Multi-layer Perceptron Neural Networks and Time Series Forecasting in Portfolio Optimization

Alina Chou
alinac@stanford.edu

Edgar Roman
emroman@stanford.edu

Tiffany Zhao
tiffzhao@stanford.edu

***Abstract*—Predicting stock market trends is key to quantitative traders to make profit and optimize their portfolios. We explore mean-variance portfolio optimization as well as multi-layer perceptron (MLP) neural network & time series forecasting models to determine the best investing strategy. Using the top ten stocks in the S&P 500 Index for our portfolio, we find that we obtain significant profit returns with the MLP model giving a 224.75% return and the time series forecasting giving a 119.85% return. Mean-variance optimization gives us great returns as well with a minimal-risk portfolio, strongly focusing on investing in Johnson & Johnson given the recent COVID-19 outbreak.**

I. Introduction

We investigate how machine learning techniques can allow investors to profit in the stock market. Trends of stock prices can vary significantly and involve huge datasets of historical values in order to predict a future stock price. With the help of machine learning, we are able to store previous results, learn and analyze them, and predict much more accurate stock trends to allow for traders to realize maximum profit from the market. We investigate machine learning methods, such as construction of neural networks and time series data modeling, to predict accurate expected returns. Then, we calculate the predicted profit made by investing in the stock with the maximum return every day to determine our own optimal portfolios. We also produce an optimal portfolio utilizing mean-variance volatility-minimizing portfolio optimization to determine the best strategy for quantitative traders to make the maximum profit.

II. Related Work

A. The Efficient Frontier

The mean-variance formulation of the efficient portfolio problem dates back to the 1900's, where now there are ideas of principal portfolios and the optimizing effect of the correlation amongst assets. Many research involved realizing that volatility is a reduction feature of the optimum portfolio while the asset set increases [6]. The efficient frontier is more simply related to the principal portfolio environment, meaning we want high-return small-risk portfolios. This also means maximizing the Sharpe ratio (return vs. risk). However, research shows that maximizing the Sharpe ratio could lead to unstable, high-risk portfolios that empirically perform worse than portfolios with minimum variance [1].

B. Machine Learning in Portfolio Optimization

Portfolio optimization models can drastically improve with the help of machine learning methods to prevent noise and estimation issues applied to real data. One method in research is performance-based regularization (PBR), where the idea is to constrain the variances of the estimated portfolio risk and return, which steers the solution toward one associated with less estimation error in the performance [2]. Another method is to rotate between high and low risk efficient frontiers to maintain an optimal selection of stocks, which can be compared to a support vector machine (SVM) model to classify optimal assets to reach a certain target of gain [3], [5]. Finally, research has been done on creating random forest (RF) and support vector regression (SVR) machine learning models for portfolio optimization, as well as implementing LSTM neural networks, convolutional neural networks, and deep multilayer perceptron (DMLP), which is specifically implemented in our work as well [4].

III. Dataset and Features

We use the Yahoo finance data API, yfinance, to gather data with features of the opening price, highest price, lowest price, trading volume, and closing price of the stocks from the top ten companies in the S&P 500 Index: Apple, Microsoft, Amazon, Alphabet, Facebook, Tesla, Berkshire Hathaway, JP Morgan, Visa, and Johnson & Johnson.

A. Mean-Variance Optimization

For our method of efficient frontier portfolio optimization, we obtain the expected returns from the closing prices of the ten stocks mentioned above over the time period from January 1st, 2014 to May 28th, 2021.

B. Multi-layer Perceptron

For dataset and features used in our multi-layer perceptron neural network, we obtain data on all 505 stocks in the S&P 500 with the objective of using as many of these stocks as possible but still have a rather sizeable amount of data. One challenge we quickly face is the fact that there are some companies that recently became public and which do not have available data. For this reason, we decide to exclude 24 stocks, namely ANET, CZR, CARR, CTLT, CFG, CTVA, DOW, ETSY, FTV, FOXA, FOX, HPE, INFO, IR, KEYS, KHC, LW, OTIS, PAYC, PYPL, QRVO, SYF, UA, and WRK. However, we now have complete data for 481 stocks from January 1st, 2014 to May 28th, 2021, and we are able to calculate the returns for each period. We split this dataset into training and testing datasets, where we train our model using the calculated returns from January 1st, 2014 to December 11th, 2019, and test it by predicting the returns from December 12th to May 28th.

C. Time Series Dataset

To implement time series machine learning, we utilize the same dataset used for our MLP neural networks model, but shift the data by one day backwards. We do so by merging two stocks, then obtaining the returns by calculating the difference over the shifted stock price. We also had to consider closely correlated stocks as features for predicting expected returns (ex. yesterday's JPM returns predicting today's BRK-A returns). We show a visualization of the correlation of all ten stocks in Fig 1.

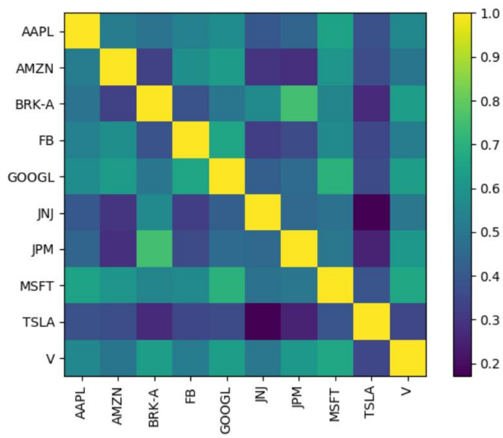

Figure 1: Correlation matrix of expected returns from the top ten stocks in the S&P 500 Index.

IV. Methods

A. Linear Regression for Stock Prices

At first, we utilized linear regression with features of the stock's opening price, highest price, and lowest price to make accurate predictions on the stock's closing price on the subsequent day instead of its overall trend. At first glance, it seems as if our model is performing quite well. However, we notice that our model is overfitting as a lot of weight is placed on the opening price of the stock. This proves the reason why quantitative traders care more about predicting the expected returns accurately to make profit rather than predicting the expected prices themselves.

B. Mean-Variance Portfolio Optimization

We implement mean-variance optimization or the Efficient Frontier method to determine the optimal portfolio of stocks given expected returns, as it is more beneficial for traders than actual stock prices themselves as mentioned previously. There are multiple optimization methods, like maximizing the Sharpe ratio (return vs. risk) or minimizing the volatility. We choose to minimize the volatility given the past research that shows minimum variance portfolios outperforming maximum Sharpe ratio portfolios as mentioned in the related work section. To visualize the efficient frontier and random tangent portfolios, see Fig 2. Because mean-variance optimization sometimes results in negligible weights for each stock due to the difficulty in forecasting future returns, we add L2 regularization to coerce the mean-variance optimizer to produce more non-negligible weights. Here, L2 regularization is used to make weights larger rather than smaller like in most machine learning methods.

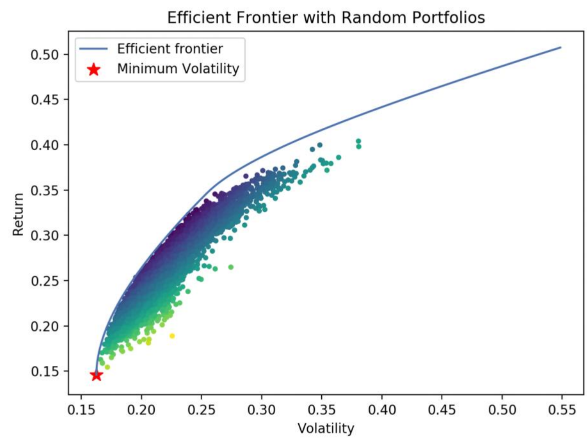

Figure 2: Visualization of efficient frontier and minimum volatility portfolio.

C. Multi-layer Perceptron (MLP) Neural Network

We use Multi-layer Perceptrons in hopes of accurately predicting the daily returns of the stocks we are tracking. More specifically, we are looking to learn a mapping from the returns of all the stocks in the S&P 500 on one particular day to the return of one particular stock the following day. To proceed, we use scikit-learn's MLPRegressor class to train our model, utilizing the 'relu' activation function and 'adam' solver.

D. Time Series Forecasting

We also examine using time series to predict stock price, since we notice that some stock prices are greatly related — an increase or decrease in one can lead to a similar trend in another stock. Therefore, by analyzing the stocks that are greatly correlated, we can use the observed information from one day of one stock to predict the other stock's return in the following day. Specifically, for each stock, we obtain its most correlated stock (ex. referring to Fig. 1, we see that BRK-A is JPM's most correlated stock and vice versa) and use the previous day's return of one of them to predict today's return of the other stock. After obtaining the correlation matrix between the stocks, we find the most correlated stock for each of them. Then, we shift the return data for one of the stocks so that the data is one day ahead of the original corresponding date. For example, we use the previous day's data of BRK-A to predict today's return of stock JPM. As for the prediction, we use both linear regression and logistic regression models for classification and regression.

V. Results

A. Linear Regression for Stock Prices

We show an example of predicted Apple (AAPL) stock prices in Fig. 3, demonstrating linear regression prediction results for the subsequent day's stock price. As one can see, this resulted in high over-fitting.

Figure 3: Predicting the subsequent day's Apple price using linear regression.

B. Mean-Variance Portfolio Optimization Weights

We obtain the optimizing weights for each asset after performing mean-variance convex optimization using the actual expected returns over the duration of January 1st, 2014 to May 29th, 2021, shown in Table 1. We can see that this gives us a strong recommendation to invest in Johnson & Johnson the most, which is consistent with the recent COVID-19 outbreak and stable upward trend in JNJ stock.

AAPL	AMZN	BRK-A	FB	GOOGL	JNJ	JPM	MSFT	TSLA	V
7.90	8.73	18.9	5.99	8.61	22.4	9.5	6.97	1.69	9.31

Table 1: Percentage weights for an optimal portfolio.

C. Multi-layer Perceptron Neural Network Performance

When we train our model and test it, we notice that scikit-learn's R^2 score when comparing our predicted returns to the actual returns is quite low, in fact is is negative.

However, we note that it is not as important to predict the actual return value as it is to predict the sign of the return. However, the results are not very promising.

AAPL	AMZN	BRK-A	FB	GOOGL	JNJ	JPM	MSFT	TSLA	V
0.4454	0.4863	0.5492	0.5027	0.5137	0.5245	0.5437	0.5410	0.5109	0.5

Table 2: Accuracy of predicting sign (positive or negative) of returns using the MLP neural network.

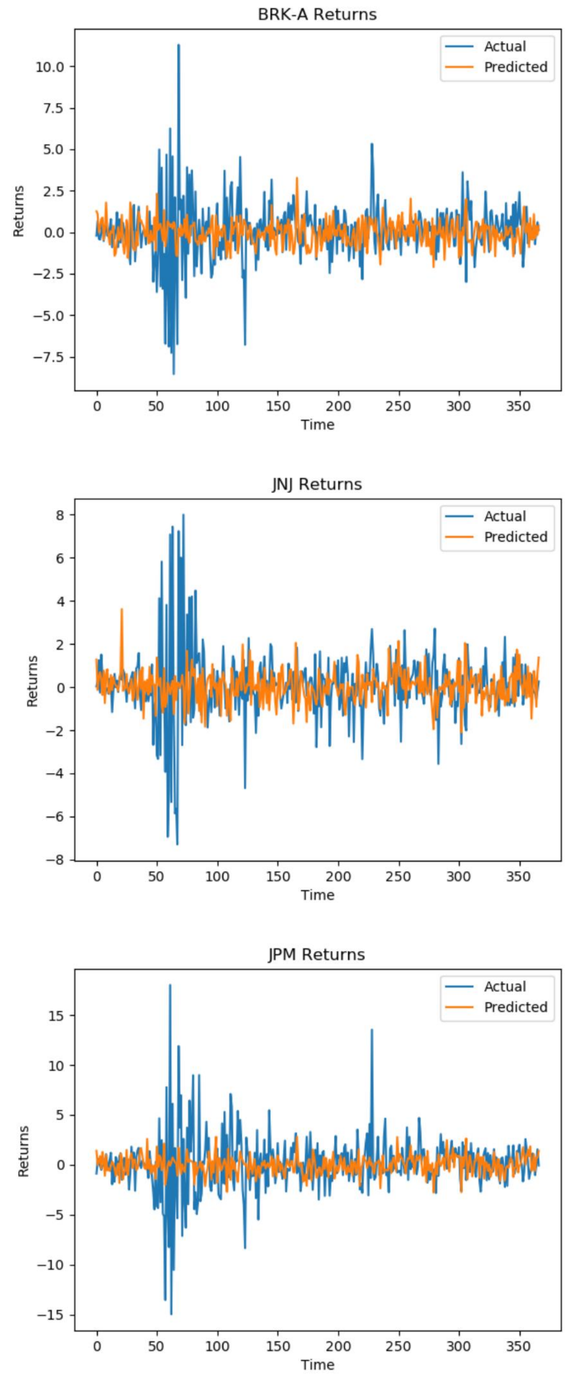

Figure 4: Predicting the subsequent day's return using the returns of the S&P 500 stocks from the previous day.

Nevertheless, since we have predictions on the sign of the returns of 10 that hover around an accuracy of 0.5 (an average of 0.51174), we decide to run an experiment. We begin with a portfolio holding $100,000 in (virtual) cash. Beginning December 11th, 2019, we use the returns of the great majority of the stocks in the S&P 500 to predict the returns for each of the top 10 stocks in the S&P 500, identifying the stock we believe will give the greatest return the following day. We then proceed to invest all of our cash into that particular stock when the market opens of the following day and liquidate our holdings at close. We update the new value of our portfolio by observing the actual returns of that day. We continue this procedure every day the stock market is open until May 28th, 2021. The final value of our portfolio is \$324,745.72, indicating a return of approximately 224.75%. To give more significance to our results, we now conduct a hypothesis test at a significance level (α) of 0.01. Our null hypothesis (H_0) states that the value of our final portfolio using our predicted returns is less than or equal to the average value of the final portfolio using randomly selected stocks each day. Our alternative hypothesis (H_a) states that the value of our final portfolio using our predicted returns is greater than the average value of the final portfolio using randomly selected stocks each day. Using 1000 samples (df = 999), we find the average value of the final portfolio using randomly selected stocks each day to be \$198,911.28 while the corresponding standard deviation is 2387.73011. As a result, the t-statistic we find is 52.70, which, given the degrees of freedom, corresponds to a p-value of $5.05E-291$. Since our p-value is lower than α, we reject the null hypothesis at a significance level of 0.01.

D. Time Series Forecasting

We plot the predicted returns using linear regression model, and we show two examples of using Microsoft (MSFT) to predict Apple (AAPL) and using JP Morgan (JPM) to predict Berkshire Hathaway Class A (BRK-A) in Fig. 5. As we can see, the model is able to predict the general trend as well as the extend that the return increases or decreases by. We also calculate the accuracy of predicting sign of returns as shown in Table 3. As we can see, the prediction accuracy is relatively high.

AAPL	AMZN	BRK-A	FB	GOOGL	JNJ	JPM	MSFT	TSLA	V
0.7514	0.7596	0.7186	0.7431	0.7432	0.6338	0.7514	0.7678	0.6721	0.6339

Table 3: Accuracy of predicting sign (positive or negative) of returns using time series forecasting.

On top of that, we also run a similar experiment as in the neural network method, investing all of our cash into the stock that, according to our prediction, will give us the greatest return the following day. Similarly, we start with \$100,000 in the portfolio, and the final value of our portfolio is \$219,845.74, indicating a return of approximately 119.85%.

E. Discussion

By comparing the results obtained through different models, we can see that using linear regression isn't sufficient enough to give us the best prediction, while multi-layer perceptron neural network as well as time series forecasting are able to predict the stock price relatively well. Time series forecasting results in a higher prediction accuracy in sign of returns compared to using the MLP neural network model. However, both result in significant profit returns in the portfolio using

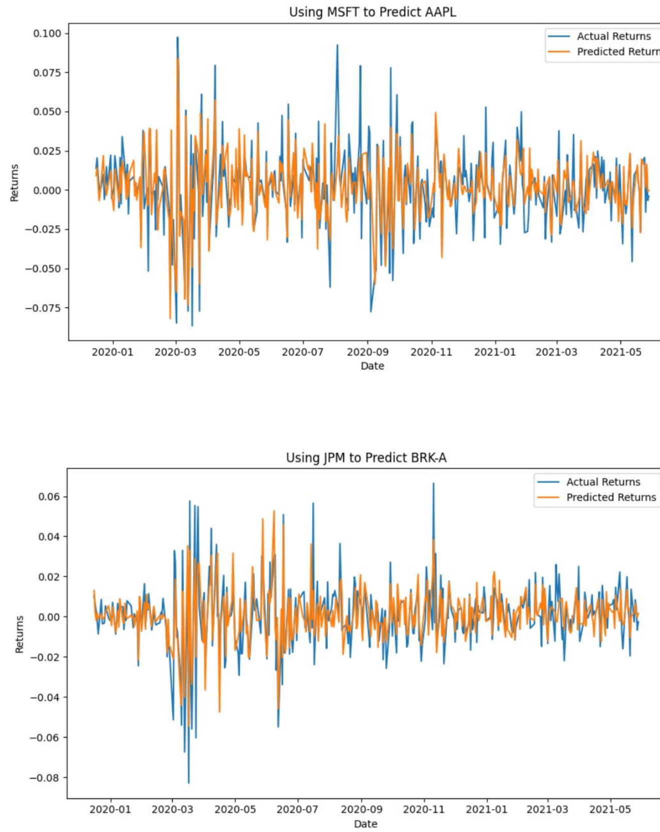

Figure 5: Predicting the subsequent day's return using the most correlated stock's previous day's return.

the neural network having a return of 224.75% and time series forecasting having a return of 119.85%. Our method of mean-variance portfolio optimization also effectively provides us with weights that maximize our profit from stock.

VI. Conclusion

Ultimately, we find that predicting stock market patterns and trends take many challenges to achieve the end profit-making goal for quantitative traders. Utilizing machine learning models like the MLP neural network, time series forecasting, and mean-variance optimization can allow for portfolio optimization and profit-making investments despite unstable predictions in overall price trends.

Future work include creating random forest (RF) and support vector regression (SVR) machine learning models for portfolio optimization, as well as implementing LSTM and convolutional neural networks. Unsupervised learning techniques like k-means clustering can also be implemented to detect closely correlated stocks and formulate optimal portfolios. Performing pre-processing techniques like Principal Components Analysis (PCA) (which we roughly explored throughout this project) to create eigen-portfolios by reducing stock market dimensions could help with higher predictions of overall market trends and benefit the quantitative finance industry.

VII. Contributions

Alina researched on potential experiments and evaluations for the models. She also trained linear regression models predicting the closing stock price as well as logistic regression models predicting the upward and downward trend of the stock price, produced graphs and compared the accuracy across different model predictions. Moreover, she researched and performed time series forecasting on different type of prediction models, generating returns for the top 10 stocks in S&P 500 companies and produced graph. She also evaluated the model by calculating the profit generated.

Tiffany mainly contributed to all portions relevant to portfolio optimization in this paper. She conducted research for the introduction and related work, and wrote code for the implementation of linear regression training pipeline along with metrics. She built a pipeline to use the efficient frontier to determine the weights of the optimal portfolio of stocks and visualize the correlation amongst stocks & tangent portfolios. She also explored uses of PCA for portfolio optimization.

Edgar created a script, featuring an aesthetic user interface to extract the data for all stocks from the S&P 500 from any time period, so that the data sets could simply be fed into our models for training. He also trained the initial linear regression models predicting price for same day and one day ahead and produced the ensuing graphs. Later, he focused most of his attention to training and testing the MLP neural network which predicted the returns of the top 10 stocks in the S&P 500 using the returns of all other stocks in the S&P 500. Finally, he wrote the code to conduct the experiments which tested the profitability of several trained models, and he conducted the hypothesis testing to determine the significance of the final results.

References

[1] D. H. Bailey and M. Lopez de Prado. The sharpe ratio efficient frontier. *Journal of Risk*, 15(2):13, 2012.

[2] G.-Y. Ban, N. El Karoui, and A. E. Lim. Machine learning and portfolio optimization. *Management Science*, 64(3):1136–1154, 2018.

[3] B. Clark, Z. Feinstein, and M. Simaan. A machine learning efficient frontier. *Operations Research Letters*, 48(5):630–634, 2020.

[4] Y. Ma, R. Han, and W. Wang. Portfolio optimization with return prediction using deep learning and machine learning. *Expert Systems with Applications*, 165:113973, 2021.

[5] F. D. Paiva, R. T. N. Cardoso, G. P. Hanaoka, and W. M. Duarte. Decision-making for financial trading: A fusion approach of machine learning and portfolio selection. *Expert Systems with Applications*, 115:635–655, 2019.

[6] M. H. Partovi, M. Caputo, et al. Principal portfolios: Recasting the efficient frontier. *Economics Bulletin*, 7(3):1–10, 2004.