Reinforcement Learning in Quantitative Wealth Investment Management (QWIM) - Proximal Policy Optimization (PPO)

Melissa Atmaca, Sid Bhatia, Kevin Lochbihler, Alvin Radoncic

April 1, 2024

1 Abstract

This paper delves into the cutting-edge application of Proximal Policy Optimization (PPO), a leading reinforcement learning algorithm, in the realm of Quantitative Wealth & Investment Management (QWIM). By foregrounding the incorporation of deep learning technologies in finance, our research showcases the utility of PPO in mastering the intricacies of financial markets to refine investment strategies. Our methodology harnesses a dual-network structure that includes actor and critic models, enabling a thorough investigation into the algorithm's adeptness at adaptively reallocating portfolios across a wide spectrum of assets, such as exchange-traded funds (ETFs) and stocks. The core of our analysis lies in the PPO agent's training over numerous episodes, utilizing a vast dataset of financial indicators and asset performances. Through meticulous training and validation, we demonstrate the agent's capability to significantly boost portfolio valuations, as evidenced by a remarkable improvement in the Sharpe Ratio, thereby indicating a superior risk-adjusted return profile in contrast to traditional investment methodologies. Notably, our findings reveal that smaller, more concentrated portfolios managed by the PPO agent tend to outperform larger, more diversified ones, achieving higher Sharpe Ratios and demonstrating the agent's effectiveness in seizing short-term market opportunities while managing risk in aggressive growth strategies. This investigation contributes to the expanding landscape of AI-facilitated investment management, highlighting the transformative potential of deep reinforcement learning in forging intelligent, dynamic, and high-performing investment strategies. Beyond illustrating the practical impacts of advanced machine learning on financial decision-making, our work lays a foundational stone for subsequent research efforts aimed at continuously enhancing the quantitative investment management field.

2 Introduction

The current investment landscape is witnessing a seismic shift, propelled by the integration of cutting-edge artificial intelligence (AI) technologies. Reinforcement Learning (RL), and more specifically Deep Reinforcement Learning (DRL), stands at the vanguard of this revolution, offering a novel approach to maneuver through the intricate and unpredictable terrains of financial markets. In this paper, we introduce an avant-garde application of Proximal Policy Optimization (PPO), a cutting-edge RL algorithm, within the sphere of Quantitative Wealth & Investment Management (QWIM). Our methodology employs a dual-network structure, consisting of distinct actor and critic models, tailored to efficaciously learn and adapt investment strategies in response to the ever-evolving market scenarios.

Our research delves into the capabilities of the PPO agent, meticulously trained across numerous episodes, showcasing its proficiency in dynamically modifying portfolio allocations among a diverse array of assets. This includes a comprehensive analysis of extensive financial data, encompassing exchange-traded funds (ETFs) and stocks. The outcomes of our investigation reveal a substantial enhancement in portfolio valuation, underscored by an impressive Sharpe Ratio. This metric signifies a superior risk-adjusted return profile when juxtaposed with conventional investment methodologies. Such a feat is realized through an intricately designed reward mechanism, anchored in the principles of the Sharpe Ratio, ensuring that the agent's decisions are not solely driven by profit motives but are also cognizant of underlying risks.

Contributing to the rapidly growing domain of AI-facilitated investment management, our research illuminates the vast potential of DRL in sculpting intelligent, adaptable, and high-performing investment stratagems. It signifies a pivotal advancement in the evolution of QWIM, laying down a foundational framework for future explorations and applications in this domain. The encouraging outcomes of our PPO-centered model advocate a paradigmatic shift in the realm of investment management. This shift is characterized by an embracement of complexity, not

as an obstacle but as a catalyst for innovation, paving the way towards more efficient, transparent, and lucrative investment strategies. Such a paradigmatic shift, rooted in the fusion of AI and financial acumen, is poised to redefine the contours of modern investment practices, marking the dawn of a new era in wealth and investment management.

3 Literature Review

This literature review critically examines key contributions in the domain of AI-driven investment strategies, particularly those leveraging Deep Reinforcement Learning (DRL). Each study under review has informed our approach, offering insights into the complexities and potentials of modern Quantitative Wealth & Investment Management (QWIM).

Direct Portfolio Construction through AI: Cong et al. pioneered the use of DRL for portfolio construction, integrating AI to enhance return predictions and investment decisions. Our methodology, inspired by this work, incorporates attention-based neural networks to process financial big data, aiming for superior performance in return estimations and investment efficacy.

Dynamic Goals-Based Wealth Management: Das and Varma mark a paradigm shift, favoring a model-free RL approach over traditional backward recursion methods in dynamic programming. Our strategies align with this advancement, adeptly handling evolving investor goals and complex financial state spaces through a more fluid RL framework.

Enhanced Trading Decision-Making: Benhamou et al. contribute significantly to trading bots' decision-making, employing contextual information for more nuanced market analysis. We incorporate similar methodologies to advance realistic market simulations and validate our model through walk-forward analysis, emphasizing stability and long-term effectiveness.

Explaining DRL Strategies in Portfolio Management: Guan and Liu offer crucial insights into interpreting DRL strategies. Their approach, using linear models and integrated gradients, informs our methodology, particularly in enhancing the interpretability and strategic depth of DRL agents in portfolio management.

Adapting to Market Regime Switches: Das et al. illuminate the importance of agility in investment strategies amidst market regime changes. Our model development incorporates these insights, valuing investor flexibility and responsiveness to market dynamics through advanced algorithmic solutions.

Robust Goal-Based Wealth Management: Bauman et al. demonstrate the effective use of DRL in surpassing traditional wealth management benchmarks. Their methodological rigor and implementation of a Markov Decision Process provide a valuable framework for our research, particularly in terms of model benchmarking and validation.

Continuous-Time Mean-Variance Portfolio Selection: The exploration of continuous-time mean-variance portfolio selection in a regime-switching market by Wu and Li, through their POEMV algorithm, introduces a new layer of complexity. This algorithm, catering to unobservable market shifts, informs our approach in considering a broad array of parameters, including those accounting for regime changes.

Cross-Sectional Investment Strategy: Nakagawa, Abe, and Komiyama's RIC-NN framework offers a groundbreaking approach in the predictability of stock returns, demonstrating the potential of DRL in diverse markets. This approach guides our exploration in cross-sectional investment strategy, aiming for sustained performance across varied market conditions.

In aggregating these scholarly inputs, we aim to develop a sophisticated, AI-driven investment strategy that aligns with the latest advancements in QWIM. Our approach, grounded in these diverse yet complementary studies, seeks to navigate the complexities of modern financial markets with enhanced efficiency and clarity.

4 Background Information

4.1 Overview of Reinforcement Learning

Reinforcement learning (RL) is a machine learning paradigm where an agent learns optimal decision-making through active interaction with its environment. The agent performs actions, receiving feedback as rewards or penalties, aiming to maximize cumulative rewards. This process involves exploring the environment, understanding outcomes of actions, and refining decisions based on accumulated experience.

Key concepts in RL include:

- **Agent:** The decision-maker, interacting with the environment, seeking to learn effective actions for maximizing rewards.

- **Environment:** It encompasses all elements the agent encounters, responding to actions by presenting new states and rewards.
- **State:** The current situation or context for decision-making, like market prices and economic indicators in finance.
- **Action:** Decisions made by the agent, such as buying, selling, or holding financial instruments.
- **Reward:** Feedback from the environment guiding the learning process, like financial returns from trades.

The agent's goal is to develop a policy, a strategy for action selection in various states, to maximize rewards. This involves balancing exploration (trying new actions for better strategies) and exploitation (using known information for rewards). RL approaches are categorized into model-based (using an environment model for planning) and model-free (learning directly from experience).

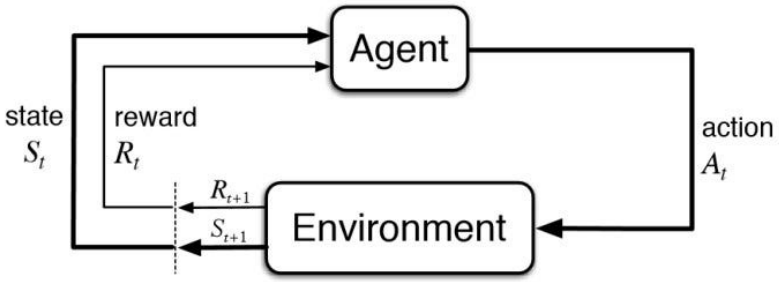

Figure 1: Basic Breakdown of Reinforcement Learning

4.2 Reinforcement Learning Algorithms

1. **Q-Learning:** A model-free algorithm learning optimal actions without requiring an environment model. It uses the Bellman equation for updating Q-values but struggles with scalability and state continuity.
2. **Deep Q Networks (DQN):** Enhances Q-Learning using deep neural networks for Q-value approximation. DQN manages high-dimensional spaces effectively but can suffer from overestimation bias and resource intensity.
3. **Policy Gradient Methods:** Directly learn the policy function to determine actions. Known for better convergence and adaptability, these methods, like REINFORCE, can exhibit high variance and sensitivity to initial parameters.
4. **Actor-Critic Methods:** Utilize an actor for action decision and a critic for action evaluation. Suitable for complex markets, these methods require careful tuning and are computationally demanding.
5. **Proximal Policy Optimization (PPO):** Known for conservative policy updates, promoting stable learning. PPO balances sample efficiency and complexity but requires intricate hyperparameter tuning.
6. **Monte Carlo Tree Search (MCTS):** Builds a tree of potential actions, using simulation for value estimation. Ideal for sequential decision-making but computationally intensive and reliant on accurate simulation models.

These algorithms represent the diverse arsenal of RL, each with unique strengths and applications, particularly in the dynamic and challenging field of quantitative finance.

5 Methodology

5.1 Model Architecture

Proximal Policy Optimization (PPO) stands out for its suitability in tasks involving continuous action spaces, making it an ideal algorithm for financial applications, particularly portfolio optimization. PPO's primary focus is on maintaining stable policy updates, a crucial feature in financial contexts where abrupt shifts in investment strategies can entail significant risks. Additionally, its sample efficiency, advantageous in scenarios with extensive historical

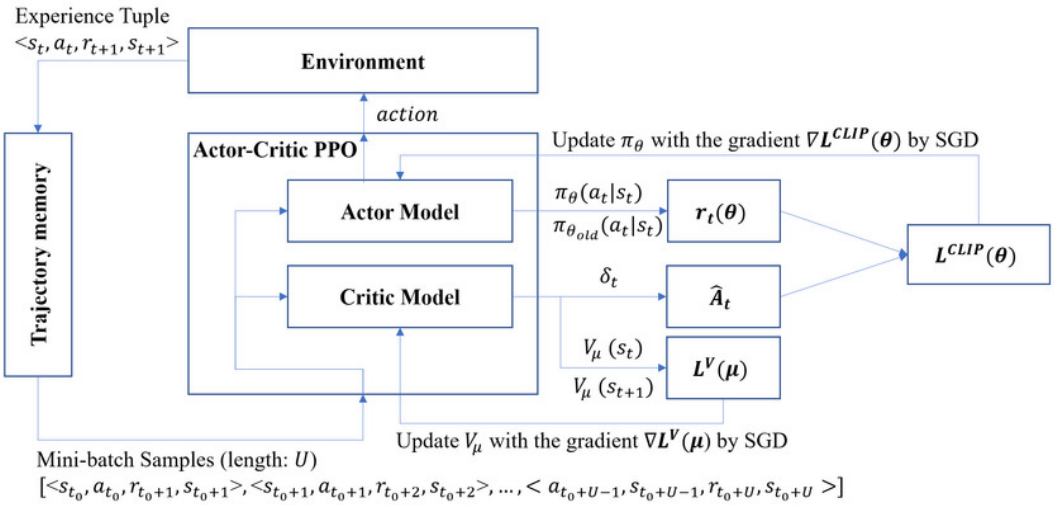

Figure 2: Actor-Critic Proximal Policy Optimization (Actor-Critic PPO) Process

data, enables effective adaptation and learning, solidifying PPO's role in developing advanced investment strategies in finance's dynamic and unpredictable landscape.

Figure 2 illustrates the PPO algorithm within the Actor-Critic framework. Here, the agent generates experience tuples (state, action, reward, subsequent state) through its interaction with the environment, stored in trajectory memory. The PPO architecture comprises two main models: the Actor, suggesting actions based on the current policy, and the Critic, evaluating state transitions. The algorithm incrementally updates these models using mini-batch samples from experiences, guided by the advantage estimate and the temporal difference error for the Actor and Critic, respectively. PPO employs a policy ratio within its clipping objective, moderating policy updates for training stability. Both models undergo iterative improvement through stochastic gradient descent, focusing on maximizing rewards while maintaining policy stability and efficiency.

Figure 3 compares baseline algorithms relative to PPO in terms of definitions, theory/underlying mathematics, the combination approach, stability & efficiency, and finally the methodology of learning as well as overall effectiveness.

Feature	Advantage Actor-Critic (A2C)	Random Forest Classifier	Logistic Regression
Definition	A deep reinforcement learning strategy with an **actor model** for action selection and a **critic model** for action evaluation.	An ensemble learning method using **multiple decision trees** for improved prediction accuracy and over-fitting control.	A statistical method **for binary classification**, predicting the probability of outcomes based on predictor variables.
Theory/Math	Utilizes the advantage function **A(s,a)=Q(s,a) − V(s)** for decision-making, combining policy-based and value-based approaches.	Builds numerous decision trees on sub-samples of the dataset, using averaging to improve accuracy and control over-fitting.	Uses the logistic (sigmoid) function **$S(x) = 1 / (1 + e^{-x})$** to model the probability that a given input belongs to a specific class.
Combination Approach	Integrates direct action selection (policy-based) with evaluative feedback (value-based) for decision-making.	Combines the simplicity of decision trees with the robustness of ensemble learning.	Adapts linear regression to **classification problems** using a sigmoid function for nonlinear transformation.
Stability & Efficiency	Enhances stability by simultaneously updating policies and value functions, but may be less efficient than some methods like PPO due to **no clipping**.	Less prone to overfitting than individual decision trees due to **prediction averaging**, offering stable and robust predictions.	Offers straightforward and computationally efficient modeling for binary classification, easy to implement and fast to train.
Learning & Comparison to PPO	Lacks PPO's policy update clipping, potentially making it less efficient in some scenarios.	Excelling in static prediction tasks with accuracy and robustness, unlike PPO which is designed for dynamic environments.	Suited for predicting binary outcomes without the need for adapting to changing conditions, unlike more complex models.

Figure 3: Comparison of Algorithms Relative to PPO

5.2 Model Implementation

5.2.1 PPO.py

The Proximal Policy Optimization (PPO) script serves as a reinforcement learning algorithm designed to optimize decision-making policies, striking a balance between exploration and exploitation. The architecture of PPO comprises two main components: the Actor Network, which determines action probabilities based on the current state and aims to maximize expected rewards, and the Critic Network, which evaluates the value of being in a given state and predicts the expected sum of rewards. Training in PPO involves the calculation of policy and value loss to guide the actor towards improved policies and refine the critic's value estimation. Additional features of the script include utility functions for calculating discounted rewards and advantages, crucial for the agent's learning and decision-making process. This implementation of PPO balances the exploration-exploitation trade-off in complex environments, rendering it a versatile tool for various reinforcement learning applications.

5.2.2 PortfolioOptimization.py

The PortfolioOptimization.py script applies PPO to the task of optimizing financial portfolios. Its objective is to maximize expected returns for a given risk level through strategic asset allocation. The script starts by fetching and processing data for ETFs from Yahoo Finance, involving data cleaning and normalization. The environment for the PPO agent is customized, including the definition of state space (historical financial data), action space (portfolio allocations), and a reward function (based on returns). Agent training in this script spans multiple episodes, each involving decisions on asset allocations and observation of the resulting rewards. Post-training, the agent's performance is evaluated on a separate test dataset, utilizing metrics like the Sharpe Ratio to assess risk-adjusted returns. The script also conducts stationarity tests on financial time series data, ensuring the reliability of model predictions, and customizes the environment to reflect financial portfolio optimization scenarios. This approach highlights the practical utility of advanced reinforcement learning algorithms in strategic financial decision-making and asset allocation.

5.3 ETF Selection through Mixed Integer Programming

5.3.1 Overview

A crucial step in the construction of a quantitative wealth management model using DRL involves the selection of financial instruments (ETFs in this instance) represented by their tickers. Given the inherent complexity of markets and the vast number of potential assets, an intelligent selection mechanism is required to construct portfolios that vary in size – categorized as small, medium, or large. Mixed Integer Programming (MIP) offers a structured approach to handle this selection by incorporating quantitative criteria and predefined constraints that align with portfolio management objectives and regulatory compliance.

5.3.2 Methodology

The methodology employed for ticker selection is formulated as a MIP problem, where the decision variables represent the inclusion or exclusion of each potential ticker. For each portfolio size—small, medium, and large—distinct sets of constraints and objectives will be defined as follows:

- **Small Portfolio:** Constraints will focus on maximizing diversification within a limited number of assets. The MIP model will include binary variables indicating the presence of a ticker in the portfolio and will optimize for the highest expected return per unit of risk, subject to a cap on the total number of assets.
- **Medium Portfolio:** In addition to the diversification and risk-return optimization, the MIP model will better incorporate constraints related to sector exposure, ensuring a balanced representation across different market sectors and maintaining a level of liquidity that matches the medium portfolio's size.
- **Large Portfolio:** This model extends the complexity by incorporating the greatest number of ETFs. The larger number of assets allows for the exploration of additional DRL behavior.

5.3.3 Constraints and Objectives

Each MIP model will operationalize the following:

- **Integration of Integer Variables:** Discrete decisions regarding asset inclusion/exclusion, investment allocation, and timing of trades will be translated into integer variables.
- **Flexibility and Risk Management:** The MIP models will be flexible enough to incorporate various investment objectives and risk measures, tailored to the size of the portfolio, to manage the trade-off between risk and return.
- **Complex Constraints Handling:** Constraints will manage practical considerations such as concentration limits, sector exposure, and minimum participation.

5.4 Quadratic Programming Formulation for Portfolio Optimization

5.4.1 Objective

Minimize the objective function:

$$\min \lambda \left(\sum_i \sum_j w_i \cdot C_{ij} \cdot w_j \right) - (1 - \lambda) \left(\sum_i r_i \cdot w_i \right) \tag{1}$$

where λ is a trade-off parameter that balances the focus between minimizing risk and maximizing return.

5.4.2 Constraints

- $\sum_i w_i = 1$ (The sum of weights is equal to 1, representing a fully invested portfolio)
- $\sum_i r_i \cdot w_i \geq R_{\min}$ (The expected return of the portfolio is at least the minimum performance criterion)
- $w_i \geq 0$ (Weights are non-negative)
- The portfolio size is strictly N

5.4.3 Decision Variables

- X_i = Whether an ETF is selected (binary)
- w_i = Weight of ETF (continuous)

5.4.4 Parameters

- r_i = Expected Return of ETF i
- C_{ij} = Covariance Matrix between ETF i and ETF j
- N = Number of available ETFs desired in the portfolio
- $R_{\min}$ = Minimum performance criterion

6 Data

An Exploratory Data Analysis (EDA) was conducted on daily log returns of 7 ETFs that range from being a basket of equities to corporate bonds: iShares Russell 1000 Growth ETF (IWF), iShares Russell 1000 Value ETF(IWD), iShares Russell 2000 Growth ETF(IWO), iShares MSCI Emerging Markets ETF (EEM), iShares MSCI Japan ETF (EWJ), iShares 0-5 Year High Yield Corporate Bond ETF (SHYG), iShares MSCI USA Momentum Factor ETF (MTUM). The time window for these datasets spans from Jan 1st, 2001 to December 31, 2022. However, given that a few of the ETFs were incepted at a later date, they were unable to adhere to the set time window. This culminated in some of the data having lower row counts.

Before further exploration, an Augmented-Dickey Fuller (ADF) test was performed on the log returns of each ETF to confirm stationarity. The descriptive statistics, namely the mean and the standard deviation, were extracted as the expected return and risk to calculate annualized Sharpe ratio.

ETF	Expected Return	Risk	Sharpe ratio
IWD	0.000262	0.0125	0.4543
IWF	0.00272	0.0131	0.5891
IWO	0.000255	0.0159	0.2682
EEM	0.000319	0.0178	0.0355
EWJ	0.00085	0.0136	0.1692
SHYG	0.00012	0.0041	0.4438
MTUM	0.00048	0.0124	0.5791

While individual Sharpe ratios are not path-breaking, the intent of this project lies in constructing portfolios of diversified ETFs that track varying asset classes. This allows for reduced portfolio volatility, potentially causing an uptick in their respective Sharpe ratios and other risk-adjusted measures.

7 Results & Analysis

7.1 Summary

The numerical approaches presented in Section 4.2 are observed to evaluate the performance of the DRL agent. This is done to acquire a clear and concise idea of how the DRL agent performs relative to its assets and the market as a whole. Additionally, varying portfolio sizes are implemented. This is done in order to discover how the DRL agent adapts to differing markets. For instance, large portfolios have persistently shown to reap the benefits of diversification under the premise of long-term stability. On the other hand, restricting the DRL agent to smaller portfolio sizes serves to evaluate whether or not it is agile enough to make the most out of short-term opportunities. This ensures that the DRL agent learns to optimize returns for contrasting portfolio sizes (i.e. smaller portfolios requiring more risk for substantial growth). Overall, the DRL agent demonstrates scalability and flexibility, which is crucial for addressing a plethora of market conditions and/or client needs. All of the following portfolios are curated by a DRL agent that was trained and validated under the following parameters:

Parameter	Value
Episodes	100
Actor Learning Rate	0.001
Critic Learning Rate	0.001
Clip Ratio	0.2
Training Interval	10

7.2 Large Portfolio

The large portfolio is comprised of all ETFs mentioned in Section 5 aside from IWO. The DRL agent is trained from 2014 to 2019, with the validation set spanning from January 2020 to November 2023. This portfolio exhibits relatively consistent growth. Within the training set, the average Sharpe Ratio is 1.58, whereas in the validation set the average Sharpe Ratio only slightly decreases to 1.55. The inclusion of ETFs encompassing a broad range of risk-return profiles, as well as an optimization strategy, has shown to enable the DRL agent to practice diversification. This reduces idiosyncratic risk. Consequently, we note this is the least performing strategy.

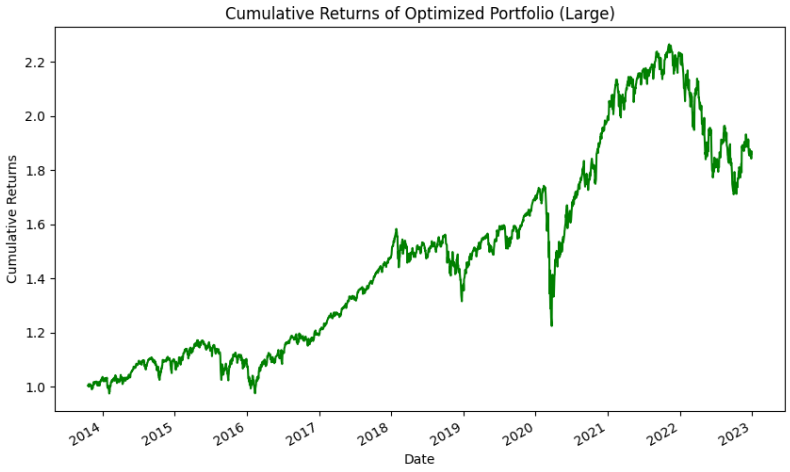

Figure 4: Cumulative Returns of Optimized Large PPO Portfolio

7.3 Medium Portfolio

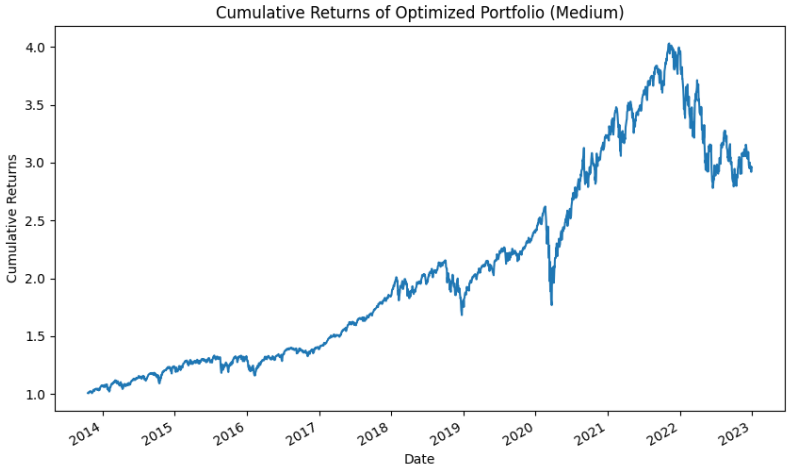

Figure 5: Cumulative Returns of Optimized Medium PPO Portfolio

The medium portfolio consists of the following ETFs: IWF, EEM, SHYG, and MTUM. Within the same training and validation period, this portfolio results in slightly more stable growth. This is visible through the portfolio's Sharpe Ratio: 1.75 in the training set and 1.73 in the validation set. It is worth noting that the DRL agent is in fact adapting to newfound market conditions, for the Sharpe Ratios holding steady suggests the DRL agent maintains a robust portfolio under a different environment. With further testing under more realistic market interactions (such as accounting for transaction fees), the DRL agent's enhanced learning under smaller portfolio sizes shines potential on being comparatively greater to larger portfolio sizes.

7.4 Small Portfolio

The small portfolio holds just two ETFs: IWD and EWJ. The DRL agent's environment and actions are by far the simplest in this instance. Nonetheless, this portfolio outperforms the aforementioned portfolios. The small portfolio boasts a Sharpe Ratio of 2.03 in the training set and 2.01 in the validation set. With the sacrifice of versatility and

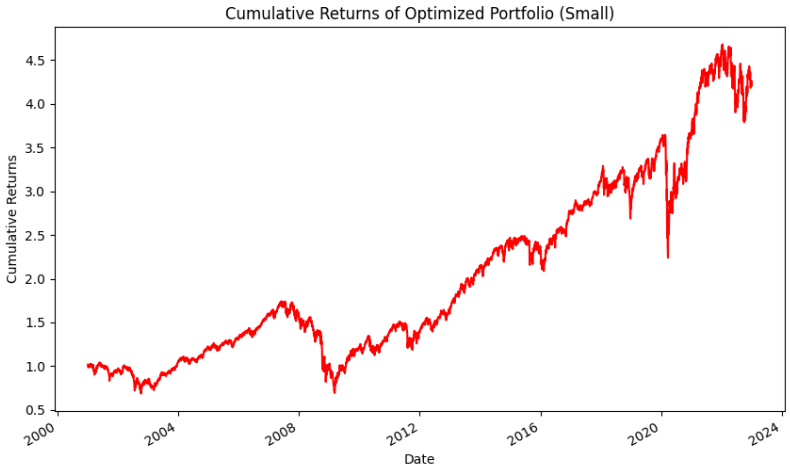

Figure 6: Cumulative Returns of Optimized Small PPO Portfolio

complexity, the DRL agent expresses a rather growth-oriented approach (as we previously hypothesized for small portfolio allocations). Despite the nature of growth being greater risk, this portfolio weathers a considerable amount of the historical volatility shocks endured by the overall market.

8 Conclusion & Future Work

This research marks a significant milestone in the integration of Deep Reinforcement Learning (DRL), specifically through the Proximal Policy Optimization (PPO) algorithm, into the realm of Quantitative Wealth & Investment Management (QWIM). Our study demonstrates that the DRL agent, trained on a dual-network structure of actor and critic models, is not only capable of adapting to the dynamic and complex nature of financial markets but also excels in optimizing investment strategies for various portfolio sizes.

The comprehensive analysis of our DRL agent, trained across a diverse spectrum of ETFs, has unveiled its adeptness in navigating the volatile terrains of the investment landscape. The performance evaluation across different portfolio sizes – large, medium, and small – indicates that the DRL agent is not only scalable and flexible but also proficient in leveraging different investment theses. The consistent Sharpe Ratios across training and validation sets for all portfolio sizes underscore the agent's ability to maintain robust performance even under varying market conditions.

Particularly noteworthy is the superior performance of the small portfolio, which, despite its simplicity and higher risk profile, achieved the highest Sharpe Ratio. This finding suggests that the DRL agent is remarkably effective in capitalizing on short-term opportunities and managing risk in growth-oriented strategies.

To further elevate the efficacy of our Deep Reinforcement Learning model in the context of Quantitative Wealth & Investment Management, our future research endeavors will encompass a multifaceted enhancement strategy. Firstly, we will expand the data set to include a broader range of financial instruments, such as bonds, commodities, and cryptocurrencies, thereby diversifying the asset classes and testing the algorithm's adaptability to a wider market spectrum.

Concurrently, we will delve into hyperparameter tuning, experimenting with various settings for learning rates, clip ratios, and training intervals. This process, potentially utilizing grid or random search methods, is aimed at fine-tuning the model's performance to its pinnacle. Additionally, our focus will shift towards algorithmic enhancements, exploring advanced variants of Proximal Policy Optimization or other reinforcement learning algorithms, particularly to augment performance in volatile market conditions. Further, we intend to sophisticate the model's input features by integrating macroeconomic indicators and sentiment analysis data, providing a more comprehensive and nuanced view of market dynamics.

Lastly, a crucial aspect of our approach will involve adapting the model to more accurately reflect real-world trading conditions. This adaptation includes simulating transaction fees to mirror the costs of trade execution,

considering market liquidity, and acknowledging the impact of significant orders on market prices, thereby ensuring that our model aligns closely with the practical realities of financial markets.

References

Bauman, Tessa, et al. "Deep Reinforcement Learning for Robust Goal-Based Wealth Management" (2023).

Benhamou, Eric, et al. "AAMDRL: Augmented Asset Management with Deep Reinforcement Learning" (2020).

Cong, Lin William, et al. "AlphaPortfolio: Direct Construction Through Deep Reinforcement Learning and Interpretable AI" (2022).

Das, Sanjiv, and Subir Varma. "Dynamic Goals-Based Wealth Management Using Reinforcement Learning" (2020).

Das, Sanjiv R., et al. "Optimal Goals-Based Investment Strategies For Switching Between Bull and Bear Markets" (2021).

Guan, Mao, and Xiao-Yang Liu. "Explainable Deep Reinforcement Learning for Portfolio Management: An Empirical Approach" (2021).

Nakagawa, Kei, Masaya Abe, and Junpei Komiyama. "RIC-NN: A Robust Transferable Deep Learning Framework for Cross-sectional Investment Strategy" (2020).

Wu, Bo, and Lingfei Li. "Reinforcement Learning for Continuous-Time Mean-Variance Portfolio Selection in a Regime-Switching Market" (2023).