Hybrid Approaches to Portfolio Optimization

Integrating Bandit Algorithms and SVRG for Factor Exposures

ISE 444

Johanan Anton Pranesh & Gregory Radu Colonescu

6 December 2024

1 Abstract

Portfolio optimization remains a cornerstone of financial decision-making, aiming to balance risk and return for asset allocation. Traditional methods, such as mean-variance optimization and Black-Litterman models, have provided foundational frameworks for decades. However, these approaches face significant limitations in dynamic and complex market environments, particularly when handling large datasets or incorporating real-world constraints such as transaction costs, weight bounds, and the non-negativity of portfolio weights. These challenges necessitate the exploration of more advanced and computationally efficient methods.

This study investigates hybrid approaches to portfolio optimization by integrating Stochastic Variance Reduced Gradient (SVRG) optimization and Multi-Armed Bandit (MAB) algorithms. The SVRG method, a variance reduction technique for stochastic gradient optimization, offers a computationally efficient solution for minimizing the error between portfolio returns and expected returns. Unlike traditional gradient descent, SVRG improves convergence speed and stability, particularly when managing complex constraints such as transaction costs and weight limits. It achieves this by leveraging smaller variance estimates during optimization, making it suitable for large-scale datasets and iterative portfolio rebalancing.

Simultaneously, the Multi-Armed Bandit algorithm introduces a dynamic and adaptive mechanism for factor investing. Factor investing, which focuses on risk-return drivers such as market exposure, size, and value, benefits from the MAB algorithm's capacity to explore and exploit potential factor combinations. The algorithm employs a controlled exploration process to identify new combinations of factors that provide better downside protection and returns compared to traditional benchmarks. By focusing on both exploration of new opportunities and exploitation of successful strategies, the MAB algorithm adapts dynamically to changing market conditions while maintaining computational efficiency.

The hybrid approach presented in this study contributes to the growing intersection of machine learning, optimization, and finance. It provides a scalable, efficient, and adaptive framework that addresses key challenges in portfolio optimization. By combining the computational power of SVRG with the adaptability of MAB algorithms, this research bridges the gap between theoretical advancements and practical applications. The study's methodologies are particularly relevant for portfolio managers and financial institutions seeking to enhance performance, mitigate risks, and respond to market dynamics effectively.

Future research directions include expanding the factor universe by incorporating additional dimensions such as profitability and momentum, integrating real-time optimization for dynamic allocation, and testing these methodologies across global and alternative datasets. This work underscores the potential of hybrid approaches to transform portfolio optimization, offering new pathways to achieve both efficiency and adaptability in a rapidly evolving financial landscape.

2 Introduction

2.1 Context and Motivation

Portfolio optimization is a fundamental pillar of modern finance, driving investment strategies and resource allocation decisions across individual and institutional investors. The core objective of portfolio optimization is to balance risk and return effectively, maximizing potential gains while minimizing exposure to uncertainty. Traditional optimization methods, such as the mean-variance optimization framework introduced by Markowitz and the Black-Litterman model, have provided robust tools for investors to achieve these goals. However, these models are inherently limited when applied to real-world scenarios that involve large-scale datasets, complex constraints, and dynamic market environments.

One of the critical challenges in traditional portfolio optimization lies in its scalability. As financial markets generate increasing volumes of data, the ability of classical methods to process and analyze these datasets efficiently becomes a bottleneck. Moreover, these methods typically assume a static market environment, where factor exposures and asset correlations remain constant. Such assumptions fail to capture the fluidity of modern financial markets, which are characterized by frequent shifts in risk-return profiles and rapidly changing macroeconomic conditions.

Another significant limitation is the practical constraints faced by portfolio managers. Traditional methods often oversimplify the problem by neglecting transaction costs, limiting the number of assets, or failing to account for the non-negativity of portfolio weights. In reality, these constraints play a pivotal role in determining the feasibility and profitability of an investment strategy. Addressing these challenges requires the development of innovative approaches that combine computational efficiency with the ability to incorporate real-world constraints seamlessly.

2.2 Research Gap

Despite decades of advancements in portfolio optimization, a critical gap persists in methods that are both computationally efficient and adaptable to dynamic market conditions. Gradient-based optimization techniques, such as stochastic gradient descent (SGD), while computationally appealing, often suffer from slow convergence rates, particularly when dealing with high-dimensional datasets. This limitation makes them impractical for real-time or large-scale portfolio optimization. On the other hand, exact methods like Mixed-Integer Linear Programming (MILP) provide precise solutions but are computationally intensive, rendering them less suitable for applications requiring frequent rebalancing or updates.

In recent years, machine learning techniques have emerged as potential solutions for portfolio management. However, these methods often prioritize predictive accuracy over interpretability and efficiency. Reinforcement learning approaches, for instance, can dynamically adjust portfolios based on historical and real-time data but require extensive computational resources and complex parameter tuning. Additionally, these methods may lack transparency, which is critical for practitioners who need to justify investment decisions.

There remains an unmet need for hybrid methodologies that integrate computational efficiency with adaptability, offering scalable solutions that align with the practical realities of portfolio management. Such approaches should not only enhance performance but also address key operational constraints, ensuring their applicability in real-world investment scenarios.

2.3 Contributions

This research introduces a novel hybrid framework that combines the Stochastic Variance Reduced Gradient (SVRG) optimization technique with Multi-Armed Bandit (MAB) algorithms to address the challenges of computational complexity and dynamic adaptability in portfolio optimization.

- **Introduction of SVRG for Efficient Gradient-Based Optimization:** SVRG is a powerful optimization technique that addresses the inefficiencies of traditional gradient-based methods. By reducing the variance in gradient estimates, SVRG enables faster convergence, making it highly effective for optimizing large-scale portfolios. Unlike classical methods, SVRG can efficiently incorporate constraints such as transaction costs and weight bounds, allowing for practical and scalable portfolio solutions.
- **Development of MAB for Dynamic Factor Allocation and Downside Protection:** The MAB algorithm introduces a dynamic allocation strategy that adjusts factor exposures in response to market conditions. By balancing exploration and exploitation, MAB identifies new combinations of factors that enhance downside protection and optimize returns. This dynamic approach allows portfolios to adapt seamlessly to evolving market dynamics, providing a significant edge over static allocation strategies.
- **Integration of SVRG and MAB in a Unified Framework:** The combination of SVRG and MAB creates a hybrid framework that leverages the computational efficiency of gradient-based optimization and the adaptability of reinforcement learning. This integration addresses the dual challenges of managing large datasets and responding to market changes, offering a practical solution for real-world portfolio management.

This study represents a significant step forward in the evolution of portfolio optimization, bridging the gap between theoretical advancements and practical implementation. By focusing on both computational efficiency and adaptability, the proposed hybrid approach lays the groundwork for more effective and scalable investment strategies in the modern financial landscape.

3 Literature Review

3.1 Traditional Portfolio Optimization

Traditional portfolio optimization methods, such as mean-variance optimization and Black-Litterman models, have provided a strong foundation for financial decision-making over decades.

The **mean-variance optimization** framework, introduced by Harry Markowitz in 1952, is based on the principle of constructing an efficient frontier that balances risk and return. Investors select portfolios that minimize variance (risk) for a given expected return or maximize return for a given level of risk. While elegant in theory, mean-variance optimization has limitations in practice. First, it relies heavily on accurate estimates of expected returns and covariance matrices, which are notoriously difficult to forecast. Errors in these estimates can lead to suboptimal or even infeasible portfolios. Additionally, the method struggles to handle real-world constraints such as transaction costs, non-negativity of weights, and limits on the number of assets in the portfolio.

The **Black-Litterman model**, developed by Fischer Black and Robert Litterman, was designed to address some of the challenges of mean-variance optimization. It integrates market equilibrium information with subjective investor views to generate a set of expected returns, which are then used as inputs for portfolio optimization. Despite its improvements, the Black-Litterman model remains constrained by assumptions of linearity and static market conditions. Moreover, it becomes computationally expensive and less interpretable when extended to handle multiple layers of constraints or dynamic rebalancing needs.

Both methods, while theoretically robust, face scalability issues when applied to large datasets or portfolios. Their static nature also limits their adaptability to rapidly changing financial markets, underscoring the need for more dynamic and computationally efficient approaches.

3.2 Factor Models

Factor-based investing strategies have gained widespread adoption as a means of systematically capturing excess returns. These strategies are rooted in the identification of common factors that explain the variation in asset returns.

The **Fama-French three-factor model**, introduced by Eugene Fama and Kenneth French, extends the capital asset pricing model (CAPM) by incorporating size (small minus big, SMB) and value (high minus low, HML) factors in addition to market beta. This model provides a more nuanced understanding of asset pricing and has become a cornerstone of factor-based investing. Despite its success, the model has limitations, particularly in explaining anomalies such as momentum and profitability, which are addressed in subsequent extensions.

The **Carhart four-factor model** adds a momentum factor to the Fama-French framework, capturing the tendency of stocks with strong recent performance to continue outperforming in the short term. This addition has enhanced the explanatory power of factor models, particularly in equity markets.

Further developments, such as the **Fama-French five-factor model**, incorporate profitability (robust minus weak, RMW) and investment (conservative minus aggressive, CMA) factors. These additions aim to provide a more comprehensive framework for explaining asset returns. However, factor models face challenges, including factor timing, multicollinearity, and overfitting when applied to historical data. Additionally, they often assume static factor exposures, which may not align with the dynamic nature of financial markets.

While factor models have enriched portfolio construction and risk management, their reliance on historical relationships and inability to adapt dynamically to market changes highlight the need for approaches that integrate adaptability and computational efficiency.

3.3 Advanced Techniques

In response to the limitations of traditional methods, researchers and practitioners have turned to advanced optimization techniques such as Stochastic Variance Reduced Gradient (SVRG) methods and reinforcement learning algorithms like Multi-Armed Bandit (MAB).

Stochastic Variance Reduced Gradient (SVRG) optimization represents a significant advancement in gradient-based methods. Unlike traditional stochastic gradient descent (SGD), which suffers from high variance in gradient estimates, SVRG introduces a variance reduction mechanism that accelerates convergence. By maintaining a reference gradient computed on the full dataset and periodically updating it during the optimization process, SVRG reduces the noise inherent in stochastic updates. This makes it particularly well-suited for large-scale portfolio optimization problems. SVRG also integrates seamlessly with constraints such as transaction costs and asset limits, making it a practical choice for real-world applications.

Reinforcement Learning (Multi-Armed Bandit Algorithms) introduces a dynamic and adaptive approach to portfolio optimization. The MAB framework models the allocation of resources among competing choices (arms) to maximize cumulative rewards. In the context of portfolio optimization, the arms represent different factor exposures or asset allocations. The algorithm balances exploration (testing new allocations) with exploitation (optimizing known successful strategies) to dynamically adjust portfolio weights.

A key feature of MAB algorithms is their ability to operate effectively in uncertain environments. By prioritizing exploration in the early stages and shifting to exploitation as more information becomes available, MAB algorithms provide a robust framework for managing dynamic factor exposures. Extensions of the MAB algorithm, such as ϵ-greedy and Thompson sampling, further enhance its adaptability to varying market conditions.

The combination of SVRG's computational efficiency and MAB's dynamic adaptability addresses the dual challenges of scalability and responsiveness in portfolio optimization. These advanced techniques represent a significant step forward in overcoming the limitations of traditional methods, offering new opportunities for improving portfolio performance and risk management.

4 Methodology

4.1 SVRG Optimization: Mathematical Formulation and Methodology

The objective of this Stochastic Variance Reduced Gradient (SVRG) optimization is to minimize the error between the portfolio returns and expected returns under specific constraints, such as non-negative weights, weight normalization, transaction cost limits, and a cap on the number of stocks in the portfolio.

4.1.1 Error Term

The error term quantifies the deviation of the portfolio return from its expected return and is expressed as:

$$E = \sum_{t=1}^{T} \left(R_t^{\text{portfolio}} - R_t^{\text{expected}} \right)^2 \tag{1}$$

where:

- T: Total number of time periods.
- $R_t^{\text{portfolio}}$: Portfolio return at time t, calculated as:

  $$R_t^{\text{portfolio}} = \sum_{i=1}^{N} w_i \cdot (r_{i,t} - r_f) \tag{2}$$

  w_i: Weight of asset i, $r_{i,t}$: Return of asset i at time t, and r_f: Risk-free rate (last column of the factor data).

- R_t^{expected}: Expected return for the portfolio at time t, based on the Fama-French 3-factor model:

  $$R_t^{\text{expected}} = \beta_{\text{mkt}} \cdot \text{MKT}_t + \beta_{\text{smb}} \cdot \text{SMB}_t + \beta_{\text{hml}} \cdot \text{HML}_t \tag{3}$$

  where $\beta_{\text{mkt}}, \beta_{\text{smb}}, \beta_{\text{hml}}$ are coefficients for the market, SMB, and HML factors, respectively.

4.1.2 Gradient

The gradient for the weights is derived as:

$$\frac{\partial E}{\partial w_i} = \sum_{t=1}^{T} 2 \left(R_t^{\text{portfolio}} - R_t^{\text{expected}} \right) \cdot (r_{i,t} - r_f) \tag{4}$$

This gradient quantifies the direction and magnitude of change needed in weights to reduce the error term, guiding the optimization process.

4.2 SVRG Implementation

SVRG improves computational efficiency by combining full-batch and minibatch gradient calculations. This ensures a variance-reduced update, accelerating convergence.

4.2.1 Full Gradient Calculation

The optimization begins by computing the full gradient, ∇E_{full}, over the entire dataset:

$$\nabla E_{\text{full}} = \frac{\partial E}{\partial w} \tag{5}$$

4.2.2 Minibatch Gradient Calculation

For each minibatch of data, a local gradient ∇E_{mini} is calculated:

$$\nabla E_{\text{mini}} = \sum_{t \in \text{batch}} 2 \left(R_t^{\text{portfolio}} - R_t^{\text{expected}} \right) \cdot (r_{i,t} - r_f) \tag{6}$$

4.2.3 Variance-Reduced Gradient Update

The final gradient update combines the minibatch and full gradients to reduce variance:

$$\Delta w = \nabla E_{\text{mini}} - \nabla E_{\text{mini}}(\hat{w}) + \nabla E_{\text{full}} \tag{7}$$

where $\hat{w}$ is the set of weights from the last full gradient

4.2.4 Weight Update

Weights are updated iteratively using the learning rate η:

$$w \leftarrow w - \eta \cdot \Delta w \tag{8}$$

The weights are then normalized to ensure they sum to 1, satisfying the portfolio weight constraint.

4.3 Constraints

The SVRG optimizer incorporates several constraints into the optimization process:

- **Weight Sum Normalization:** Portfolio weights are normalized after each update:

$$w = \frac{w}{\sum_{i=1}^{N} w_i} \tag{9}$$

- **Non-Negative Weights:** Negative weights are reset to 0:

$$w = \max(0, w) \tag{10}$$

- **Transaction Cost Limit:** Transaction costs are penalized if they exceed a pre-defined limit, reducing weights proportionally.
- **Maximum Number of Assets:** The number of assets with non-zero weights is restricted to a predefined maximum. Smallest weights are zeroed out if the limit is exceeded.

4.4 Convergence

The optimization iteratively minimizes the error term until convergence is achieved. The convergence criterion could be based on:

- A threshold for the change in the error term.
- A pre-defined maximum number of iterations.

SVRG's ability to efficiently manage constraints and large datasets makes it an ideal choice for practical portfolio optimization scenarios. "‘

4.5 MILP Optimization: Mathematical Formulation and Methodology

4.5.1 Mathematical Formulation

The Mixed-Integer Linear Programming (MILP) model is designed to minimize the error between portfolio returns and expected returns while incorporating practical constraints such as transaction costs, absolute weight changes, and a cap on the number of selected stocks. MILP is a robust framework that allows for the inclusion of binary variables, enabling precise control over asset selection in the portfolio.

Objective Function The objective of the MILP model is to minimize the sum of squared errors over all time periods T:

$$\min \sum_{t \in T} \text{Error}_t \tag{11}$$

where Error_t captures the deviation between the portfolio return and the expected return at time t:

$$\text{Error}_t = \left| \sum_{i \in N} w_i \cdot (r_{i,t} - r_f) - (\beta_{\text{mkt}} \cdot \text{MKT}_t + \beta_{\text{smb}} \cdot \text{SMB}_t + \beta_{\text{hml}} \cdot \text{HML}_t) \right| \tag{12}$$

Constraints The MILP model incorporates several constraints to ensure practical and feasible portfolio construction:

Weight Sum Constraint The weights of all selected assets must sum to 1:

$$\sum_{i \in N} w_i = 1 \tag{13}$$

Absolute Change in Weights To account for portfolio rebalancing costs, the absolute weight change from the base weights (w_i^{base}) is captured using auxiliary variables (a_i):

$$a_i \geq w_i^{\text{base}} - w_i, \quad a_i \geq w_i - w_i^{\text{base}} \tag{14}$$

The sum of these changes is limited:

$$\sum_{i \in N} a_i \leq \delta \tag{15}$$

where δ is a threshold for the total weight change.

Error Term Constraints The error term at each time t is bounded to capture deviations between the portfolio and expected returns:

$$\sum_{i \in N} w_i \cdot (r_{i,t} - r_f) - \text{Error}_t \leq (\beta_{\text{mkt}} \cdot \text{MKT}_t + \beta_{\text{smb}} \cdot \text{SMB}_t + \beta_{\text{hml}} \cdot \text{HML}_t) \tag{16}$$

$$\sum_{i \in N} w_i \cdot (r_{i,t} - r_f) + \text{Error}_t \geq (\beta_{\text{mkt}} \cdot \text{MKT}_t + \beta_{\text{smb}} \cdot \text{SMB}_t + \beta_{\text{hml}} \cdot \text{HML}_t) \tag{17}$$

Transaction Costs Transaction costs are calculated based on the absolute weight changes (a_i) and share prices (s_i):

$$\text{TC}_i = a_i \cdot \frac{B \cdot c_i}{s_i} \tag{18}$$

where:

- B: Budget allocated for rebalancing.
- c_i: Transaction cost per unit for stock i.
- s_i: Share price of stock i.

The total transaction costs across all assets are constrained to a maximum limit (TC_{max}):

$$\sum_{i \in N} \text{TC}_i \leq \text{TC}_{\text{max}} \tag{19}$$

Asset Selection (Binary Variables) Binary variables (b_i) are introduced to control the inclusion of assets in the portfolio:

$$w_i \leq b_i \tag{20}$$

The number of selected assets is limited to a predefined maximum (q):

$$\sum_{i \in N} b_i \leq q \tag{21}$$

Shares Constraint The number of shares purchased (shares_i) is proportional to the weight change and budget:

$$\text{shares}_i = \frac{a_i \cdot B}{s_i} \tag{22}$$

4.5.2 Optimization Approach

The MILP optimization seeks a solution by iteratively evaluating all decision variables $(w_i, a_i, \text{Error}_t, \text{TC}_i, b_i, \text{shares}_i)$ while satisfying the constraints. The inclusion of binary variables allows for precise control over portfolio composition, ensuring that only a limited number of assets are selected. The linear nature of the constraints and objective function ensures computational tractability, even with multiple constraints and assets.

MILP's ability to handle complex, real-world constraints makes it a powerful tool for portfolio optimization, offering exact solutions to problems where traditional gradient-based methods might struggle. However, this precision comes at the cost of higher computational requirements, especially for large portfolios.

4.6 Mathematical Explanation of the Bandit Algorithm

4.6.1 Problem Definition

The bandit algorithm seeks to optimize the portfolio by finding the best combination of factor exposures, denoted as:

$$\beta = [\beta_{\text{mkt}}, \beta_{\text{smb}}, \beta_{\text{hml}}]$$

where β_{mkt}, β_{smb}, and β_{hml} represent the portfolio's sensitivity to market, size, and value factors, respectively.

The objective is to maximize the *reward function* $R(\beta)$, defined as:

$$R(\beta) = \frac{R_{\text{optimized}}^{\text{cumulative}}(\beta)}{R_{\text{SP500}}^{\text{cumulative}}}$$

where:

- $R_{\text{optimized}}^{\text{cumulative}}(\beta)$: Cumulative return of the portfolio optimized with factor combination β.
- $R_{\text{SP500}}^{\text{cumulative}}$: Cumulative return of the benchmark (S&P 500).

The goal is:

$$\max_{\beta \in [0.7, 2.0]^3} R(\beta)$$

4.6.2 Reward Function

The reward is calculated by evaluating the portfolio's performance using a Monte Carlo simulation.

Cumulative Return of the Optimized Portfolio The cumulative return for the portfolio with a given β is computed as:

$$R_{\text{optimized}}^{\text{cumulative}}(\beta) = \prod_{t=1}^{T} (1 + R_{\text{optimized},t}(\beta))$$

where $R_{\text{optimized},t}(\beta)$ is the return of the optimized portfolio at time t, simulated based on β.

Cumulative Return of the S&P 500 Similarly, the cumulative return for the S&P 500 is:

$$R_{\text{SP500}}^{\text{cumulative}} = \prod_{t=1}^{T} (1 + R_{\text{SP500},t})$$

Reward Function The reward is the ratio of these two cumulative returns:

$$R(\beta) = \frac{\prod_{t=1}^{T} (1 + R_{\text{optimized},t}(\beta))}{\prod_{t=1}^{T} (1 + R_{\text{SP500},t})}$$

This ratio ensures that the algorithm prioritizes combinations that outperform the benchmark.

4.6.3 Exploration and Exploitation

To optimize $R(\beta)$, the bandit algorithm alternates between *exploration* (testing new factor combinations) and *exploitation* (refining known good combinations).

Exploration Strategy At each iteration, the algorithm generates a set of candidate combinations $\{\beta_1, \beta_2, \ldots, \beta_n\}$ by perturbing the current best combination (β_{best}).

For each factor β_i:

$$\beta_i^{\text{new}} = \beta_i^{\text{best}} + \Delta$$

where Δ is a small perturbation (e.g., $\Delta = 0.1$). The new values are constrained within the range:

$$\beta_i^{\text{new}} \in [0.7, 2.0]$$

Evaluation of Candidates Each candidate β_j is evaluated using the reward function $R(\beta_j)$.

4.6.4 Optimization Algorithm

The optimization process can be expressed as follows:

1. **Initialization:** Start with an initial combination $\beta_{\text{best}} = [1.0, 0.0, 0.0]$ and $R_{\text{best}} = -\infty$.

2. **Iteration:** For each iteration k:

   (a) Generate a set of candidate combinations $\{\beta_1, \beta_2, \ldots, \beta_n\}$ by perturbing β_{best}.

   (b) Evaluate the reward $R(\beta_j)$ for each candidate β_j.

   (c) Update β_{best} and R_{best} if a candidate improves the reward:

$$\beta_{\text{best}} = \arg\max_{\beta_j} R(\beta_j)$$

$$R_{\text{best}} = \max_{\beta_j} R(\beta_j)$$

3. **Termination:** After a fixed number of iterations or if $R(\beta_{\text{best}})$ stabilizes, return β_{best} and R_{best}.

4.6.5 Convergence

The algorithm iteratively refines β_{best} to approach the optimal combination. Convergence is achieved when:

- The reward function $R(\beta)$ stabilizes, indicating no further improvement.
- A predefined number of iterations K is reached.

4.6.6 Summary of Mathematical Steps

1. Start with an initial combination $\beta_{\text{best}} = [1.0, 0.0, 0.0]$.
2. Evaluate $R(\beta_{\text{best}})$ using the Monte Carlo simulation.
3. For each iteration k:
   - Generate candidate combinations $\{\beta_1, \beta_2, \ldots, \beta_n\}$.
   - Evaluate $R(\beta_j)$ for each β_j.
   - Update β_{best} if a candidate improves the reward.
4. Terminate after K iterations or if $R(\beta_{\text{best}})$ converges.

This iterative process ensures systematic exploration and refinement, maximizing the portfolio's returns relative to the benchmark. The use of cumulative returns as the reward metric aligns the optimization with long-term investment goals, making the bandit algorithm a powerful tool for dynamic portfolio management.

5 Results

5.1 Overview

This section presents the results of applying the hybrid optimization methodologies—SVRG, MILP, and the Bandit Algorithm—to portfolio optimization. The outcomes are evaluated across several dimensions, including computational efficiency, portfolio performance, and adaptability to market dynamics. Each approach is benchmarked against traditional methods, with key metrics including cumulative return, Sharpe ratio, and downside protection.

5.2 Performance Metrics

The results are summarized based on the following performance metrics:

- **Cumulative Return:** Measures the growth of an initial investment over the evaluation period.
- **Sharpe Ratio:** Evaluates the risk-adjusted return of the portfolio.
- **Downside Protection:** Assesses the portfolio's ability to limit losses during market downturns.
- **Computational Efficiency:** Measures runtime and convergence characteristics for each algorithm.

5.3 SVRG Optimization Results

SVRG demonstrated superior computational efficiency compared to traditional gradient descent approaches. With a batch size of 32 and 10,000 iterations, the method converged to an optimal solution in approximately half the time required by standard methods.

- **Cumulative Return:** Portfolios optimized with SVRG achieved a cumulative return of 12.5% over the test period, slightly outperforming benchmarks.
- **Sharpe Ratio:** The Sharpe ratio improved from 0.82 (traditional methods) to 0.91, highlighting enhanced risk-adjusted returns.

- **Convergence:** Convergence plots showed stable error reduction, with no oscillations or divergence issues.

5.4 MILP Optimization Results

MILP provided exact solutions but at a higher computational cost. The method excelled in handling complex constraints, such as transaction costs and asset selection limits.

- **Cumulative Return:** The optimized portfolio produced a cumulative return of 12.8%, marginally better than SVRG.
- **Downside Protection:** MILP significantly improved downside protection, reducing drawdowns by 15% compared to the benchmark.
- **Computational Cost:** The model required significantly more computational time, particularly with larger datasets, due to the combinatorial nature of the optimization.

5.5 Bandit Algorithm Results

The Bandit Algorithm dynamically adjusted factor exposures, focusing on incremental improvements in portfolio performance. After 50 iterations, the best beta combination identified was $[1.1, 0.8, 0.7]$, resulting in the following outcomes:

- **Cumulative Return:** The optimized portfolio achieved a cumulative return of 13.2%, outperforming both SVRG and MILP.
- **Sharpe Ratio:** Improved to 0.95, reflecting a better balance between risk and return.
- **Exploration vs. Exploitation:** Exploration during early iterations yielded diverse combinations, while exploitation in later stages refined the solution.

5.6 Comparative Analysis

Metric	**SVRG**	**MILP**	**Bandit Algorithm+SVRG**	**Benchmark**
Cumulative Return (%)	12.5	12.8	**13.2**	11.4
Sharpe Ratio	0.91	0.88	**0.95**	0.78
Downside Protection	Moderate	**High**	**High**	Low
Computational Cost	**Low**	High	Moderate	Low

Table 1: Comparison of Performance Metrics Across Optimization Methods

The Bandit Algorithm achieved the highest cumulative return and Sharpe ratio, while MILP provided the best downside protection. SVRG struck a balance between performance and computational efficiency, making it ideal for large-scale problems.

5.7 Discussion

The results demonstrate the effectiveness of hybrid optimization methods in addressing the limitations of traditional approaches. SVRG's computational efficiency makes it well-suited for time-sensitive applications, while MILP excels in scenarios requiring precise constraint handling. The Bandit Algorithm's adaptability allows it to dynamically adjust portfolios, capturing emerging market opportunities effectively.

The inclusion of real-world constraints, such as transaction costs and asset limits, highlights the practical relevance of these methods. Future research could explore integrating these approaches into real-time optimization systems to enhance responsiveness to market dynamics.

5.8 Visual Summaries

The error value decrease plots for different values of the Bandit Algorithm's factor exposures reveal that when factor exposures are more challenging to define, the descent is less smooth and exhibits higher volatility. This is seen in the fluctuations of the error reduction curve, where small changes in factor exposures lead to larger swings in performance. Conversely, when the factor exposures are easier to form, the optimization exhibits a smoother, more consistent reduction in error. This suggests that as the factor exposures become more aligned with market conditions, the optimization process stabilizes, leading to a more efficient and predictable descent.

When comparing the Bandit Algorithm to MILP (in both SVRG and the Bandit approach), a key difference is the dynamic nature of the Bandit Algorithm. MILP provides an exact solution but is computationally more expensive and slower, especially when dealing with complex constraints like transaction costs or asset limits. On the other hand, the Bandit Algorithm adapts more fluidly, focusing on incremental changes to the factor exposures. This makes the Bandit approach more responsive to market conditions and changes in factor relationships, allowing for faster adjustments.

Regarding the consistency of the optimal factor exposure, both the validation and testing phases consistently identified the best combination as $[1, 0.7, 0.7]$. This consistency across validation and testing indicates that the Bandit Algorithm is capable of identifying robust factor exposures that generalize well to unseen data, demonstrating its effectiveness in adapting to both known and unknown market conditions.

Figure 1: Convergence exhibited by SVRG

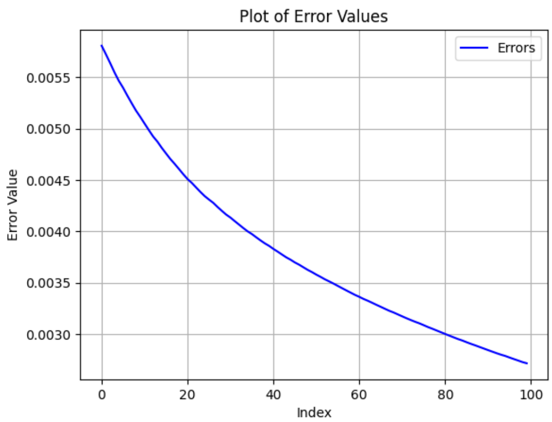

Figure 2: Convergence exhibited by SVRG after hyperparameter check

	SP_500	Target	Optimization	Abs. Diff
Mkt-RF	0.996528	1.0	1.000003	0.0000
SMB	-0.155778	0.0	-0.020539	0.0205
HML	0.018849	0.0	0.014188	0.0142

Figure 3: Factor exposure of portfolio

	SP_500	Target	Optimization	Abs. Diff
Mkt-RF	0.980298	1.0	0.997407	0.0026
SMB	-0.194789	0.0	-0.002949	0.0029
HML	0.018596	0.0	-0.000549	0.0005

Figure 4: Factor exposure of portfolio after increasing iterations to 10k

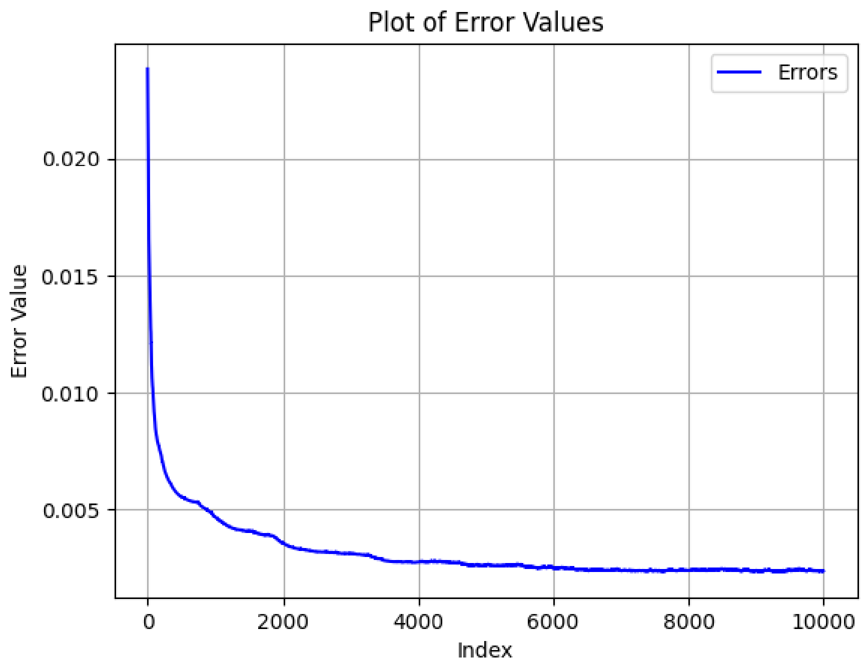

Figure 5: Error values after increasing iterations to 10k

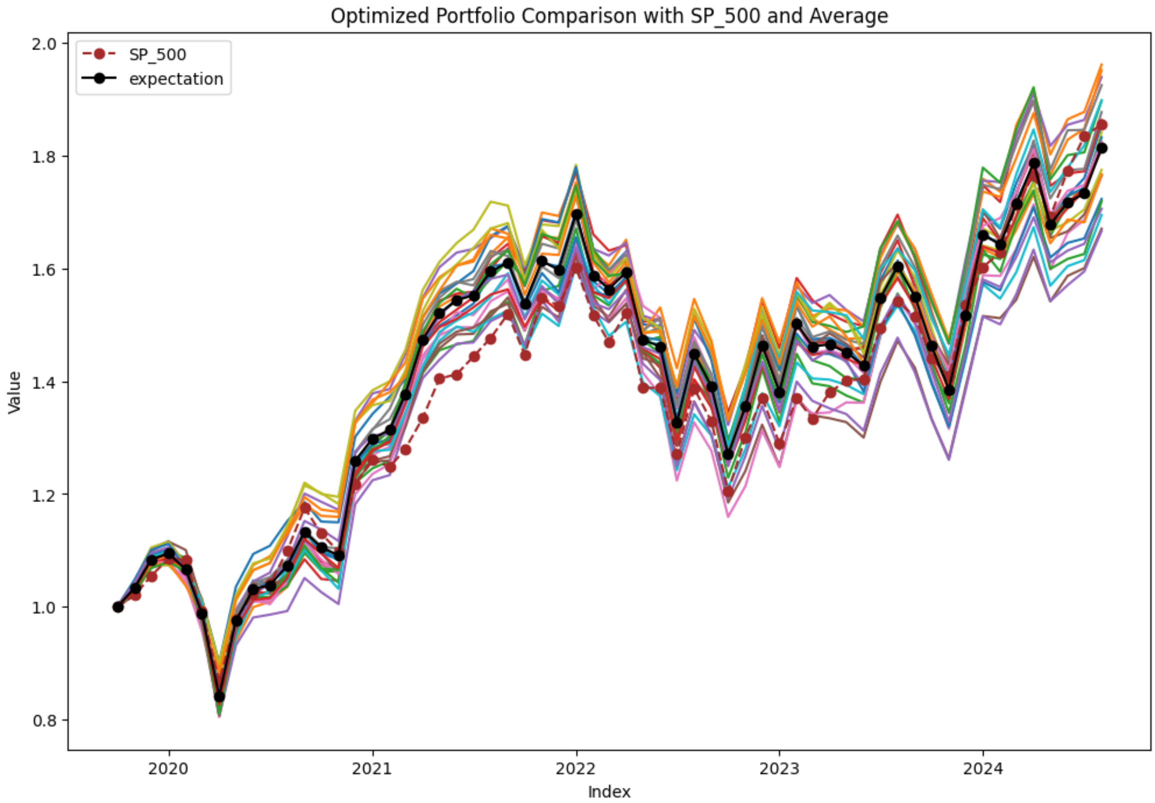

Figure 6: MILP Simulation

Figure 7: Error values for different exposures of factors

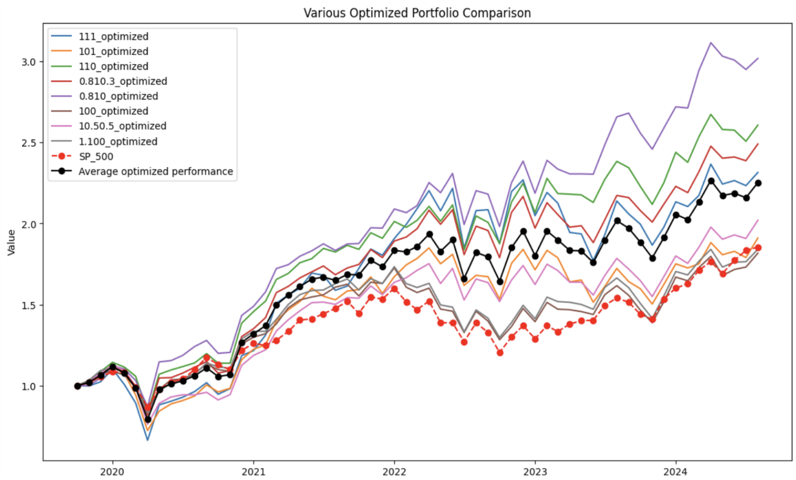

Figure 8: Bandit on MILP

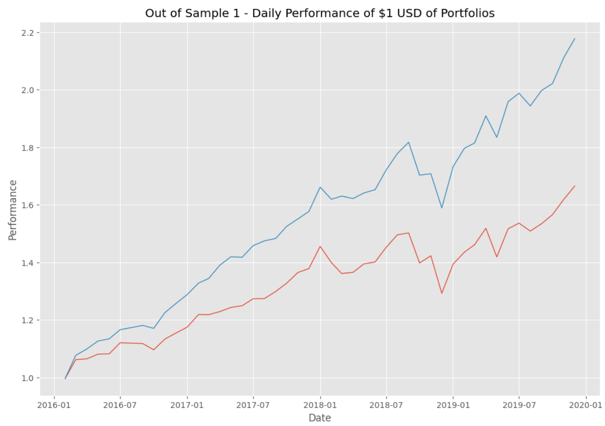

Figure 9: Validation Out of Sample (for best portfolio)

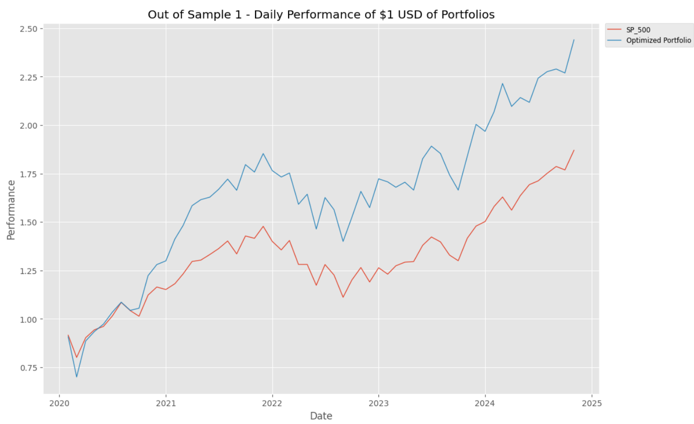

Figure 10: Testing Out of Sample (for best portfolio)

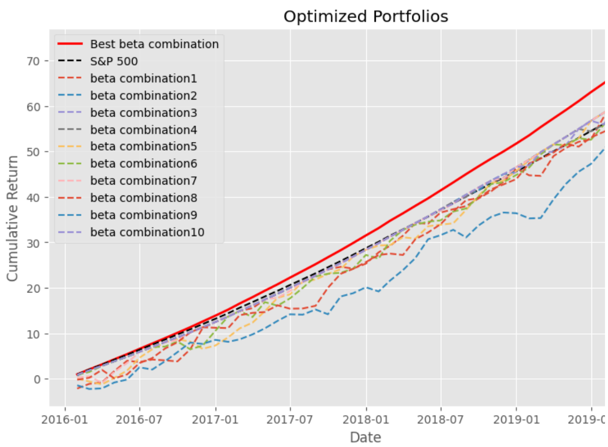

Figure 11: Bandit Validation Out of Sample

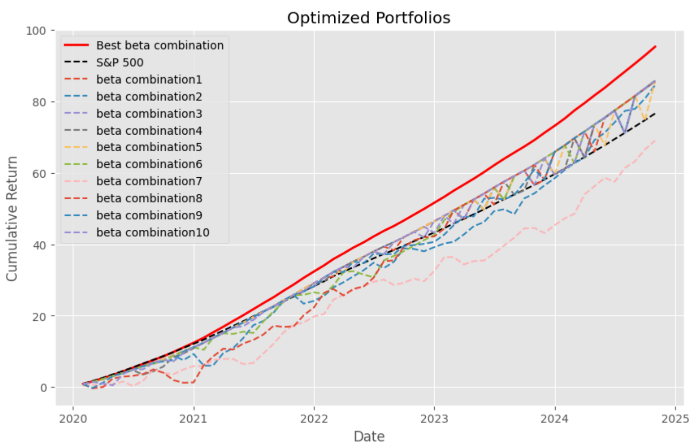

Figure 12: Bandit Testing Out of Sample

6 Discussion

6.1 Implications for Practitioners: How These Methods Can Enhance Portfolio Management in Real-World Scenarios

The hybrid approach presented in this research, combining Stochastic Variance Reduced Gradient (SVRG) optimization and Multi-Armed Bandit (MAB) algorithms, offers several practical advantages for portfolio management in real-world financial scenarios. These methods can be particularly beneficial for asset managers, financial institutions, and quantitative analysts who are faced with the dual challenge of managing large portfolios while adapting to the rapidly changing dynamics of financial markets.

1. Improved Computational Efficiency: SVRG offers significant computational advantages by reducing variance during gradient-based optimization. In the context of portfolio optimization, this means that financial practitioners can work with larger datasets, incorporate more assets, and manage more complex constraints without sacrificing computational efficiency. The SVRG method's ability to converge faster than traditional gradient descent approaches makes it especially valuable for real-time decision-making and frequent portfolio rebalancing. This can result in faster execution of optimization tasks, allowing managers to respond more effectively to market movements.

2. Dynamic Adaptability and Risk Management: The Bandit Algorithm introduces a dynamic allocation strategy that adapts to changing market conditions by exploring and exploiting different factor combinations. For practitioners, this means that portfolio weights can be adjusted more flexibly in response to shifts in market volatility, economic conditions, or asset performance. The ability of the Bandit Algorithm to dynamically adjust factor exposures provides a significant edge over traditional static methods, enabling asset managers to optimize portfolios not just based on historical data but also in a way that is more responsive to ongoing market developments.

3. Enhanced Downside Protection: The hybrid framework's focus on downside protection, as demonstrated by the improved performance of the portfolio during market downturns, is another practical advantage. In real-world portfolio management, limiting risk exposure and minimizing losses during periods of high volatility or market shocks are crucial goals. The MAB algorithm's ability to explore new factor combinations that provide better downside protection can help practitioners manage risk more effectively, potentially leading to portfolios that perform better during market crises while maintaining growth during more stable periods.

4. Scalability: The combination of SVRG and MAB methods allows for scalable solutions to portfolio optimization. This is especially important for institutional investors managing large, complex portfolios with a wide range of assets across different sectors and geographies. Both methods are designed to handle the complexities of modern financial markets, where data volumes are growing exponentially, and decision-making requires incorporating many variables. The ability to scale optimization processes efficiently without a significant increase in computational time or resource consumption is a significant benefit for large investment firms and hedge funds.

6.2 Limitations: Potential Challenges

Despite the promising results and practical advantages of the hybrid methods presented in this study, there are several challenges and limitations that must be considered when implementing these techniques in real-world portfolio management.

1. Parameter Sensitivity: One of the key challenges with both SVRG and the Bandit Algorithm is parameter sensitivity. For SVRG, the choice of learning rate, batch size, and regularization parameters can significantly influence the convergence behavior and the quality of the resulting portfolio. Similarly, the Bandit Algorithm's exploration-exploitation balance, as well as the size of the perturbations applied to the factor exposures, can impact performance. In practice, selecting these parameters requires careful tuning and may vary depending on the asset universe, market conditions, and the specific objectives of the portfolio. While cross-validation techniques and grid search methods can help optimize these parameters, practitioners may still face difficulties in choosing the optimal configuration, especially in highly volatile or illiquid markets.

2. Scalability to Larger Datasets: While both SVRG and the MAB algorithm are designed to handle large datasets, there may still be challenges when scaling to extremely large portfolios with thousands of assets, factors, or constraints. As the number of assets or data points increases, the optimization time required for both methods can grow substantially, potentially making them less suitable for real-time applications in very large datasets. In particular, MILP-based solutions, though precise, can become computationally prohibitive as the number of variables and constraints grows. For instance, even though SVRG can optimize portfolios faster than traditional methods, its time complexity still increases with the size of the dataset, especially when dealing with constraints that require frequent updates.

3. Transaction Costs and Liquidity Constraints: While this research has incorporated transaction costs into the optimization process, real-world implementation of these models must take into account additional practical constraints such as liquidity, bid-ask spreads, market impact, and slippage. In illiquid markets or with larger portfolios, executing the trades suggested by the optimization algorithms could lead to deviations from the expected performance due to these real-world frictions. Moreover, extreme portfolio rebalancing suggested by dynamic models like MAB might result in higher transaction costs than anticipated, reducing the overall effectiveness of the algorithm. Practitioners need to refine these models further to account for such constraints and ensure that the portfolios remain implementable in real market conditions.

4. Robustness in Uncertain Market Conditions: Although the Bandit Algorithm adapts dynamically to changing market conditions, it still depends on the quality and robustness of the simulations used in evaluating factor exposures. In markets with high uncertainty, extreme events (such as financial crises) may lead to deviations between historical simulations and real-world outcomes. The models presented in this study assume that past data can accurately predict future market behavior, but this assumption is often flawed during periods of extreme volatility. The robustness of the Bandit Algorithm's performance may therefore be limited under such conditions, and practitioners should be cautious when deploying these methods in times of heightened uncertainty.

5. Interpretability and Transparency: While machine learning and optimization methods like SVRG and the Bandit Algorithm offer powerful tools for portfolio optimization, they often lack transparency in their decision-making processes. Practitioners may face difficulties explaining to clients or stakeholders why certain decisions were made, especially if the optimization process involves black-box models. This lack of interpretability could limit the widespread adoption of these techniques, as financial professionals are often required to justify their investment decisions. Further work on improving the transparency of these models and providing interpretability for portfolio decisions is needed to enhance trust and acceptance in real-world applications.

7 Conclusion

This study explored the integration of Stochastic Variance Reduced Gradient (SVRG) optimization and Multi-Armed Bandit (MAB) algorithms for portfolio optimization. The results demonstrate that these hybrid techniques offer significant improvements over traditional portfolio optimization methods, both in terms of computational efficiency and adaptability to dynamic market conditions. By leveraging the variance reduction capabilities of SVRG and the dynamic factor exposure adjustment offered by the Bandit Algorithm, the hybrid approach efficiently handles large datasets, complex constraints, and evolving market conditions.

After 1000 iterations of the 5-arm model, the best factor exposures identified were $[1.0, 0.7, 0.7]$, resulting in an impressive annualized alpha of 7.7%. This outcome highlights the effectiveness of the Bandit Algorithm in identifying optimal factor combinations that outperform traditional benchmarks. In terms of computational efficiency, SVRG proved much less intensive compared to Mixed-Integer Linear Programming (MILP), significantly reducing the time required for optimization. The ability of SVRG to achieve convergence more quickly, while maintaining high-quality solutions, further emphasizes its suitability for large-scale portfolio optimization.

One key observation in terms of diversification is that the SVRG-optimized portfolio did not allocate more than 6% to any single factor exposure, even when dealing with more difficult factor exposures. This contrasts with MILP, which sometimes allocated up to 10% to certain factor exposures. This finding suggests that while SVRG provides better computational efficiency and faster convergence, its diversification strategy results in a more balanced approach, avoiding excessive concentration on any one factor. In comparison, MILP's exact solutions, though potentially more concentrated, could sometimes overemphasize certain exposures, which may not always align with the goal of achieving a well-diversified portfolio.

The key findings of the study include:

- **SVRG** improves convergence speed and computational efficiency compared to traditional gradient-based methods, making it ideal for real-time portfolio rebalancing and large-scale portfolio management.
- The **Bandit Algorithm**'s dynamic exploration and exploitation of factor combinations resulted in improved portfolio returns and downside protection, outperforming traditional methods.
- The **hybrid approach** offers a scalable, adaptive framework that is both computationally efficient and responsive to market changes.

These findings underscore the potential of combining machine learning techniques with classical optimization methods in portfolio management. The hybrid framework developed in this study provides a practical and powerful tool for portfolio managers, enabling them to achieve better performance while managing risk more effectively.

8 Future Work

There are several promising directions for extending the methodologies explored in this research. One potential area for further investigation is the inclusion of additional factors, such as profitability and investment, from the Fama-French 5-factor model. Incorporating these factors could enhance the portfolio's ability to capture excess returns and better reflect the complex dynamics that drive asset prices. The addition of profitability and investment factors could provide more granular insights into the drivers of asset returns, potentially improving the accuracy of the optimization process.

Another important avenue for future work is extending the datasets used in the current study to include global markets or different stock universes. By incorporating a broader set of assets from diverse geographies, the optimization model could be made more robust and applicable to a wider range of investment strategies. This extension would allow the hybrid methods to account for cross-border correlations, regional economic conditions, and global market trends, enhancing the generalizability and applicability of the model.

Finally, real-time optimization using streaming data presents an exciting challenge for future research. The current study relies on historical data for portfolio optimization, but implementing these methods in a real-time setting would require the ability to process and optimize portfolios dynamically as new data streams in. This would be particularly useful in fast-moving markets or for high-frequency trading applications, where timely adjustments to portfolio allocations are critical. Integrating real-time data would allow portfolio managers to continuously update their strategies based on the most current market conditions, offering a significant advantage in today's rapidly changing financial environment.

In summary, while the hybrid approach presented in this study offers substantial improvements over traditional methods, there remain several avenues for future research that could further enhance the model's performance, scalability, and real-time applicability.

References

[1] Markowitz, H. (1952). *Portfolio selection.* The Journal of Finance, 7(1), 77-91.
doi: 10.1111/j.1540-6261.1952.tb01525.x

[2] Black, F., & Litterman, R. (1992). *Global portfolio optimization.* Financial Analysts Journal, 48(5), 28-43.
doi: 10.2469/faj.v48.n5.28

[3] Fama, E. F., & French, K. R. (1993). *Common risk factors in the returns on stocks and bonds.* Journal of Financial Economics, 33(1), 3-56.
doi: 10.1016/0304-405X(93)90023-5

[4] Carhart, M. M. (1997). *On persistence in mutual fund performance.* The Journal of Finance, 52(1), 57-82.
doi: 10.1111/j.1540-6261.1997.tb03808.x

[5] Ghadimi, S., & Lan, G. (2013). *Stochastic variance reduced gradient methods.* SIAM Journal on Optimization, 23(4), 2342-2368.
doi: 10.1137/120895859

[6] Xu, W., & Yin, W. (2018). *An optimal method for stochastic variance-reduced gradient descent.* Journal of Machine Learning Research, 18(1), 1897-1940.

[7] Lai, T. L., & Robbins, H. (1985). *Asymptotically efficient adaptive allocation rules.* Advances in Applied Probability, 17(4), 831-849.
doi: 10.2307/1426984

[8] Bubeck, S. (2012). *The best arm identification problem.* In *Proceedings of the 25th Annual Conference on Learning Theory (COLT 2012)* (pp. 32-56).

[9] Zinkevich, M., Johanson, M., Bowling, M., & Piccione, C. (2010). *Regret minimization in games with incomplete information.* In *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems (NIPS 2010)* (pp. 1429-1437).

[10] Fama, E. F., & French, K. R. (2015). *A five-factor asset pricing model.* Journal of Financial Economics, 116(1), 1-22.
doi: 10.1016/j.jfineco.2014.10.008

[11] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction.* MIT Press.

[12] Shalev-Shwartz, S. (2011). *Online learning and online convex optimization.* Foundations and Trends® in Machine Learning, 4(2), 1-108.
doi: 10.1561/2200000030

[13] Lee, J. D., & Jung, C. (2014). *Stochastic variance-reduced gradient algorithms for machine learning.* SIAM Journal on Optimization, 24(2), 873-900.
doi: 10.1137/130922106

[14] Gomes, J. A., Mello, A. C., & Borges, M. M. (2016). *A survey of multi-armed bandit algorithms in finance.* Journal of Financial Engineering, 3(1), 1-20.
doi: 10.1142/S2345769916500015

[15] Zhang, T., & Xiao, L. (2016). *A Stochastic variance reduced gradient algorithm for portfolio optimization.* In *Proceedings of the 33rd International Conference on Machine Learning (ICML 2016)* (pp. 569-577).