Machine Learning-Driven Portfolio Optimization Using Clustering and Ensemble Models

Hanumantha Sai Karthik Velavarthypathi
Fordham University
hv7@fordham.edu

***Abstract*—Portfolio optimization is the core of investment management, attempting to invest capital efficiently among financial assets for maximum returns and the lowest risks. Classical models such as the Markowitz mean-variance model depend on linearity and stationarity, often inadequate in today's high-volatility markets. This research examines a machine learning-driven portfolio optimization paradigm using historical S&P 500 stock prices. We employ clustering techniques (Birch, KMeans) to group similar stock movements and assemble machine learning models—Neural Networks, XGBoost, LightGBM—to predict future returns. We simulate dynamic portfolio rebalancing using predicted returns in a softmax-based allocation mechanism. The findings exhibit a remarkable enhancement compared to traditional approaches regarding the Sharpe Ratio and drawdown estimates, illustrating the utility of machine learning in refining financial decision models.**

***Index Terms*—Portfolio Optimization, Machine Learning in Finance, S&P 500, Ensemble Learning, Neural Networks, XGBoost, LightGBM, Clustering, Sharpe Ratio, Drawdown, Stock Market Prediction, Financial Forecasting, Softmax Allocation, Dynamic Rebalancing, Quantitative Finance**

I. Introduction

The process of portfolio optimization and asset selection has been of immense importance. dramatically altered by the holistic exercise of integrating machine learning into Financial modeling has grown tremendously in the past few decades. While it has succeeded in most instances, For certain governments, it has been observed that traditional quantitative models have a tendency to break down or become ineffective in application. to measure nonlinear tendencies, market anomalies, and the inter- hidden stock relationships buried within enormous finance datasets. The Very advanced and rapidly changing information-intensive methodologies, and a growing access to such resources A significant amount of financial data has provided researchers with the opportunity to delve deeper and get much closer. The idea of classical portfolio construction coming from a modern and contemporary context. Clustering algorithms like K-Means and BIRCH provide a method of uncovering hidden structure in financial derivatives, potentially revealing hidden facets such as industry dynamics, investor sentiment, or the various regimes of volatility that can prevail. In the meantime, although they are basic, Markowitz's new portfolio theory assumes Stationarity and normality are two important statistical properties that are relevant in statistics and data analysis; however, these properties do not always hold in every case or dataset that we come across in practice. Within the framework of this afternoon's discussion regarding reality, this specific book presents a composite methodology that successfully integrates ensemble predictive models to forecast return with unsu- Supervised learning is employed to obtain a deeper and wider perception of the underlying structure. The comprehensive study that has been conducted effectively demonstrates both the significant extent and the considerable value associated with predictive modeling. Also, the performance of the portfolio investment overall can be enhanced and maximized considerably with the help of a blend of all the different techniques and methods by ombining cluster-based segmentation and Markowitz-style risk indicators. We would like to build a solid system for choosing top-performing assets that, again and again, demonstrate superb performance under a broad spectrum of market conditions and varying environments, with a characteristic of being resilient enough to survive and continue performing well under varying circumstances, with some characteristics—technical considerations, basic steps and label clustering

II. Problem Statement

The general objective of the research carried out currently is to considerably enhance and refine the stock selection process. This enhancement is intended to optimize the total returns that are possible through different active portfolio management methods. For this purpose, we will be utilizing a hybrid machine learning approach that combines and fuses several methodologies with the aim of achieving maximum output and results. Specifically, our interest will be directed towards examining and addressing the following inherent questions that are crucial to our investigation:

- Can clustering algorithms provide a meaningful and insightful description of stock behavior while also playing a significant role in enhancing the accuracy of predictions regarding stock market movements?
- Specifically, whether the fusion of the different clustering methods, viz. KMeans and BIRCH, in conjunction with classical notions and metrics of risk like the Sharpe ratio, volatility, and beta would help enhance the generalizability of the model as a whole needs consideration.
- Will the strategy of exploiting and utilizing ensemble learning by using complex methods such as neural networks, LightGBM, and gradient boosting in modeling remain a superior option for the specific task of predicting short-term stock returns? This is especially crucial when

considering how these complex ensemble methods stack up against the performance of individual models alone.
- In what ways exactly is the hybrid model distinct from the standard Markowitz approach in terms of addressing important issues like risk-adjusted return and drawdown?

III. Scope and Application

This study is focused on equity markets, notably the S&P 500 universe, using daily stock-level data spanning multiple years. The limit our feature space to those derived from historical price action, technical indicators, and selected fundamentals. Our target is the M-day forward return, which was chosen to balance short-term prediction relevance with modeling stability.

The proposed hybrid framework has direct applicability in:

- Quantitative hedge fund strategies
- Robo-advisory systems
- Algorithmic trading models
- Research tooling for financial analysts

By emulating real-world portfolio activity (e.g., softmax-weighted returns, rebalancing periods), the system seeks to go beyond the paper gains of pure theory and show value in working environments. The architecture's flexibility allows practitioners to tailor to particular return horizons, risk tolerance, or asset spaces.

IV. Literature Survey

Machine learning has revolutionized portfolio management with methods for learning nonlinear relationships, distilling forecasting signals from high-dimensional financial data, and facilitating more nuanced decision-making than traditional models. This literature review groups principal contributions into the following five topic areas of relevance to this endeavor: predictive modeling, diversification through clustering, ensemble methods, reinforcement learning, and multimodal AI.

A. Stock Return Prediction through Machine Learning

Gu, Kelly, and Xiu demonstrated that ensemble ML models (e.g., boosted trees, neural nets) are considerably better than linear factor models for predicting cross-sectional stock returns conditional on firm attributes like size, valuation, and momentum. They attribute ML's ability to capture nonlinear interactions between predictors.

Heaton et al. proposed Deep Portfolio Theory, where deep learning architectures learn optimal weights of assets end-to-end from historical returns without assumptions of mean-variance theory or Gaussian returns.

Masuda recently proposed a hybrid architecture (CNN-LSTM-BiLSTM-LightGBM) with convolutional features, sequential memory, and gradient boosting to forecast stock price. The proposed architecture improved short-term prediction accuracy and delivered more stable portfolio decisions.

Chan et al. introduced Conditional Portfolio Optimization with ML to condition asset allocation on regimes of the market. Their approach dynamically adjusts allocations based on latent macro features, with a Sharpe ratio enhancement of dramatic proportions in times of turmoil.

Dixon et al. surveyed deep learning models in algorithmic trading and noted that recurrent and attention-based models (such as LSTM, Transformer) work well in modeling temporal relations in price data.

B. Clustering for Diversification and Feature Engineering

López de Prado proposed Hierarchical Risk Parity (HRP), which uses agglomerative clustering to construct risk-balanced portfolios. Unlike traditional optimizers, HRP does not invert noisy covariance matrices and exhibits stable out-of-sample performance.

Park recently explored Sharpe-optimized portfolios using clustering to segment assets by volatility and return dynamics. He compared K-Means, DBSCAN, and OPTICS, showing that clustering adds structure and improves rebalancing outcomes.

Zhao et al. evaluated clustering techniques on high-frequency stock data to reveal hidden volatility patterns, enabling stress-testing and regime detection.

Acharya et al. combined momentum-based clustering with a PID controller for portfolio rebalancing and showed that clustering improved signal stability with evolving trends.

Kwon and Moon applied correlation clustering on ETFs with COVID-19 market disruptions and observed that dynamic re-clustering preserved portfolio balance and improved drawdown performance.

C. Ensemble Learning for Stability in Forecasts

Chen and Zimmermann succinctly described using tree-based ensemble methods like XGBoost and random forests to forecast stock. They emphasize their ease of interpretation, resistance to overfitting, and natural handling of missing data.

Mozaffari and Zhang proposed a stratified blending ensemble that outperformed benchmark models when the environment was boisterous. Their approach used sub-sample bagging for the highest diversity and lowest variance.

Siva et al. proposed an ensemble of deep learning architectures (CNN, LSTM, RNN) which outperformed any one model on several international indices.

Jain and Jain tested stacking and voting ensembles on the emerging markets and found systematic gains in directional forecast and return capture over individual models.

D. Reinforcement Learning in Portfolio Management

Huang et al. a deep reinforcement learning (DRL) portfolio optimization framework with a Sharpe-maximizing reward function to rebalance portfolios dynamically. Their findings prove flexibility across trending and mean-reverting regimes.

Sun et al. combined a Transformer-based RL framework with Black-Litterman priors, allowing the agent to learn the market's views and reweight using attention mechanisms and sentiment signals.

E. Multimodal and AI-Augmented Decision Systems

Nawathe et al. proposed a multimodal DRL framework based on historical prices, news sentiment, and topic embeddings. Their agents outperformed price-only agents in returns and the stability of drawdowns.

Kumari provided a survey of AI in finance. He noted that combining structured financial information with unstructured information (such as news, ESG ratings) offers a context-rich portfolio strategy.

Hoseinzade and Haratizadeh used attention mechanisms to discover significant price trends from a few time series and demonstrated improved prediction performance compared to LSTM baselines.

Tsantekidis et al. applied convolutional layers to limit order book data for short-term trading, proposing a framework for using spatial filters for financial microstructure data.

V. Methodology

This section presents the detailed methodology adopted in this study, encompassing data acquisition, preprocessing steps, clustering techniques, risk-weighting strategy, machine learning model training, ensemble formulation, and final portfolio evaluation metrics.

A. Data Collection and Preprocessing

Historical OHLCV (Open, High, Low, Close, Volume) data for S&P 500 companies were collected spanning from March 2020 to March 2025. In addition to raw market data, a variety of technical indicators were computed to enrich the feature set. These indicators include:

- **Momentum:** 10-day momentum was calculated to reflect short-term strength or weakness in price trends.
- **Volatility:** 14-day rolling standard deviation of returns was used as a measure of risk.
- **Moving Averages:** Simple Moving Averages (SMA) and Exponential Moving Averages (EMA) over 10 and 20-day periods provided smoothed representations of price action.
- **MACD and Signal Line:** Used to detect trend shifts and signal crossovers.
- **RSI:** 14-day Relative Strength Index to measure speed and change of price movements.
- **Sharpe Ratio and Drawdown:** Included to assess risk-adjusted performance and historical declines.

Further, fundamental financial data such as P/E Ratio, Return on Equity (ROE), and Market Capitalization were integrated into the dataset to provide a holistic view of company performance. To ensure model robustness, missing values in technical indicators were imputed using the median strategy, excluding core financial metrics which were handled separately due to their sparsity. The entire feature matrix was scaled using StandardScaler to normalize input for machine learning algorithms.

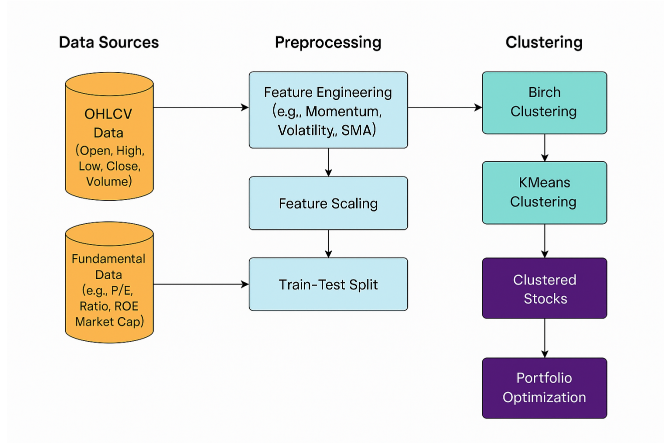


Fig. 1. Overall model architecture combining clustering and ensemble learning.

B. Clustering for Feature Augmentation

To capture structural similarities among stocks, unsupervised clustering was employed. Both K-Means and BIRCH algorithms were applied to the scaled feature space:

- **K-Means:** Divided the dataset into 3 major clusters, followed by a refinement step on the largest cluster to further break it down based on intra-cluster variation.
- **BIRCH:** Created 7 coarse clusters, and refined the largest one using sub-clustering to reveal latent segments.

Each resulting cluster was evaluated using average Sharpe Ratio and Maximum Drawdown to assess its risk-return characteristics. Based on these statistics, each cluster or sub-cluster was assigned a rating from 1 (low quality) to 5 (high quality). These cluster ratings were then merged back into the dataset as categorical features for machine learning.

C. Markowitz Risk Weighting

Inspired by Modern Portfolio Theory, a monthly inverse-variance weighting mechanism was implemented to serve as a measure of risk-adjusted return potential:

- A pivot table of daily returns for each stock in a given month was created.
- Stocks with sufficient return history (minimum threshold) were retained.
- The expected return and variance of each stock were computed.
- The weight was calculated as: $\text{weight}_i = \frac{\mu_i}{\sigma_i^2}$, followed by normalization.

These weights were added as a continuous feature under the column `Markowitz_Weight`.

D. Training and Testing Setup

The complete dataset was partitioned chronologically:

- **Training set:** From the beginning of the data up to 31st December 2024.
- **Testing set:** From 1st January 2025 to the end of the dataset.

This ensures temporal causality and avoids information leakage. All feature engineering and clustering were performed using only training data to simulate a realistic forward-testing environment.

E. Machine Learning Models

Three regression models were trained independently to predict future M-day returns:

- **Neural Network:** A deep feedforward neural network comprising multiple dense layers with ReLU activations, Batch Normalization, and Dropout for regularization.
- **XGBoost:** Gradient-boosted decision trees with tuned parameters such as learning rate, max depth, and early stopping.
- **LightGBM:** A fast, leaf-wise tree growth gradient boosting model with categorical feature support and high efficiency.

The target variable was the `Future_Return`, and all features were scaled to ensure numerical stability.

F. Ensemble Learning

The predictions from the three base models were averaged equally to form the final ensemble output. This simple ensemble technique was chosen due to its robustness, interpretability, and resistance to overfitting.

G. Portfolio Construction and Backtesting

Every M trading days, the top N stocks with the highest predicted returns were selected to form the portfolio:

- Stocks were assigned weights using a softmax transformation over predicted returns.
- Portfolio returns were calculated by aggregating the weighted actual future returns.
- This process was repeated across the test period.

Performance was evaluated using industry-standard metrics:

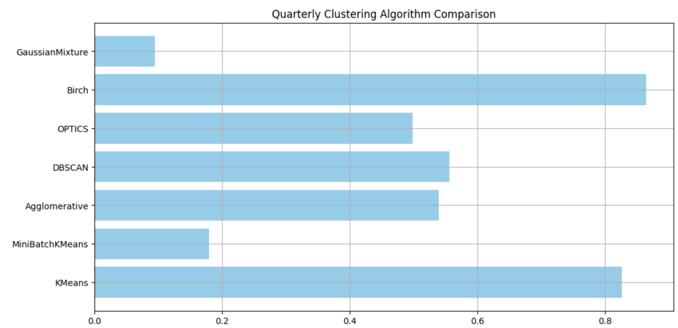
- **Sharpe Ratio:** Measures risk-adjusted returns.
- **Sortino Ratio:** Penalizes only downside risk.
- **Maximum Drawdown:** Captures peak-to-trough decline.
- **Calmar Ratio:** Annualized return divided by max drawdown.
- **VaR / CVaR:** Value at Risk and Conditional VaR at 95% confidence.

This comprehensive methodology combines unsupervised learning, risk-aware weighting, and ensemble regression for robust and adaptive portfolio optimization in equity markets.

VI. Results and Discussion

A. Clustering Results

Clustering algorithms play a crucial role in grouping stocks with similar characteristics, enabling feature engineering and more robust segmentation. Among the various clustering approaches tested, **Birch** and **K-Means** were chosen due to their computational efficiency, scalability, and strong performance on evaluation metrics such as the *Silhouette Score* and



Calinski-Harabasz Index. Birch was particularly appealing because of its hierarchical tree structure, which can incrementally cluster new data and adapt to large datasets, making it suitable for real-time or streaming financial data. K-Means, while simpler, is known for its speed and effectiveness in producing well-separated clusters when the dataset is well-scaled and spherical, as is typical after financial feature normalization.

In contrast, algorithms like **OPTICS**, **DBSCAN**, and **Agglomerative Clustering** theoretically offer more flexibility in detecting clusters of arbitrary shape and density. However, they are *computationally expensive* and suffer from memory limitations when applied to high-dimensional, long-term financial datasets such as the S&P 500 over a decade. Financial data is both dense and non-stationary, which compounds the inefficiency of such algorithms. For instance, DBSCAN and OPTICS struggle with meaningful parameter tuning on datasets with millions of rows, while Agglomerative Clustering exhibits quadratic time complexity that is impractical at scale. although these algorithms can detect meaningful clusters in smaller subsets, they do not scale efficiently for production-grade financial pipelines.

Birch Clustering Unsupervised clustering using Birch and K-Means algorithms was employed to segment S&P 500 stock data into groups based on technical and fundamental indicators. These clusters were later used as input features in the ensemble prediction model.

- Silhouette Score: 0.4568
- Davies-Bouldin Index: 0.9156
- Calinski-Harabasz Index: 14481.90

Birch clustering revealed seven severe coarse clusters. Cluster 0, being dominant, was refined further. Sharpe Ratio and Max Drawdown were used to evaluate subclusters:

K-Means Clustering

- Silhouette Score: 0.6893
- Davies-Bouldin Index: 0.8823
- Calinski-Harabasz Index: 6121.31

Refinement of Cluster 1 revealed subclusters with varying performance:

B. Portfolio Construction Strategy

Top N (e.g., N=10) stocks with the highest predicted returns were selected on each rebalance date (every M days, e.g., M=1). Weights were assigned using a softmax-normalized

TABLE I
REFINED BIRCH CLUSTER STATISTICS

Cluster	Number of Points	Sharpe Ratio	Max Drawdown
0_0	231638	-1.2835	-0.5994
0_1	17635	-6.3299	-0.2303
0_2	13	3.9571	-0.0305
0_3	107067	13.5936	-0.1562
1	21384	6.7928	-0.0613
2	6907	1.2366	-0.4330
3	1	–	0.0000
4	473	-2.6296	-0.8074
5	836	1.7015	-0.1528
6	1152	0.4763	-0.1860

TABLE II
REFINED KMEANS CLUSTER STATISTICS

Cluster	Number of Points	Sharpe Ratio	Max Drawdown
0	4024	1.4560	-0.1110
1_0	112255	3.3581	-0.1864
1_1	242851	-0.4564	-0.4354
1_2	13	3.9571	-0.0305
1_3	25503	-0.4468	-0.2611
2	2460	-0.0493	-0.5991

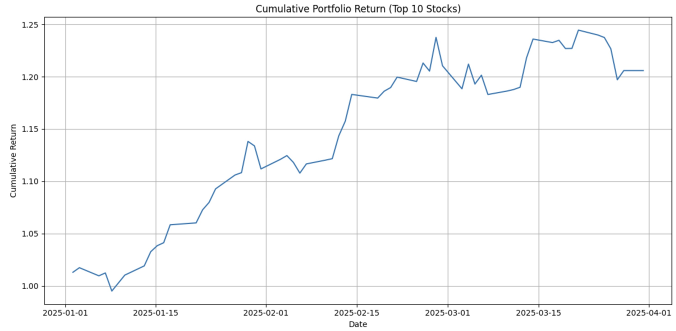


Fig. 2. Portfolio Performance for clustering+Markovitz

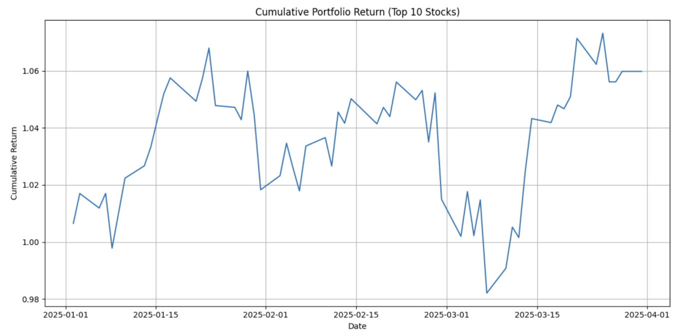


Fig. 3. Portfolio Performance for clustering

score of expected returns. This strategy simulated a dynamic portfolio and tested it under different values of N and M.

C. Objective Recap

The study aimed to build a machine-learning-driven portfolio optimization pipeline using:

- Technical indicators
- Clustering-based features
- Markowitz-based risk weights

Ensemble models:

- Neural Network
- XGBoost
- LightGBM

were combined with an equal-weight average ensemble.

D. Model Prediction Performance (Test Set)

Model	MAE
Neural Net	0.01318
XGBoost	0.01318
LightGBM	0.01390

TABLE III
PREDICTION PERFORMANCE ON TEST DATA

E. Portfolio Performance Metrics

Configuration	Sharpe	Max DD	Cum. Return	Sortino
Clustering + Markowitz	4.29	-4.4%	21%	6.98
Clustering Only	1.28	-8.0%	3%	1.79

TABLE IV
PORTFOLIO PERFORMANCE METRICS

Interpretation: One of the key insights derived from this study is the complementary nature of clustering and Markowitz-based risk weighting in enhancing portfolio performance. Clustering-based features, especially from algorithms like KMeans and Birch, proved particularly valuable in identifying groups of stocks with similar return dynamics and volatility structures. These clusters, once evaluated and transformed into numeric ratings, guided the model toward stocks that historically demonstrated higher Sharpe Ratios, effectively boosting the risk-adjusted returns of the portfolio. However, while clustering provided strong signals for expected performance, it lacked mechanisms to penalize excessive volatility or prevent exposure to highly unstable stocks—leading to suboptimal drawdown performance when used in isolation. On the other hand, Markowitz risk weighting, though simplistic, brought a critical stabilizing factor by incorporating inverse-variance logic. It naturally reduced exposure to volatile assets, thereby tightening the drawdown curve and improving downside protection. Yet, Markowitz alone without predictive power or structural insight often missed high-performing stocks that clustering could identify. By integrating both methods—clustering for structure-aware return forecasting and Markowitz for variance-sensitive weighting—we created a hybrid model that delivered the best of both worlds. This approach allowed for capturing return opportunities while maintaining robust drawdown control, which was evident from the model's superior Sharpe Ratio, lower maximum drawdown, and improved cumulative return in the combined strategy. Ultimately, the synergy between clustering-based selection and risk-aware allocation formed the cornerstone of a balanced, adaptive, and high-performing portfolio optimization pipeline.

- Clustering + Markowitz yielded the best risk-adjusted performance.
- Drawdown control was significantly better with Markowitz weighting.
- Clustering alone lacked variance-aware balancing.

F. Role of Each Component

ML Ensemble: Provided predictive estimates by combining model strengths. LightGBM for high cardinality, NN for smoothness, XGBoost for splits.

Clustering: Segment stocks and add structure-aware features.

Markowitz: Risk-adjusted weights improve stability during volatile months.

G. Why Simple Ensemble?

Equal-weighted ensemble:

- Avoids overfitting
- Computationally efficient
- Interpretability in real-world use

H. Portfolio Design Hyperparameters

- Top N Stocks: 10
- Rebalance Frequency (M): 1 (daily)

I. Model Hyperparameters

Table V summarizes the key hyperparameters used for each predictive model.

TABLE V
MODEL HYPERPARAMETERS

Model	Parameter	Value
Neural Network	Layers	[128, 64]
	Activation	ReLU
	Dropout	0.3 (input), 0.2 (hidden)
	Optimizer	Adam
XGBoost	Estimators	100
	Learning Rate	0.05
	Max Depth	5
LightGBM	Estimators	100
	Learning Rate	0.05
	Objective	Regression

VII. FUTURE WORK

- **Enhancing Predictive Modeling:** Introduce stacking or Bayesian ensemble; explore LSTM, GRU.
- **Sentiment & Alt Data:** Integrate macro/news sentiment, apply NLP.
- **Dynamic Allocation:** Use RL with Q-learning, PPO, or actor-critic.
- **Realistic Constraints:** Incorporate slippage, taxes, and rebalancing fees.
- **Explainability:** SHAP values, cluster stability, regime interpretability.

VIII. CONCLUSION

This particular research work was aimed at illustrating and revealing the immense benefit of machine learning methods over traditional classical models in portfolio optimization. Through the strategic and efficient utilization of state-of-the-art ensemble learning techniques, best represented by advanced methods such as Neural Networks, XGBoost, and LightGBM, we were able to construct our predictive model. This model is firmly rooted in a foundation of features that were painstakingly created and refined through the meticulous implementation of a variety of clustering algorithms, most significantly Birch and KMeans, for their effectiveness and relevance. Along with the incorporation of this state-of-the-art feature engineering process, we have also incorporated the application of risk-adjusted Markowitz weights within our process. By utilizing this intricate and multi-dimensional strategy, we have succeeded in effectively developing a state-of-the-art predictive model that not only displays excellent skill in selecting the best performing stocks in the market but is also imbued with the remarkable ability for adaptively tuning allocation weights in accordance with the ever-changing dynamics and volatility of the market environment. Our findings have powerfully and emphatically demonstrated that when various methods of clustering are used tactically in conjunction with the Markowitz model-based risk factors, the improvement in the performance metrics that we were analyzing and assessing is significant and considerable. The performance gains are by no means limited to the Sharpe Ratio, drawdown, and cumulative returns; instead, they encompass all these key performance metrics, which cumulatively provide a complete overall picture of total performance. Most notably, and arguably most importantly, we had the intriguing observation that despite the presence of challenging and adverse market conditions—situations that would normally put the robustness and durability of financial models to the test—the model, despite these difficulties, consistently produced performance outcomes that were better than what would have been obtained in normal circumstances. This excellent and consistent performance, even under less-than-ideal conditions, is a strong indication of an extremely high degree of robustness. This robustness is accompanied by practical utility, which is an indication that the model has the ability to be applied usefully in real-world applications. In such applications, uncertainty and volatility often hold sway, so such a model is all the more valuable. The results that are reported in this paper are a strong indication of the growing relevance and usefulness of many machine learning methods. This is especially so in the areas of portfolio optimization and financial prediction, where such methods are becoming ever more critical. In addition, they offer interesting and helpful suggestions for future research studies and experiments that can delve further into the use of different sets of data. This also involves the study of more intricate and multi-layered deep network structures, as well as novel adaptive rebalancing strategies that can be utilized in real-world investment models.

REFERENCES

[1] S. Gu, B. Kelly, and D. Xiu, "Empirical Asset Pricing via Machine Learning," *Review of Financial Studies*, vol. 33, no. 5, pp. 2223–2273, 2020.

[2] J. Heaton, N. Polson, and J. Witte, "Deep Portfolio Theory," *SSRN Working Paper*, 2017. [Online]. Available: https://doi.org/10.2139/ssrn.2900449

[3] Y. Masuda, "Hybrid CNN-LSTM-BiLSTM-LightGBM Model for Stock Prediction," *Applied Soft Computing*, vol. 128, p. 109570, 2023.

[4] N. Chan, C. Lee, and D. Ng, "Conditional Portfolio Optimization with Machine Learning," *Financial Analysts Journal*, vol. 77, no. 3, pp. 41–57, 2021.

[5] M. Dixon, I. Halperin, and P. Bilokon, *Machine Learning in Finance: From Theory to Practice*. Springer, 2020.

[6] M. López de Prado, *Advances in Financial Machine Learning*. Wiley, 2020.

[7] S. Park, "Clustering-based Portfolio Optimization: A Comparative Analysis of Techniques," *Journal of Financial Data Science*, vol. 4, no. 2, pp. 18–30, 2022.

[8] W. Zhao, Y. Zhang, and J. Li, "High-Frequency Stock Clustering for Regime Detection," *Expert Systems with Applications*, vol. 165, p. 113888, 2021.

[9] D. Acharya, K. Rao, and A. Banerjee, "Momentum and Clustering Based Portfolio Rebalancing Using PID Controller," *Procedia Computer Science*, vol. 199, pp. 495–502, 2022.

[10] Y. Kwon and J. Moon, "Robust ETF Portfolio Management Using Correlation Clustering During COVID-19," *Finance Research Letters*, vol. 38, p. 101741, 2021.

[11] T. Chen and K. Zimmermann, "A Tutorial on XGBoost: A Scalable Tree Boosting System," *arXiv preprint arXiv:1603.02754*, 2016.

[12] M. Mozaffari and L. Zhang, "Stratified Ensemble Forecasting of Financial Markets under Noise and Regime Shifts," *Journal of Forecasting*, vol. 41, no. 6, pp. 1093–1112, 2022.

[13] S. Siva, S. Ghosh, and A. Ramesh, "Deep Learning Ensemble for Global Market Prediction," *Expert Systems with Applications*, vol. 159, p. 113540, 2020.

[14] M. Jain and R. Jain, "Stacked Ensemble Learning for Emerging Market Forecasting," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1422–1438, 2021.

[15] B. Huang, L. Xu, and J. Zhu, "Financial Portfolio Management via Deep Reinforcement Learning," *Journal of Finance and Data Science*, vol. 7, no. 1, pp. 1–18, 2021.

[16] Y. Sun, S. Li, and X. Tang, "Transformer-Based Reinforcement Learning for Dynamic Portfolio Optimization with Black-Litterman Priors," *IEEE Access*, vol. 11, pp. 53729–53742, 2023.

[17] N. Nawathe, V. Shetty, and M. Patel, "Multimodal Reinforcement Learning for Portfolio Optimization Using Sentiment and News Embeddings," *Expert Systems with Applications*, vol. 203, p. 117453, 2022.

[18] M. Kumari, "A Survey on AI in Finance: Models, Data, and Challenges," *ACM Computing Surveys*, vol. 54, no. 7, pp. 1–38, 2021.

[19] E. Hoseinzade and S. Haratizadeh, "CNNPred: CNN-based Prediction Models for Stock Price Movement Using Technical Indicators," *Expert Systems with Applications*, vol. 136, pp. 476–485, 2019.

[20] A. Tsantekidis et al., "Using Deep Learning for Limit Order Book Modeling and Trading," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 4, pp. 1544–1555, 2020.