Portfolio Optimization

Kripa Bansal, Natalia Ramirez
Sophia Ford

STA-395: Intro to Machine Learning

Link to code and data: https://github.com/sford6023/ML-Final-Project

1 Introduction

In a world of ever-increasing market volatility with threats such as geopolitical risk, pandemics and climate change, investors have started to place greater emphasis on reducing portfolio risk rather than focusing solely on portfolio returns. Volatility, a statistical measure of the dispersion of returns for a given market index, is associated with inherent investment risk. Volatility and risk are often used interchangeably and are both mathematically represented by standard deviation. In periods of high volatility, asset prices can swing dramatically, which can lead to significant monetary losses or gains. In fact, new research from EY Global Wealth Management shows that 57% of high net-worth individuals who feel unprepared to meet their financial goals cite market volatility as a primary reason. [1]

Additionally, recent global economic challenges, such as geopolitical tensions, economic policy adjustments, and unforeseen global health crises, have underscored the importance of robust risk management strategies in investing. Thus, leading investors to seek smarter, data-driven methods to shield their portfolios from potential downturns while still capitalizing on opportunities for growth. In this context, leveraging advanced machine learning techniques to optimize portfolio management strategies has gained significant traction. One such example is the popularization of Quantitative or Quant Trading, a technique that uses automated models to predict trends in the current market and trades based on the resulting analysis, optimizing portfolio returns[2].

[1] EY Global Wealth and Asset Management Sector, "Investor behavior changes in face of increasing market volatility as demand shifts to active investments and FinTech," EY - Global, accessed May 7, 2024, `https://www.ey.com/en_gl/newsroom/2023/04/investor-behavior-changes-in-face-of-increasing-market-volatility-as-demand-shifts-to-active-investments-and-fi`

[2] Kessler, Vishy Tirupattur, and Stephan, "Quant Strategies Offer a Bright Spot," Morgan Stanley, accessed May 7, 2024, https://www.morganstanley.com/ideas/quantitative-investing-outperformance-2023.

Among the various strategies available, mean-variance optimization, a fundamental component of modern portfolio theory introduced by Harry Markowitz in 1952, aims to create a portfolio that has the best possible balance between expected returns and risk. According to this theory, for a given level of expected return, there exists an optimal portfolio that offers the lowest possible volatility, and vice versa. This set of optimal portfolios is known as the Efficient Frontier, and it serves as an essential tool in strategic asset allocation. Our research employs mean-variance optimization derived from the Efficient Frontier theory to explore whether this approach can effectively reduce the volatility of a portfolio. Our project looks at two types of portfolios: the first one has equal weights assigned to each asset, and the second one has the optimized weights that we calculate using the Efficient Frontier theory, assigned to it. We then use machine learning models to both predict portfolio returns for a period of 3 years into the future, as well as compare the returns for both portfolios and see which one performs better. By integrating machine learning algorithms with this traditional financial theory, our study seeks to advance the field of quantitative finance by enabling more precise and dynamic portfolio management through predictions of asset prices. Our research hypothesis is to investigate if using the Efficient Frontier theory to optimize portfolio weights reduces the overall portfolio volatility when predicting future asset prices.

2 Data and Methodology

This study used the Python SciPy library to retrieve financial data from Yahoo Finance. The dataset merges a selection of four distinct assets: SPDR S&P Regional Banking ETF, Google, Walt Disney, and a Natural Gas stock, each representing different sectors and asset classes. These assets were chosen based on their low correlation coefficients to diversify [3] risk and optimize potential returns in accordance with the principles of the Efficient Frontier theory. The dataset spans from May 8, 2014, to May 6, 2024.

Initial data preprocessing involved pulling each ticker's public information and concatenating it into a DataFrame with a "Ticker" column, and "Date" as index. Each stock had a corresponding open price, closing price, adjusted closing price, lowest price, and highest price. The adjusted closing price of all stocks was then plotted to identify the general growth of the asset over time (Figure 1). Subsequently, the returns of each stock over a period of 10 years was calculated using the `pct_change` function, which calculates the percentage change from one row to another, replicating a daily returns calculation. The returns of each stock were stored in the `returns` DataFrame where each column represents a ticker and the date is set as index.

The covariance matrix between each ticker was calculated in a period of 10

[3] Diversification is the practice of spreading your investments around so that your exposure to any one type of asset is limited. This practice is designed to help reduce the volatility of your portfolio over time`https://www.fidelity.com/learning-center/investment-products/mutual-funds/diversification`

years to start applying Modern Portfolio Theory (MPT) principles and identify the Efficient Frontier. The function `portfolio_performance` uses the initial weights of each asset, their mean returns, and the covariance matrix to calculate the weighted returns and the standard deviation (volatility) of different portfolio weight compositions, enabling the evaluation of potential investment strategies using the Efficient Frontier.

The calculations performed to obtain expected weighted returns were based on the formula:

$$E(R_p) = \sum_{i=1}^{n} w_i \cdot E(R_i)$$

Where n is the number of assets in the portfolio, w_i is the weight of an asset i, and $E(R_i)$ is the expected return for that asset.

Moreover, the following formula was utilized to calculate the portfolio standard deviation:

$$\sigma_p = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} w_i \cdot w_j \cdot \sigma_{ij}}$$

Where σ_{ij} is the covariance between two assets.

The weights are constrained in the range between $0 - 1$. Furthermore, the objective `risk_free_rate` or maximum standard deviation was set to 0.0001, and the initial weights were uniformly calculated as $w_i = n * \frac{1}{n}$. Through the `minimize` function from the Scipy library, the optimized weights are calculated with the goal of minimizing the standard deviation while optimizing returns. The function uses the Sequential Least Squares Programming method and follows the optimization problem formula:

$$\min_{w_1, w_2, \ldots, w_n} \sigma_p^2$$

Two new DataFrames were assembled with the calculated expected returns, the `returns` DataFrame contains columns with ticker names and their respective returns, a new column called "Weighted Returns" was added to display the portfolio performance with optimized weights at a certain date. Moreover, the `unweighted_returns` has the same information, except with a column called "Unweighted Returns" that displays the portfolio performance with uniform weights, or without optimization.

In order to construct the machine learning model, first we defined the features that would be used to predict the asset prices. The data splitting is performed based on time; the training data will span from 2014/07/24 to 2021/06/10, while the testing data ranges from 2021/06/10 to 2024/05/10. Due to the nature of this time-based model, the study uses the moving average of portfolio returns along with its moving standard deviation over a 10 years period as predictors. Moreover, the previous daily returns of the portfolio and the moving average of the original adjacent closing prices over a 10 years period and a 50 years period are also used as features.

Both `returns` and `unweighted_returns` are split into train and testing sets with the correct time frames. Then, GridSearch is used to determine the best parameters for a `RandomForestRegressor` model. After a total of 240 fits, the best performing model using unweighted returns as the training data has a max depth of 3, 5 max features, and 200 estimators. The model's Negative Mean Squared Error was -0.0079.

Finally, the model is tuned further through a `VotingRegressor` that will average the predictions of the input estimators, resulting in a reduction of variance. The `VotingRegressor` takes the best performing model from a `StackedRegressor` with three estimators, a `SVR`, a `DecisionTreeRegressor`, and a `KNeighborsRegressor`, and the best `RandomForestRegressor` model. The resulting model is then fitted with the unweighted training data, and tested with both the weighted and unweighted test sets.

We have integrated a comparative analysis between the traditional `RandomForestRegressor` and a neural network approach using Long Short-Term Memory (LSTM) models to enhance the robustness of the research effort. Parallel to the `RandomForestRegressor`, the objective of the LSTM model is to specifically address the sequential nature of stock market data. This model was designed to capture patterns prevalent in financial time series. Unlike traditional models, the LSTM has the capability to remember information over extended periods, which is critical for predicting outcomes based on long-term trends.

The LSTM network was trained using the same datasets, ensuring that both models operated under comparable conditions. The LSTM model's performance was then evaluated against that of the enhanced RandomForest model.

3 Results

Our Stacked Regressor model displays an RMSE of 0.0101 for weighted predictions and 0.0108 for unweighted predictions, in the time range 2021/06/10 to 2024/05/10. As illustrated in Figure 3, the predicted unweighted returns (shown in blue) and the actual unweighted returns (shown in orange) are closely related, they follow similar patterns of ups-and-lows. Suggesting that the predictive techniques employed are robust, capturing the essential trends and fluctuations in future portfolio returns. Additionally, when comparing the volatility of both weighted and unweighted predicted returns, the weighted set of assets shows a lower volatility with a difference of $4.0605e^{-}05$. This comparison suggests that in a three year period the returns of a weighted portfolio will be safer and have successfully minimized volatility, based on a Stacked Regressor model trained on unweighted returns. To further confirm our hypothesis and eliminate the bias from one single model trained on non-optimized weights, two neural models were constructed and tested to accurately predict returns over three years.

The LSTM models were trained over 10 epochs for the weighted and unweighted data frames, demonstrating rapid convergence from initial high loss values to minimal losses. Specifically, the training loss for the weighted model decreased from 0.0076 to 0.00008907, and for the unweighted model from 0.0057 to 0.00012. This rapid improvement highlights the LSTM's effective adaptation to the complex patterns within the financial time-series data. Validation losses mirrored this decline in a lower scale. The models stopped training after achieving minimal improvements, which was noticed at 10 epochs.

Visual analysis from the figures shows a close alignment between actual and predicted returns, with error distributions mostly clustering around zero, indicating high predictive accuracy. The Mean Squared Error (MSE) values recorded were exceptionally low, with 0.0002 for the weighted model and 0.0003 for the unweighted model, confirming the effectiveness of the LSTM models in financial forecasting.

The average predicted weighted volatility over this period was 0.0062, while the average predicted unweighted volatility was significantly higher at 0.0097. The difference in volatility, calculated as 0.0035, indicates that the weighted predictions were more stable compared to the unweighted predictions. This lower volatility in the weighted model suggests that applying weights to the training data contributes to a reduction in the prediction risk, leading to more stable outputs over the 3 year period.

As illustrated in Figure 8, the neural network's predictions (both weighted and unweighted) align closely with the actual returns, capturing the primary trends and movements within the market effectively. The lower volatility observed in the weighted predictions further supports the hypothesis that a carefully calibrated weighting strategy in the training phase can lead to more reliable and safer investment predictions.

Figure 4 below demonstrates the comparative volatility over time between the weighted and unweighted predictions, highlighting the periods of significant difference and the overall more stable nature of the weighted predictions. Thus, both constructed models are robust in their predictive capabilities.

4 Discussion

Our Stacked Regressor model displays a low RMSE which indicates a great accuracy, as evidenced by its application to unseen future data beyond the training set. The model effectively estimates general trends in the stock returns when compared to the real results. Moreover, the Stacked Regressor Model showcases a lower volatility for weighted returns when minimizing the volatility using Mean-Variance optimization. However, the model could be biased based on its training for unweighted data. In order to address the possible influence of unweighted data we evaluated the two LSTM.

Our LSTM weighted and unweighted models contributed to our previous results and estimated a lower volatility for the unweighted data set. Furthermore, the best predictive model was the LMST model created for weighted returns with a evaluating RMSE of 0.0002.

All models were trained on a substantial historical dataset and subsequently used to forecast asset returns. This forecasting was not limited to the immediate future but extended three years ahead, predicting with the patterns and trends identified during the training phase. Importantly, the predictive success observed suggests that all model can generalize well from historical data to future conditions. This is pivotal for applications in financial planning and investment strategy, where accurate long-term forecasts are crucial.

In conclusion, the study succeeded in creating an accurate LSMT model for predicting future asset returns. After training, this model can estimate further optimized portfolio returns.

5 Bibliography

The resources Wealth and Sector, Fidelity, and Kessler and Stephan, are finance related resources.

References

Fidelity. "Diversification", 2024. Visited on 05/07/2024. `https://www.fidelity.com/learning-center/investment-products/mutual-funds/diversification`.

Kessler, Vishy Tirupattur, and Stephan. "Quant Strategies Offer a Bright Spot", 2024. Visited on 05/07/2024. `https://www.morganstanley.com/ideas/quantitative-investing-outperformance-2023`.

Wealth, EY Global, and Asset Management Sector. "Investor Behavior Changes in Face of Increasing Market Volatility as Demand Shifts to Active Investments and FinTech", 2024. Visited on 05/07/2024. `https://www.ey.com/en_gl/newsroom/2023/04/investor-behavior-changes-in-face-of-increasing-market-volatility-as-demand-shifts-to-active-investments-and-fintech-new-ey-report-finds`.

6 Figures

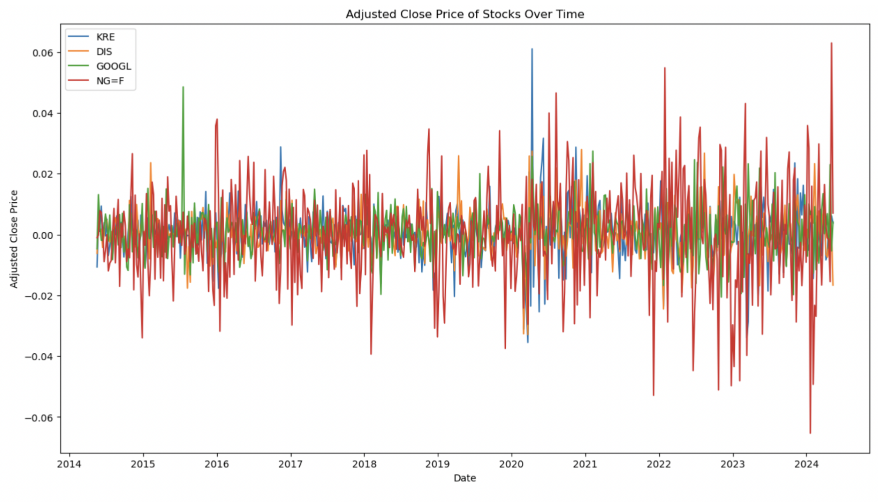

Figure 1: Adjusted Close Prices for KRE, DIS, GOOGL, and NG=F over 10 years

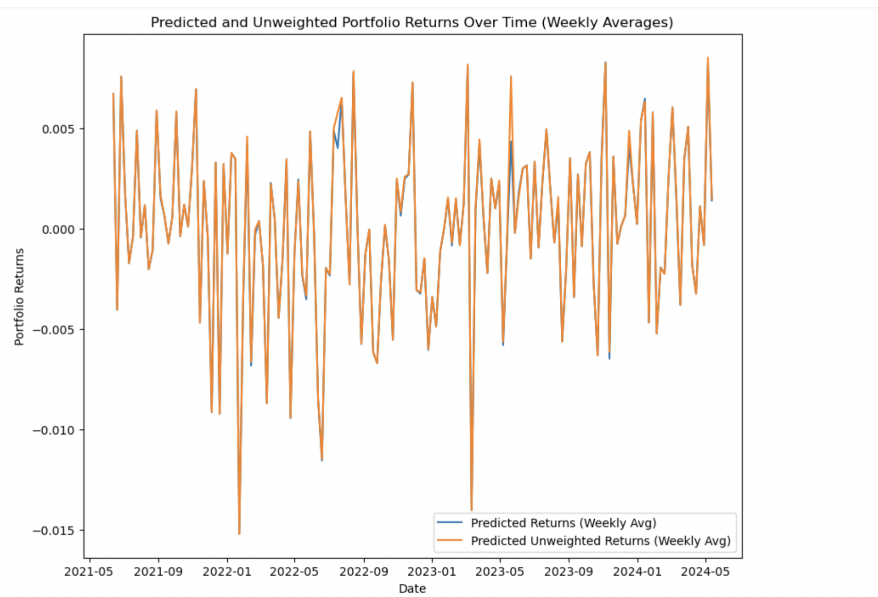

Figure 2: Predicted Weighted Returns and Predicted Unweighted Returns for a 3 year period with Stacked Regressor Model

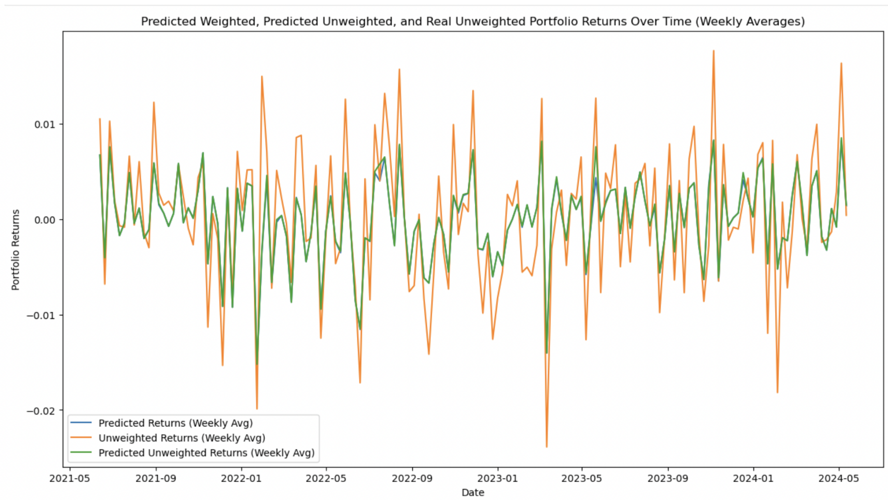

Figure 3: Predicted Weighted, Predicted Unweighted, and Real Unweighted Portfolio Returns Over Time with Stacked Regressor Model (Weekly Averages)

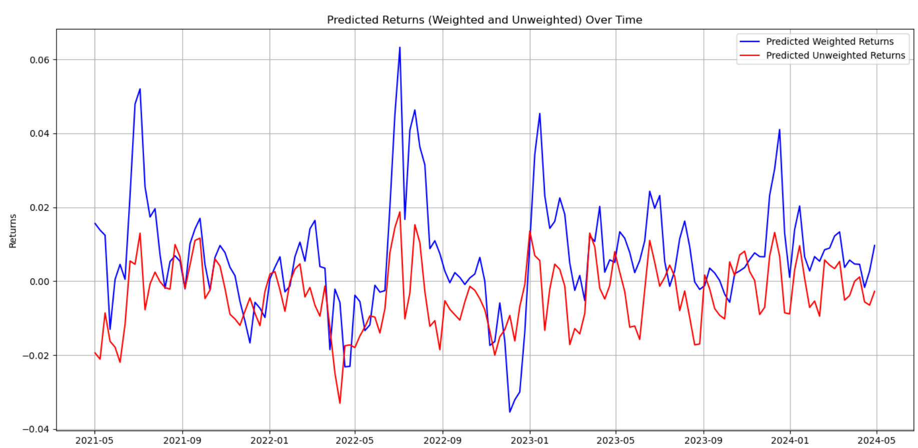

Figure 4: Comparison of Actual vs. Predicted Returns for Weighted and Unweighted Portfolios Over Time with LSTM Model

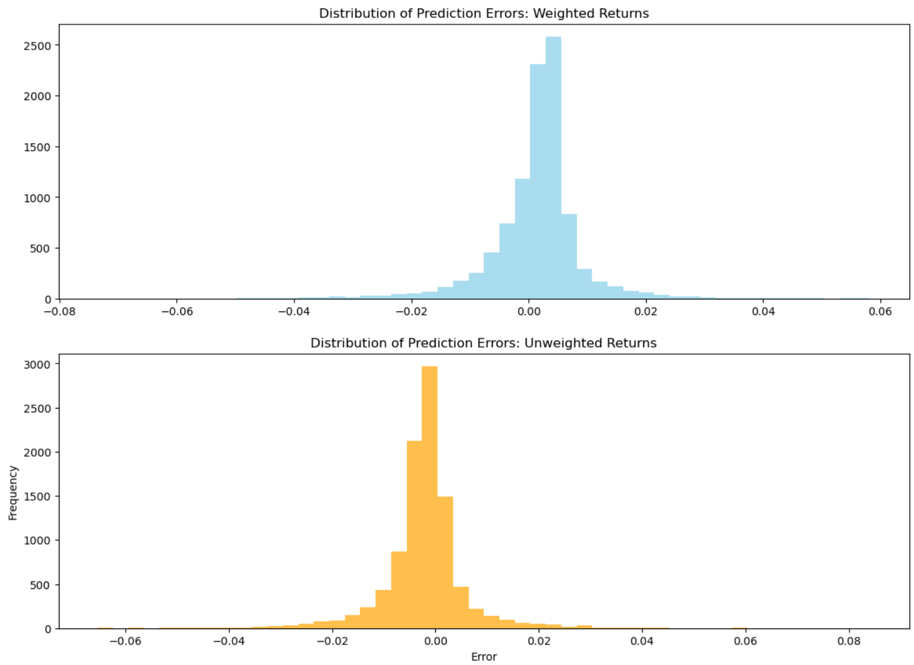

Figure 5: Distribution of Prediction Errors for Weighted and Unweighted Returns for LTSM Model

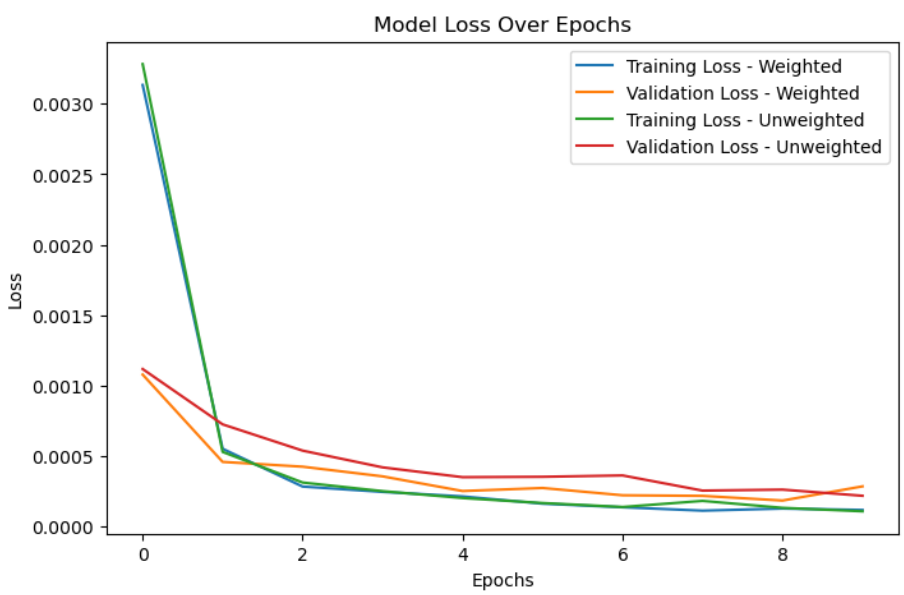

Figure 6: Model Loss by Epoch for LTSM Neural Network

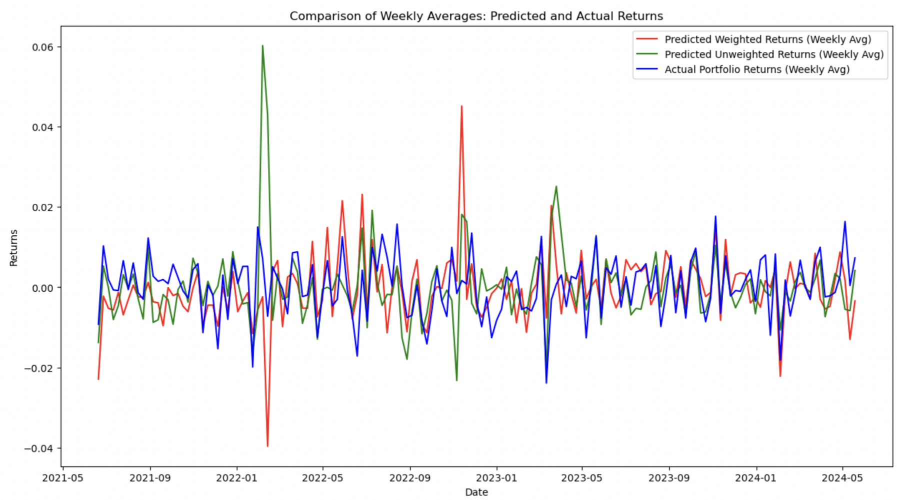

Figure 7: Predicted Weighted and Unweighted Portfolio Return in LSTM compared to actual returns