Comparative Analysis of S&P 500 and NASDAQ: A Machine Learning Approach to Understanding Differential Market Sensitivity and Growth Stocks

Cheng Lin[1,*]

[1]School of Finance, Zhejiang University of Finance & Economics, Hangzhou, 310018, China

Abstract. The purpose of this study is to gain a deeper understanding of the dynamics and market sensitivity of technology stocks by constructing and predicting random stock portfolios in the S&P500 and NASDAQ indices using machine learning models. The article employs a rolling time window cross-validation approach to train the models, ensuring optimization based on continuously updated data. By comparing the predicted results with actual outcomes, the article analyze whether technology stocks possess characteristics of growth stocks and how key events impact the price fluctuations of technology stocks. The results also indicate the need to dynamically adjust the learning rate based on market characteristics, volatility, trends, and other factors to enhance the adaptability of the machine learning models in different conditions and better uncover the latent information within diverse data samples. Overall, the findings of this study underscore the importance of dynamically adjusting the learning rate of machine learning models under various market conditions, providing a fresh perspective for financial application research. Future research can further explore and develop cutting-edge methods related to optimizing the learning rate of models, such as adaptive learning rates, multi-task learning, and incremental learning, to more effectively cope with different market situations and similar issues in other fields.

1 Introduction

With the rapid development of financial markets, the analysis and prediction of stock markets have become hot topics of research. In particular, the stocks of technology and growth companies, such as many companies listed on NASDAQ, have attracted widespread attention. Therefore, understanding the dynamics, market sensitivity of these technology stocks, and their differences with other standard stocks, as well as their sensitivity to different market conditions, is crucial [1].

Previous and current research has extensively used various machine learning methods to predict and analyze stock markets. For example, some studies have used neural networks, decision trees, and random forests to predict changes in stock prices or indices [2]. In previous research, it observed that market fluctuations can affect the learning rate of machine learning models. This suggests that different markets may require different learning rates to optimize model performance [3].

Influenced by many factors, the market sensitivity shows a strong temporal change. The learning results of many machine learning models have large deviation and low accuracy, which is caused by the inherent limitations of the corresponding methods to accurately outline the dynamic characteristics of financial market trends [4]. To address this issue, this paper considers adopting validation methods more suited to the characteristics of time series data. Rolling Time Window Cross-Validation is an effective method that accommodates the sequential and dependent nature of time series [5]. In this approach, training sets are adjusted through continuously moving time windows, ensuring that models are always predicated on the most current data.

The aim of this study is to construct and predict random stock portfolios from the S&P 500 and NASDAQ indices using machine learning models, in order to understand the dynamics and market sensitivity of technology stocks, as well as the intrinsic relationship between market sensitivity and the learning rate of machine learning models [6, 7]. Specifically, it employed a walking-forward machine learning model to build and predict investment portfolios randomly selected from the S&P 500 and NASDAQ indices, and hoped to analyze whether technology stocks conform to the characteristics of growth stocks by comparing predicted results with actual results during three significant moments that greatly affected technology stocks. Additionally, it explored how these critical events affected the price fluctuations of technology stocks, the distinct features of technology stocks and standard stocks in prediction, and whether there exists a particular requirement for market sensitivity to the learning rate of machine learning models. It hopes to reveal insights that are beneficial for future research.

* Corresponding author: cheng_lin@zufe.edu.cn

Many machine learning methods in financial applications face the problem of overfitting or underfitting [8]. This is often due to the improper adjustment of the model's learning rate. Therefore, it is necessary to incorporate hyperparameter optimization techniques into the model [7, 9]. While some advanced methods, such as adaptive learning rates and incremental learning, have been proposed to address this issue, how to dynamically adjust for different market conditions remains a challenge [10, 11].

This research conclusion further confirms the viewpoint that it is necessary to dynamically adjust the learning rate according to the characteristics, volatility, trends, and other factors of the market for different market sensitivities. This not only improves the model's adaptive ability under different conditions, but also better explores the underlying patterns of different data samples. Therefore, when training machine learning models, it is necessary to dynamically adjust the learning rate α according to the specific characteristics of different markets to ensure the effectiveness of the model in application. In addition, this study can also promote the development of the machine learning field by demonstrating its effectiveness in financial applications and improving its generalization performance through dynamic adjustment of the learning rate.

2. Methodology

2.1 Data preprocessing

This study first ensured reproducibility of results by setting a random seed, and generated a sample of 50 assets comprising components of both the S&P 500 and NASDAQ 100 indices.

The data collection and processing procedure involved the following steps:

1. Obtaining the composition lists of the S&P 500 and NASDAQ 100 from Wikipedia, and randomly selecting a subset as the sample.
2. Utilizing the yfinance library to download price data for the chosen stocks, covering a time span from January 1st, 2015 to June 30th, 2023.
3. Conducting data cleaning and processing, including calculation of daily returns, and employing box plot methods to handle outlier values, thus ensuring data quality and accuracy.
4. Computing and resampling 20-day returns and daily returns.

Notably, despite successfully downloading data for 50 randomly selected stocks, there were instances of failed downloads due to the time frame within the sample; some stocks were either not listed or had been suspended or delisted.

These processed data form a crucial foundation for the subsequent model construction and portfolio building. This data report provides a clear overview of the data collection, downloading, and processing procedures, laying the groundwork for further research.

2.2 Model establishment and evaluation

The establishment and optimization of the model is a key step in this study. It adopted the Walking Forward Machine Learning method and used Ridge regression model for prediction. The following will detail the process of model establishment and the process of model hyperparameter optimization and evaluation.

2.2.1 Model establishment

This paper first defines a time sliding window method to control the size of the training set. By setting variable sliding window sizes, it can adjust the minimum time period of the training set to ensure that the model has sufficient data for training. Next, it selected a set of hyperparameters (alpha_values) as tuning parameters for the Ridge regression model. These hyperparameters will be adjusted in the subsequent model optimization process. It uses the train_and_evaluate_model function to implement model training and evaluation. The function accepts the data of training set and testing set as inputs, and trains the model by iterating over different alpha values. In each time sliding window, it takes one stock as the target variable (y) and use the returns of other stocks as feature variables (x) to train the model. By adjusting the hyperparameter alpha, it obtained the optimal model with the smallest mean squared error (MSE).

2.2.2 Hyperparameter optimization

This paper optimizes the model by traversing through different training set sizes. For each training set size, it trained the model using the train_and_evaluate_model function and selected the best alpha with the minimum MSE. For each training set size, it calculated the prediction results of the model, and calculated the MSE and the MAE of the prediction results. By comparing the MSE with different alpha values, it chose the best alpha with the minimum MSE as the hyperparameter of the model. At the same time, it also used the MSE and the MAE as evaluation indicators to evaluate the performance of the model and observe the training process of the model by drawing the learning curve.

2.2.3 Model evaluation

This paper used the MSE as a metric to evaluate the model performance. MSE measures the average squared error between the predicted and actual values of the model and can reflect the accuracy and accuracy of the model. Furthermore, it can use other evaluation metrics such as MAE to further evaluate the model performance. MAE measures the mean absolute error between the model predicted and actual values and can provide a more comprehensive assessment of the model error.

Through the above modeling and optimization process, it obtained a predictive model based on S&P500 data, and selected the optimal model through hyperparameter tuning and evaluation metrics. This

model can be used to predict stock returns and provide important references for investment decisions.

2.2.4 Portfolio construction

Based on the predictive results of the model, it can construct investment portfolios. The specific steps include the following:

Using CVXPY library for portfolio optimization: it utilizes predicted stock returns and covariance matrix to optimize investment portfolios. By minimizing the volatility of portfolios and satisfying some constraint conditions such as weight sum of 1 and non-negative weights, it can obtain the optimal weight for investment portfolios.

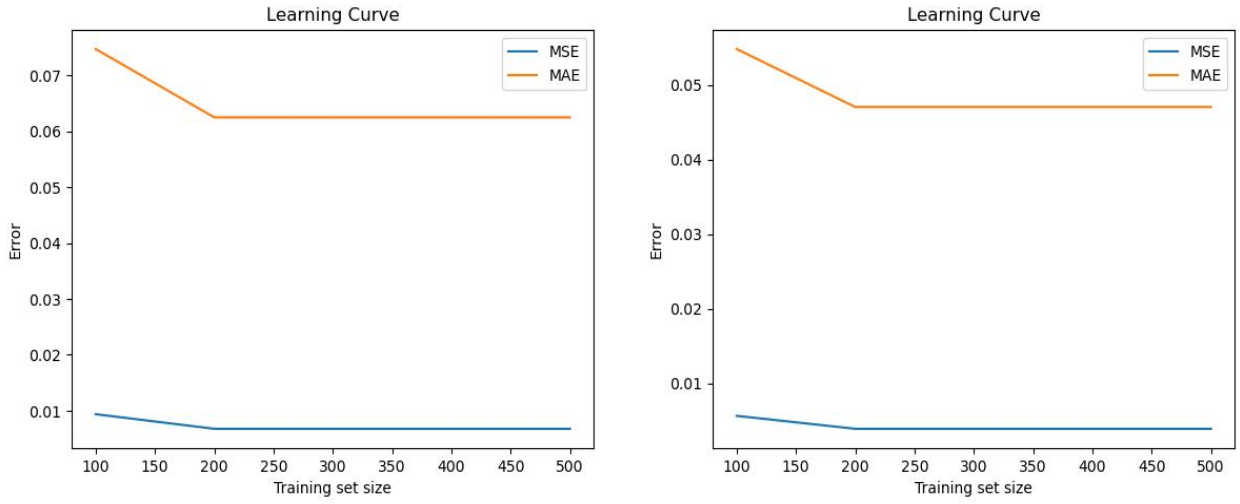
Portfolio backtesting: Apply each time window's portfolio weight to each stock's daily return rate to calculate the cumulative return of the portfolio. At the same time, this paper also compares it with the cumulative return of a benchmark index (such as SPY) to evaluate the performance of the portfolio. Through the above investment portfolio construction process, it can make effective investment decisions based on the predictive results of the model and evaluate the return of investment portfolios.

3 Data analysis

3.1 MSE & MAE in the process modelling evaluation

In the modeling of the S&P 500 index, a comparison between MSE and MAE indicates that after 200 samples, the reduction in MAE becomes minimal, suggesting limited benefits from additional data in terms of decreasing the mean absolute error (see Figure 1). The stability observed in MSE across the sample range may suggest sensitivity to specific patterns or noise within the data, or insufficient information in the training data for further error reduction. Future research should explore strategies to mitigate the impact of noise on the model. The current model's performance appears to have reached a threshold, indicating a need to consider model improvements or alternative approaches for enhancing predictive accuracy.

This phenomenon also applies to return forecasts for the NASDAQ index. Does this indicate that for this method, a sample size of 200 is a limit? To further increase accuracy, this paper recommends expanding the time window.



(a) S&P 500 (b) NASDAQ

Fig. 1. The MSE & MAE learning curve (Photo/Picture credit: Original).

Simultaneously, through different alpha values, it can get the best alpha values adapted to the corresponding data set to represent the best model parameters for model training. It has derived different ridge alpha values for the two indices: 0.01 for the S&P 500 and 1 for NASDAQ. The alpha parameter, also known as the regularization parameter, controls the degree of constraint on the model's weights. Different alpha values affect how closely the model fits the data and its ability to generalize to new data. Thus, it can speculate that NASDAQ stocks, being more heavily weighted towards technology stocks with potentially greater price volatility, require a larger alpha. In contrast, the S&P 500, which includes a broader range of industries, may be relatively more stable and thus require a smaller alpha. This indirectly supports the theory mentioned at the beginning.

3.2 S&P 500 and NASDAQ prediction vs returns display

This paper has developed a code to dynamically display a comparative trend graph of actual versus predicted results for a randomly selected stock from the pool of 50. This chart presents the actual and predicted returns of a particular stock over time. The solid blue line represents the actual returns, while the orange dashed line represents the predicted returns. Here is an analysis of the chart:

Fig. 2. Stock return forecast in SPY index (Photo/Picture credit: Original).

Figure 2 demonstrates the fluctuations of stock return rates over time, indicating a high degree of instability. In some periods, such as early 2018 and early 2021, the prediction aligns closely with the actual trend. However, in other periods, particularly at the beginning of 2020 and mid-2022, there are significant discrepancies between the prediction and the actual values, suggesting that the model fails to accurately predict the stock's performance during these times.

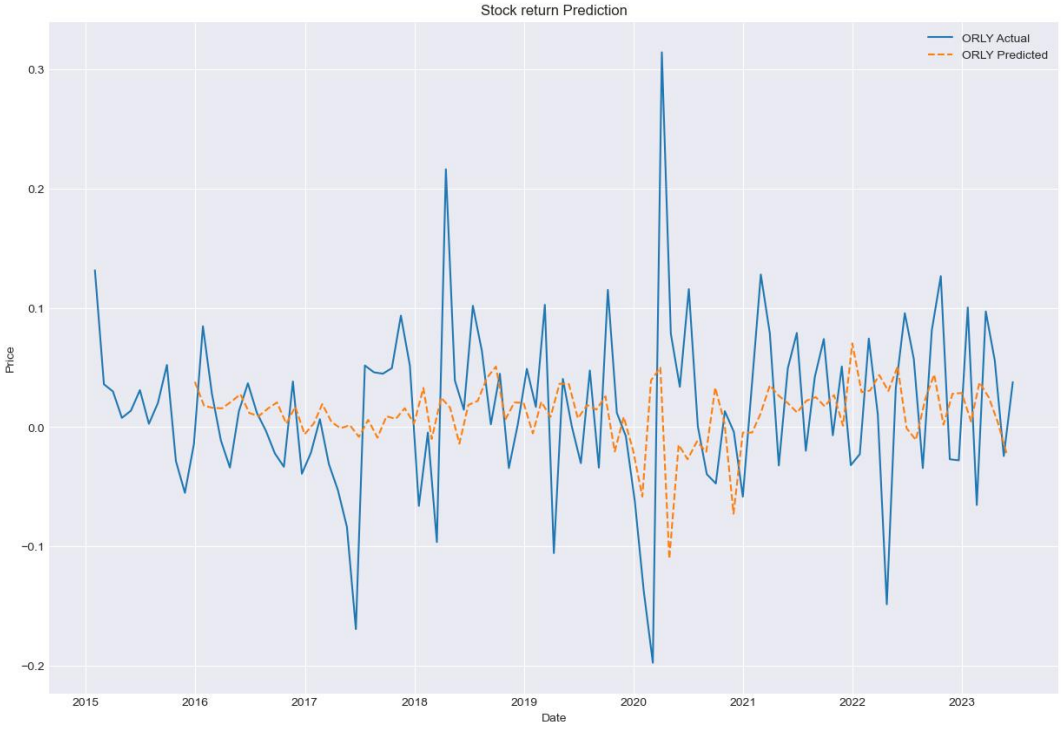


Fig. 3. Stock return forecast in NASDAQ index (Photo/Picture credit: Original).

In comparing predictive outcomes with the SPY index and those with the NASDAQ, it is evident that predictions for the S&P 500 are more accurate (see Figure 3). Observationally, the NASDAQ exhibits significantly greater volatility, a factor not anticipated by the model. This highlights the need for future improvements in the forecasting approach for high-

volatility data, such as the application of differencing methods.

Predictions seem to align with actual data in certain periods. However, there are intervals, especially in early 2020 and early 2022, where the prediction significantly deviates from reality, suggesting that the model failed to accurately foresee the stock's behavior during volatile periods in both the SPY and NASDAQ 100 Indices.

A sharp decline in actual return rates is observable at the beginning of 2020, likely related to market turbulence caused by the global COVID-19 pandemic. The predictive line failed to capture this downturn, possibly due to the model not accounting for the impact of such rare events.

To improve the model's predictive power, it may be necessary to consider more complex time-series models, such as ARIMA or GARCH, or to incorporate macroeconomic indicators as predictive variables. Furthermore, it is essential to consider whether the model includes mechanisms to adapt to extreme market events and whether special modeling for such events is required.

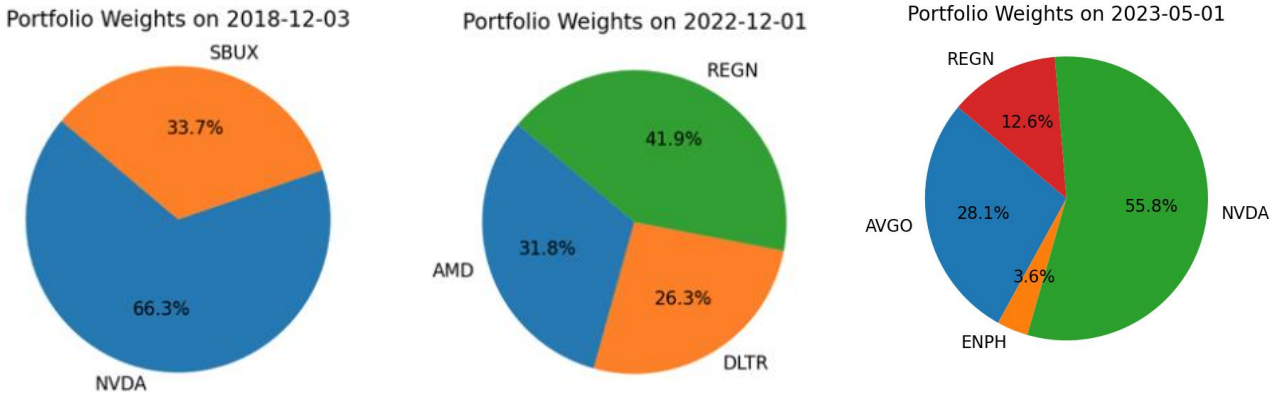
3.3 S&P 500 and NASDAQ portfolio weights pie charts comparison

This paper will select three significant moments that greatly impacted tech stocks: the Facebook lawsuit in December 2018, the public registration opening of OpenAI in November 2022. The White house disclosed that there might be potential further regulation targeting artificial intelligence products in May 2023. It intends to explore how these pivotal events influenced the price movements of tech stocks and evaluate the effectiveness of the Walking Forward Model in predicting the outcomes of these black swan events.

(a) Dec. 3rd, 2018 (b) Dec. 1st, 2022 (c) May 1st, 2023

Fig. 4. S&P 500 weights on 3 time points (Photo/Picture credit: Original).

SPY index portfolio (see Figure 4): Despite the selection of three tech-related milestones, the S&P 500's top ten constituents are tech-centric firms such as Apple, Microsoft, Amazon, Tesla, Alphabet A and C shares, Berkshire Hathaway B, UnitedHealth, Nvidia, and Johnson & Johnson. Yet, none of these tech-linked stocks were included in the portfolios at these specific times. The debut of OpenAI didn't result in a direct inclusion of Microsoft in the portfolio. This could be due to covariance constraints designed to minimize risk rather than increase the weight of tech stocks. During this period, a mining company was allocated instead, a decision that requires further analysis.



(a) Dec. 3rd, 2018 (b) Dec. 1st, 2022 (c) May 1st, 2023

Fig. 5. NASDAQ weights on 3 time points (Photo/Picture credit: Original).

NASDAQ index portfolio (see Figure 5) : At the time of OpenAI's release, the portfolio opted for an 31.8% stake in AMD, a computer chip company, focusing on designing the CPU and other computer components, also chose a retail company and a biotechnology company. Overall, there are no companies

directly related to the OpenAI supply chain. This indicates that the event did not significantly influence the optimal portfolio outcome. Additionally, the forecasting failed to predict the severe volatility trend in tech stocks and the impact of market fluctuations, leading to a discrepancy between the projected and actualized portfolio returns. At all three time points, despite including some technology-related stocks, more than 50% of the portfolio composition is actually unrelated to OpenAI. Overall, the portfolio tends to rely on companies with a tangible economy, such as e-commerce retail, energy development, and branded coffee

3.4 Portfolio compared to benchmark

Figure 6 indicates that the portfolios of both indices have surpassed their benchmark indexes, with the S&P 500 aligned with the SPY ETF and the NASDAQ with the QQQ ETF. However, it harbours doubts about the accuracy of these predictions in practice. This is due in part to questions about whether the forecasted results can genuinely align with actual prices, and the adjustment of the learning rate during the model building process is not enough to have real insight into the underlying information of the data, and also because the simplicity of the model's factors may not account for market risks, potentially resulting in forecast biases.

(a) S&P 500 (b) NASDAQ

Fig. 6. Portfolio return benchmarking (Photo/Picture credit: Original).

4 Conclusion

Through the simulations and forecasts, this paper has substantiated the initial hypothesis that technology stocks exhibit greater volatility. This study of three specific time points indicates that these events have an impact on short-term fluctuations but do not affect the long-term investment strategy. The portfolio still favors companies with solid industrial foundations in the long run, such as those in energy, brick-and-mortar e-commerce, automotive manufacturing, and the healthcare sector. This outcome may be attributed to the use of covariance constraints to hedge risks, leading to the exclusion of stocks with high returns but also high risks.

In addition, the results show that facing the sensitivity of different markets, it is necessary to make dynamic fine adjustment to the learning rate based on market characteristics, volatility, trend and other factors, so as to improve the automatic fitness of the model under different conditions, so as to better mine the underlying information of different data samples. Therefore, in the training process of the machine learning model, the learning rate alpha should be dynamically adjusted according to the specific characteristics of different markets to ensure the effectiveness of the model in its application. This has important implications for investors, policy makers, and market analysts who can use these insights to develop more effective investment strategies, make informed decisions, and better understand market dynamics.

Overall, the results of this paper emphasize the importance of dynamically adjusting the learning rate of machine learning models to improve the model generalization performance in various market conditions. This provides a new perspective for future research on financial applications, where the model performance can be optimized by more finely adjusting the learning rate. In the future, more research is expected to focus on the application and development of some relevant cutting-edge methods, such as adaptive learning rate, multi-task learning, incremental learning and other methods, to adjust the learning rate of machine learning models more effectively, so as to better cope with different market situations and similar problems in other fields.

References

1. R. Zhang, SF 11, 5 (2018).
2. D. Shah, H. Isah, F. Zulkernine, IJFR, 7 (2), 26 (2019).
3. Y. Wu, et al., *Unmysterious the learning rate strategy for high-precision training of deep neural networks* in the Proceeding of IEEE Big Data (Big Data), 1971-1980 (2019).
4. B. Liu, MET (09), 83-87 (2022).
5. L. Li, F. Noorian, M. Moss, W. Leong, *Rolling window time series prediction using MapReduce* in

the Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IEEE IRI 2014), 757-764 (2014).

6. W. Galbraith, S. Zernov, AFE, 19(13), 1019-1028 (2009).
7. R. Chen, Master's dissertation on CSI 300 prediction model research based on SSA-LSTM-LightGBM, Guangxi University for Nationalities (2023).
8. J. Klaas, Machine learning for finance: principles and practice for financial insiders. Packt Publishing (2019).
9. H. Li, D. Song, J. Kong, et al., Hyperparameter optimization technology evaluation of traditional machine learning model. computer science (2023).
10. S. Yang, AS 11 (2021).
11. A. Folly, *A Brief Survey of Group-based Incremental Learning Algorithms* in the Proceedings of IEEE Seminar Series on Computational Intelligence (SSCI), 339-344 (2019).