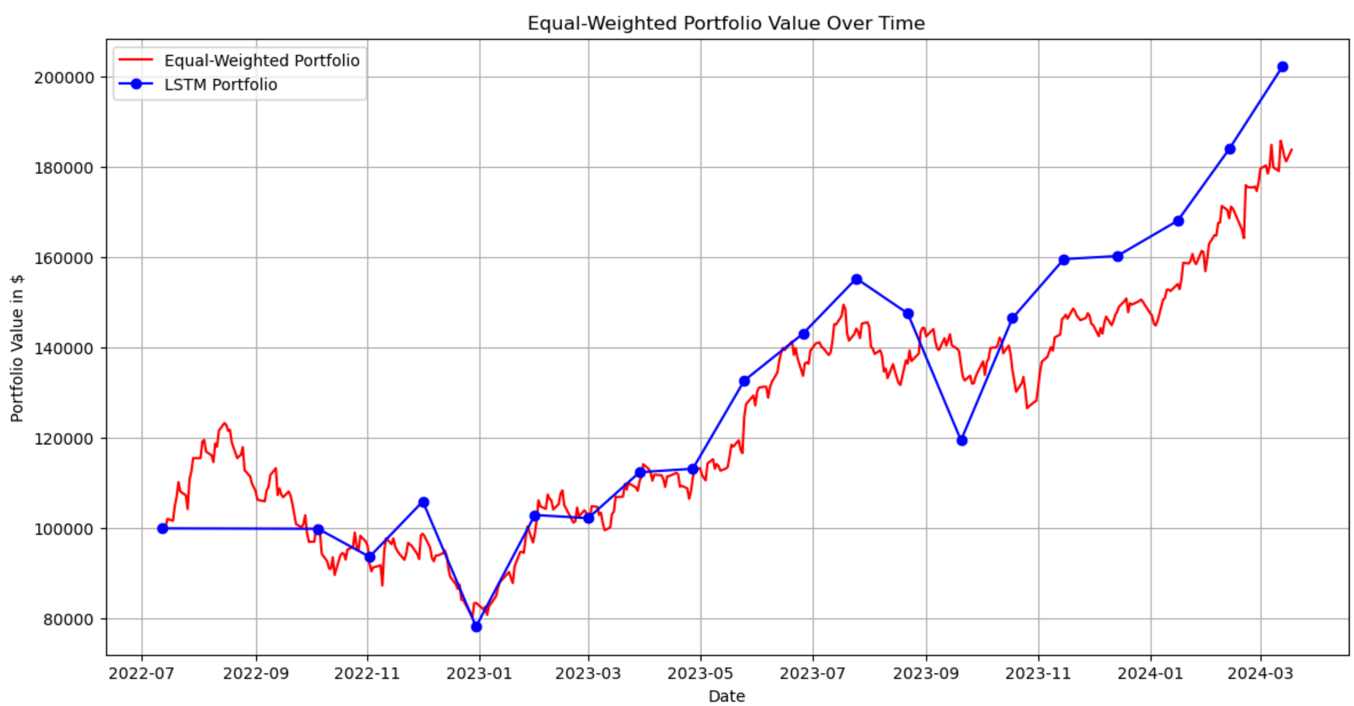


Figure 16. Comparison: Enhanced LSTM vs. Equal-weighted

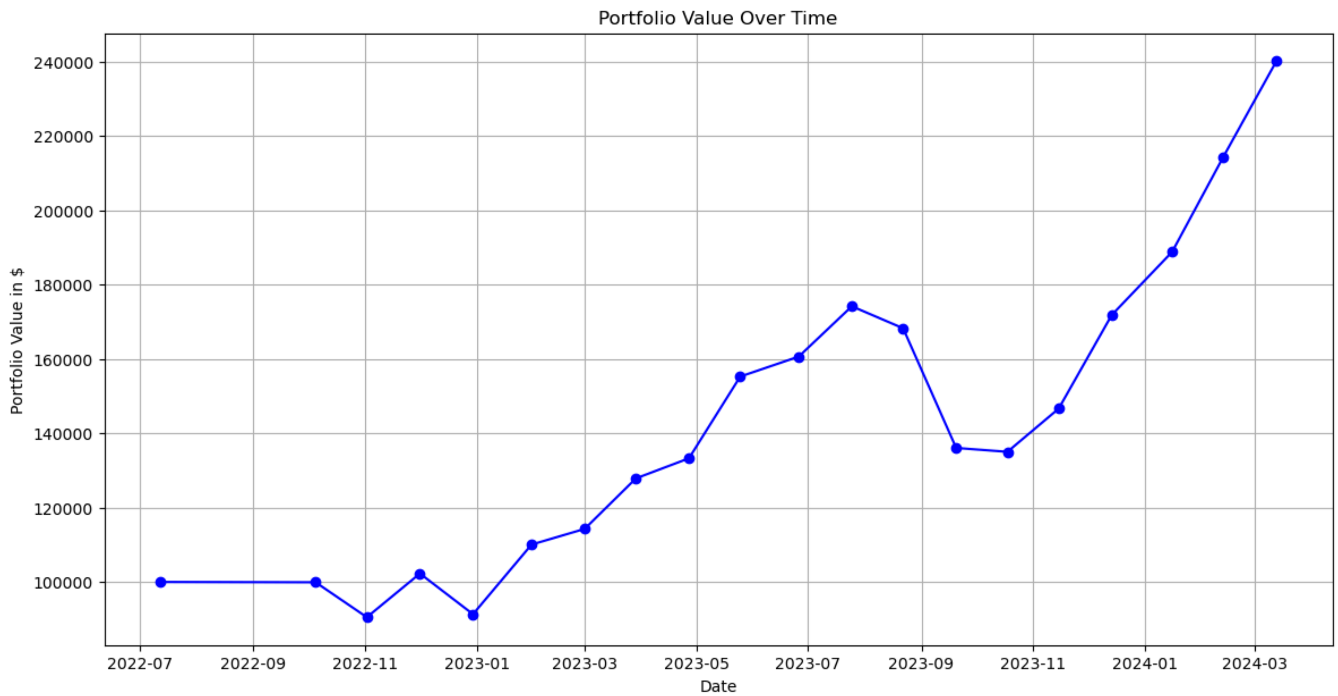


Figure 17. Enhanced LSTM with cash - portfolio value over time

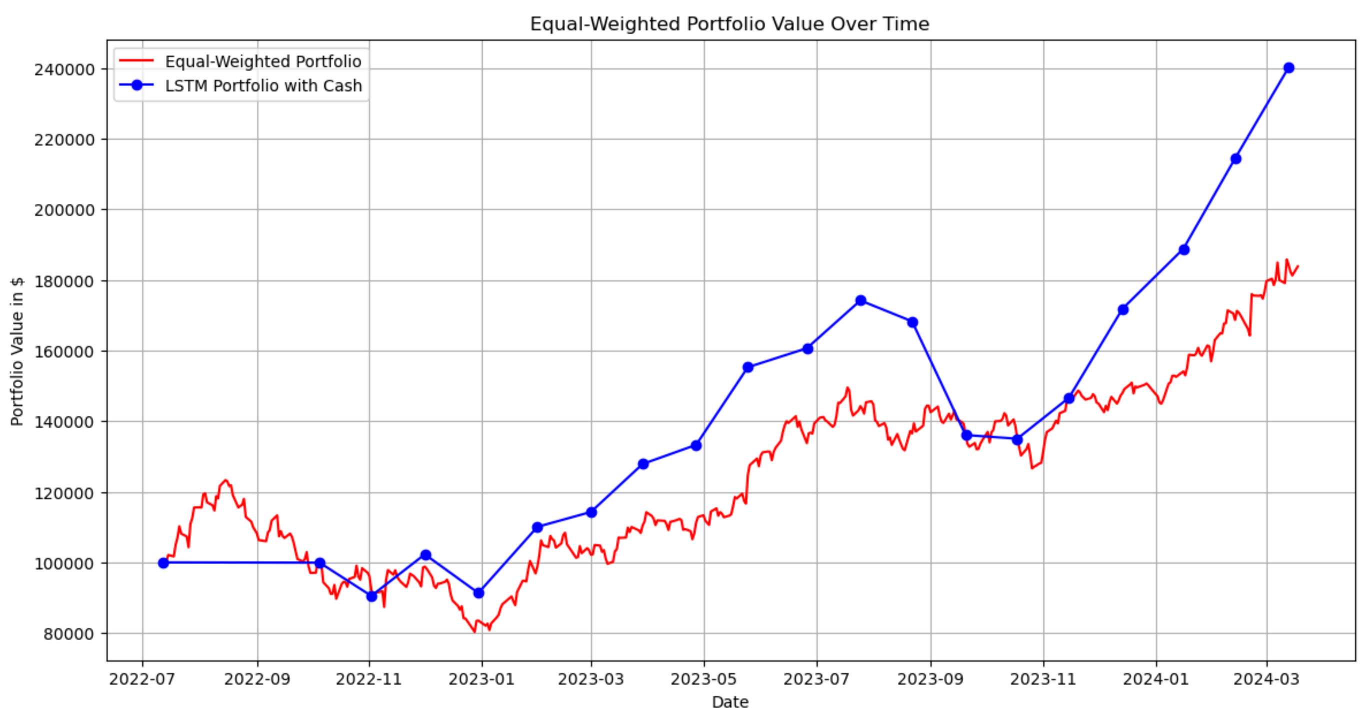


Figure 18. Comparison: Enhanced LSTM with cash vs. Equal-weighted

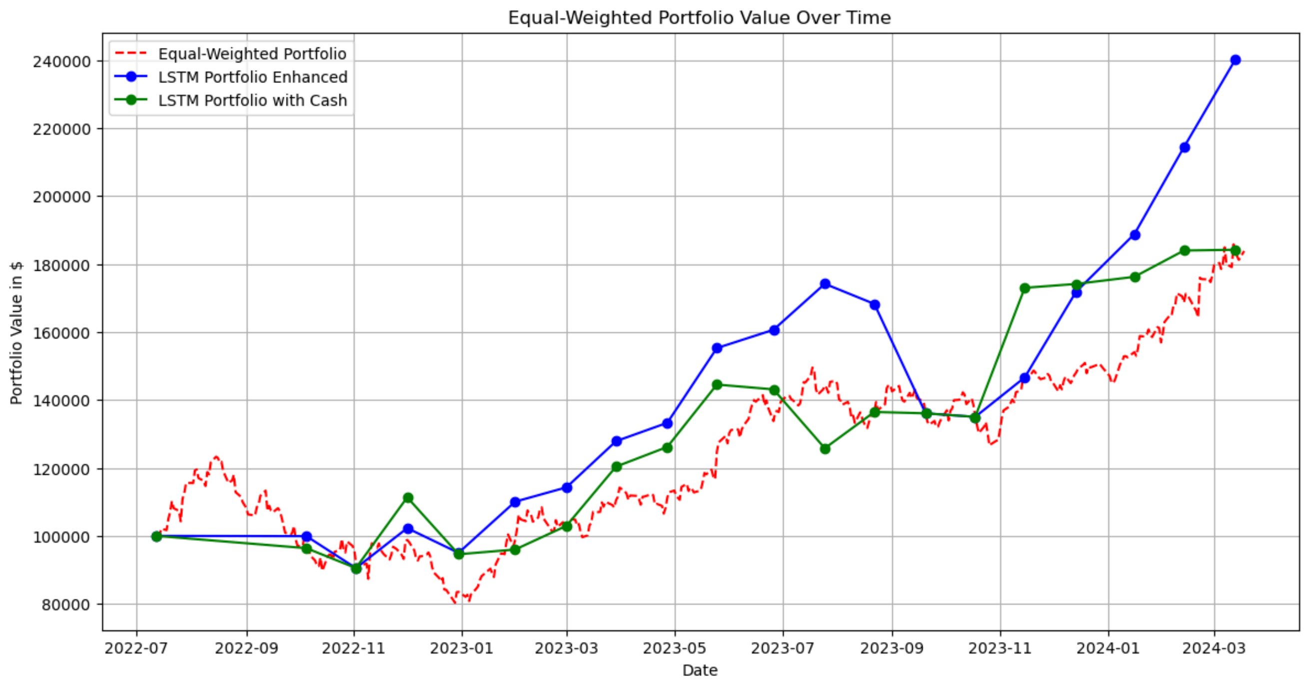


Figure 19. Comparison: LSTM with cash vs. Enhanced LSTM with cash vs. Equal-weighted

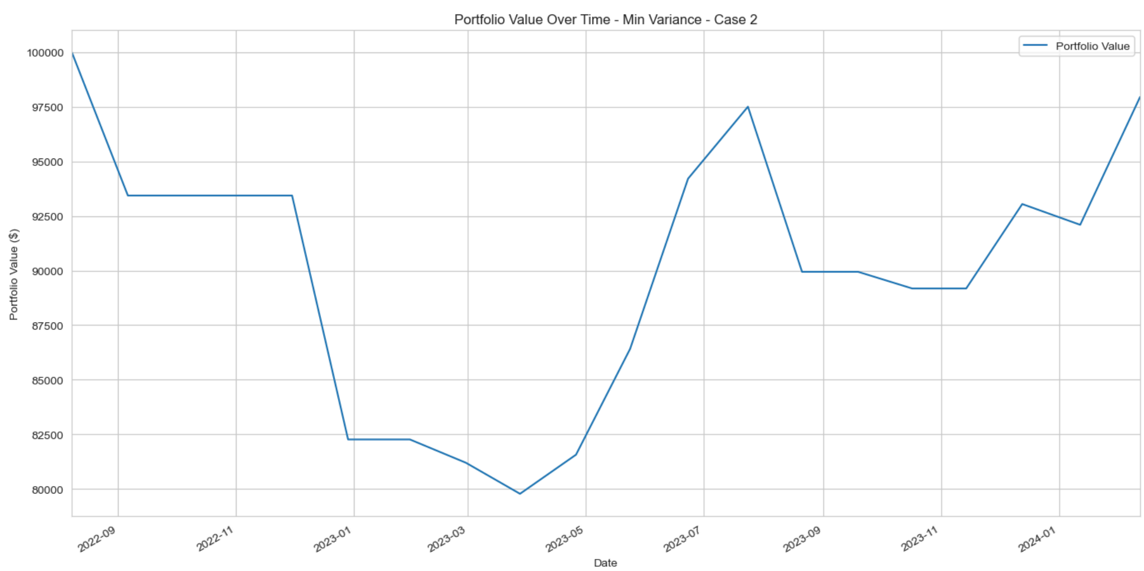


Figure 20. MV - portfolio value over the observed period

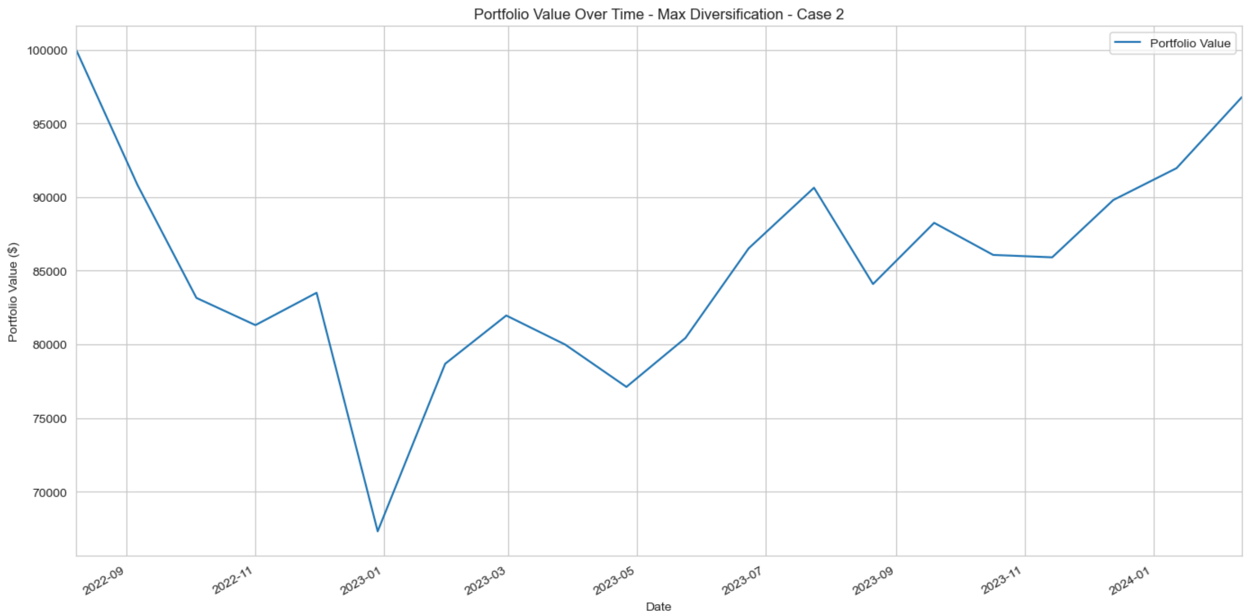


Figure 21. MD - portfolio value over the observed period

References

1. Duggan, W., & Schultz, J. (2024). 8 Top Nancy Pelosi stocks to buy. *US News & World Report*. https://money.usnews.com/investing/articles/top-nancy-pelosi-stocks-to-buy

2. Zhang, Z., Zohren, S., & Roberts, S. (2020). Deep learning for portfolio optimization. *The Journal of Financial Data Science*, *2*(4), 8–20. https://doi.org/10.3905/jfds.2020.1.042

3. Lauricella, T. (2022). Lessons from 2022's tech stock collapse. Morningstar, Inc. https://www.morningstar.com/stocks/lessons-2022s-tech-stock-collapse

4. Choueifaty, Y. (2008). Towards maximum diversification. *Social Science Research Network*. https://doi.org/10.2139/ssrn.4063676

5. University of South Carolina. (n.d.) IEOR 4500 Maximizing the Sharpe Ratio. Retrieved from https://people.stat.sc.edu/sshen/events/backtesting/reference/maximizing%20the%20sharpe%20ratio.pdf