

一类加速ADMM算法在投资组合选择中的应用研究

胡同

河北工业大学理学院, 天津

收稿日期: 2024年2月27日; 录用日期: 2024年3月21日; 发布日期: 2024年3月29日

摘要

Mean-Variance模型为现代投资组合选取奠定了基础。近年来, 随着金融资产数量的提高, 求解M-V模型的经典算法效率逐渐变低。因此, 有关学者提出了资产分割的ADMM算法(AS-ADMM)以提升ADMM算法的效率。该算法能够比经典的算法更高效, 但在一些超高维的情况下, AS-ADMM也不足以显著提高求解效率。为了解决这个问题, 本文应用外推思想, 提出了部分加速的资产分割算法(PA-AS-ADMM), 并证明了该算法的非遍历收敛速率为 $O(\frac{1}{K})$, 最后在数值实验中验证了PA-AS-ADMM的有效性。

关键词

加速ADMM, 非遍历收敛速率, M-V模型

Research on the Application of a Class of Accelerated ADMM Algorithms in Portfolio Selection

Tong Hu

College of Sciences, Hebei University of Technology, Tianjin

Received: Feb. 27th, 2024; accepted: Mar. 21st, 2024; published: Mar. 29th, 2024

Abstract

The Mean-Variance model laid the foundation for modern portfolio selection. In recent years, as the number of financial assets has increased, the efficiency of classic algorithms for solving the M-V model has gradually decreased. Therefore, scholars have proposed Asset-Splitting ADMM algorithm (AS-ADMM) to improve the efficiency of ADMM algorithm. This algorithm can be more efficient than the classic ones, but in high-dimensional cases, AS-ADMM is not sufficiently effective. To address this problem, this paper applies the extrapolation idea and proposes Partially Accelerated Asset Segmentation algorithm (PA-AS-ADMM), and proves that the non-ergodic convergence rate of this algorithm is $O(\frac{1}{K})$. Finally, the effectiveness of PA-AS-ADMM is verified in numerical experiments.

Keywords

Accelerated ADMM, Non-Ergodic Convergence Rate, M-V Model

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

M-V模型是由美国学者马科维茨 (Markowitz) [1]在1952年建立的, 模型的具体形式为:

$$\begin{aligned} \min_{\omega} \quad & \omega^T \Sigma \omega \\ \text{s.t.} \quad & \mu^T \omega \geq \mu_0. \end{aligned} \quad (1.1)$$

其中, ω 是资产权重向量, $\mu = (\mu_1, \dots, \mu_p)^T$ 是p维资产的超额均值回报向量, Σ 是资产超额回报的协方差阵, μ_0 是投资者对资产组合的期望回报。

尽管M-V模型的引入为投资者做出投资策略提供了数学依据, 但随着金融产品数量的不断增加, 由M-V模型产生的最优资产组合往往不够稀疏 [2], 这使得M-V模型在应用于大规模资产管理时存在致命缺陷 [3]。为了克服M-V模型在高维问题下产生的缺陷, 学者们尝试在M-V模型上加入

正则项 [4–8] 以达到更好的稀疏效果, 研究结果显示, 加入 l_p 范数或者稀疏算子会让模型产生的资产稀疏且具有比较好的金融学性质。但随着研究的不断开展, 使用正则项等方法的研究逐渐趋于完善, 因此近期许多学者尝试着转换方向, 从原始M-V 模型解的估计入手, 试图从新的角度来研究M-V 模型。

M-V模型的解可以写为下面的形式:

$$\omega_{opt} = \frac{\mu_0 \Sigma^{-1} \mu}{\mu^T \Sigma^{-1} \mu}, \quad (1.2)$$

则对M-V模型的求解可以转换为对 Σ 和对 Σ^{-1} (又被叫做精确矩阵) 的估计, 在一般情况下, 通常可以使用样本收益的协方差阵对 Σ 进行估计, 用样本收益的协方差阵的逆对 Σ^{-1} 进行估计, 但在高维问题中, $\hat{\Sigma}^{-1}$ 难以被计算出来, 于是, 如何对 Σ 和对 Σ^{-1} 的进行准确估计从而能得到M-V模型的最优解, 是高维资产组合求解中面临的重要问题。

包括但不限于At-Sahalia [9]等人 (2011)、Frahm [10]和Memmel (2010)、Engle [11]等人 (2019)、Boudt [12]等人 (2017)、Cai [13]等人 (2020)、Ding [14]等人 (2021) 和So [15]等人 (2022) 对如何估计 Σ 进行了研究。研究发现在某些正则性条件下, 上述工作中的估计收敛于真实协方差矩阵。但他们的研究重点在于对估计量的概率收敛性质进行证明, 没有将得到的投资组合和其他策略得到的进行对比, 所以仍不清楚导出的投资组合配置是否提供了最优策略。

另一方面, 在2011年和2019年, Cai [16]和Liu, Pun [17]和Wong对精确矩阵 Σ^{-1} 的估计进行了研究。他们的研究首先令 $\Omega = \Sigma^{-1}$, 根据 (1.2), 最优权重 ω_{opt} 与 $\Omega \mu$ 成比例, 因此他们定义 $\beta = \Omega \mu$ 为实现(1.1)这个优化问题的有效参数。Pun等提出基于带约束 l_1 最小化方法的LPO估计量, 这个方法是根据下面的模型直接对 β 进行估计

$$\begin{aligned} \min_{\beta \in R^p} \quad & \|\beta\|_1 \\ \text{s.t.} \quad & \left\| \hat{\Sigma} \beta - \hat{\mu} \right\|_{\infty} \leq \lambda. \end{aligned} \quad (1.3)$$

其中 λ 是一个调整型参数, $\hat{\mu} = (\hat{\mu}_1, \dots, \hat{\mu}_p)$ 是样本超额回报均值向量, $\hat{\Sigma}$ 是样本超额回报的协方差, $\|a\|_{\infty} = \max_i |a_i|$, 根据这个模型, 稀疏性假设可以直接加在 β 上而不是 μ 上和 Ω 上。Cai和Liu [16]的研究显示, 即使当 μ 或 Σ 不能被很好的单独估计, β 也能被直接和有效的被估计出来。并且, Pun [17]和Wong进一步的证明, LPO 具有渐进最优等良好的统计性质。

最小化问题 (1.3) 是一个非光滑的凸最小化问题, 在资产数量不是很大时, 传统的算法例如线性规划和经典ADMM算法可以有效求解这个问题。然而, 随着资产维度的增加, 这些经典算法的效率会变得很低。因此, 在2022年, Cai和Li [18]受到统计学上“特征分割”(featuree-splitting) 的启发, 提出了分割资产的ADMM算法 (AS-ADMM) 希望解决这个问题, 其思路是首先对资产进行分组处理, 在每个组上并行执行计算, 达到减少计算时间和提高运行效率的目的。然而, 从实验结果来看, 对于大规模的资产选择问题该方法仍然不够有效。

受到外推加速思想的启发, 本文期望在AS-ADMM的基础上加入外推项, 在保证理论收敛的同时, 也能够提高算法在求解超高维投资组选择问题时的效率。

2. 预备知识

下面简单介绍与本文相关的一些符号与定义。

定义 R^n 为 n 维欧式空间, $\langle \cdot, \cdot \rangle$ 为标准内积, I_n 是维度为 n 的单位矩阵, $\|\cdot\|_\infty, \|\cdot\|_1$ 分别为无穷范数与 l_1 范数, 集合 S 上的示性函数 $\delta_S(x)$ 定义为

$$\delta_S(x) := \begin{cases} 0, & \text{if } x \in S, \\ +\infty, & \text{otherwise.} \end{cases} \quad (2.1)$$

3. 主要算法及定理

在本节中, 我们将会给出求解问题 (1.3) 算法的具体迭代框架, 并对算法的收敛速率进行证明。

Cai [18]等使用了辅助向量 $m = \hat{\Sigma}\beta - \hat{\mu}$, 从而可以将约束条件 $\|\hat{\Sigma}\beta - \hat{\mu}\|_\infty \leq \lambda$ 加入问题 (1.3) 的目标函数中。并且为了使用算法并行的思想, 他们对 β 和 $\hat{\Sigma}$ 进行了分组。此外, 他们加入了松弛变量 $\gamma = (\gamma_2, \dots, \gamma_N), \gamma_i \in R^{p_i}, i = 2, \dots, N$, 使得问题变为3块可分离问题。因此, 问题 (1.3) 可以写为以下形式:

$$\begin{aligned} \min_{\beta, m, \gamma} \quad & \sum_{i=1}^N \|\alpha_i \circ \beta_i\|_1 + \delta_{\mathcal{M}_0}(m) \\ \text{s.t.} \quad & \hat{\Sigma}_1\beta_1 + \gamma_2 + \dots + \gamma_N - m = \hat{\mu}, \\ & \hat{\Sigma}_i\beta_i = \gamma_i, i = 2, \dots, N. \end{aligned} \quad (3.1)$$

其中, 有效参数 β 被分为 $N > 1$ 组, $\beta = (\beta_1^T, \dots, \beta_N^T)$, 矩阵 $\hat{\Sigma} = (\hat{\Sigma}_1, \dots, \hat{\Sigma}_N)$, $\hat{\Sigma}_i \in R^{p \times p_i}$, $\beta_i \in R^{p_i}$, $\sum_{i=1}^N p_i = p$, $\alpha_i \circ \beta_i = (\alpha_{i1}\beta_{i1}, \dots, \alpha_{ip_i}\beta_{ip_i})$, $\mathcal{M}_0 = \{m : |m_s| \leq \lambda\alpha_s, s = 1, \dots, p\}$, 并且 α_s 是 $\alpha = (\alpha_1, \dots, \alpha_N)$ 的第 s 个元素。

问题 (3.1) 的增广拉格朗日函数为:

$$\begin{aligned} \mathcal{L}_\phi(\beta, m, \gamma; z) = & \sum_{i=1}^N \|\alpha_i \circ \beta_i\|_1 + \delta_{\mathcal{M}_0}(m) + z_1^T(\hat{\Sigma}_1\beta_1 + \sum_{i=2}^N \gamma_i - m - \hat{\mu}) + \sum_{i=2}^N z_i^T(\hat{\Sigma}_i\beta_i - \gamma_i) \\ & + \frac{\phi}{2} \left\| \hat{\Sigma}_1\beta_1 + \sum_{i=2}^N \gamma_i - m - \hat{\mu} \right\|^2 + \frac{\phi}{2} \sum_{i=2}^N \left\| \hat{\Sigma}_i\beta_i - \gamma_i \right\|^2. \end{aligned} \quad (3.2)$$

为了求解问题 (3.1), Cai等在文献 [18]提出了资产分割ADMM算法 (AS-ADMM)。然而在高维情况下, AS-ADMM的效率不高, 因此我们采用了Li [19]等提出的外推技巧。并且为了保证算法的收敛性, 我们对松弛变量 γ 进行了三次更新, 并将构建的外推加速项只用于部分变量的更新中而不是全部。

我们提出了部分加速的资产分割ADMM算法 (PA-AS-ADMM), 其具体迭代框架如下:

步骤0 输入 $\phi > 0, 0 < \tau < 1, t_0 = 1, \beta^0, z^0, \gamma^0, m^0$. 设 $k = 0$.

步骤1 当算法不满足终止条件时:

通过 $\frac{1-t_k}{t_k} = \frac{1}{t_{k-1}} - \tau$ 求解 t_k ,

计算外推项 $v^k = m^k + t_k \frac{(1-t_{k-1})}{t_{k-1}} (m^k - m^{k-1})$.

步骤2 计算

$$\gamma_i^{k+\frac{1}{3}} = \frac{1}{N} (\hat{\mu} + v^k + N \hat{\Sigma}_i \beta_i^k - \hat{\Sigma} \beta^k), i = 2, \dots, N,$$

$$\beta_1^{k+1} = S(\beta_{1j}^k - \frac{\phi}{\eta_1} \hat{\Sigma}_{1j}^T (\hat{\Sigma}_1 \beta_1^k + \sum_{i=2}^N \gamma_i^k - v^k - \hat{\mu} + \frac{z_1^k}{\phi}), \frac{\alpha_{1j}}{\eta_1})_{j=1, \dots, p_1},$$

$$\beta_i^{k+1} = S(\beta_{ij}^k - \frac{\phi}{\eta_i} \hat{\Sigma}_{ij}^T (\hat{\Sigma}_i \beta_i^k - \gamma_i^k + \frac{z_i^k}{\phi}), \frac{\alpha_{ij}}{\eta_i})_{j=1, \dots, p_i}, i = 2, \dots, N,$$

$$\gamma_i^{k+\frac{2}{3}} = \frac{1}{N} (\hat{\mu} + m^k + N \hat{\Sigma}_i \beta_i^{k+1} - \hat{\Sigma} \beta^{k+1}), i = 2, \dots, N,$$

$$m^{k+1} = \min(\max(\hat{\Sigma}_1 \beta_1^{k+1} + \sum_{i=2}^N \gamma_i^{k+\frac{2}{3}} - \hat{\mu} + \frac{z_1^k}{\phi}, -\lambda\alpha), \lambda\alpha),$$

$$\gamma_i^{k+1} = \frac{1}{N} (\hat{\mu} + m^{k+1} + N \hat{\Sigma}_i \beta_i^{k+1} - \hat{\Sigma} \beta^{k+1}), i = 2, \dots, N,$$

$$z_1^{k+1} = z_1^k + \tau\phi(\hat{\Sigma}_1 \beta_1^{k+1} + \sum_{i=2}^N \gamma_i^{k+1} - m^{k+1} - \hat{\mu}),$$

$$z_i^{k+1} = z_i^k + \tau\phi(\hat{\Sigma}_i \beta_i^{k+1} - \gamma_i^{k+1}), i = 2, \dots, N.$$

转步骤3.

步骤3 令 $k \leftarrow k + 1$ 转步骤1.

为了方便证明PA-AS-ADMM算法的收敛速率, 我们将先说明PA-AS-ADMM这个具有三块结构 (β, γ, m) 的算法和PA-AS-ADMM2这个具有两块结构 (β, m) 的算法具有等价性, 然后再证明PA-AS-ADMM2的收敛速率从而得到PA-AS-ADMM算法的收敛速率. PA-AS-ADMM2求解问题的增广拉格朗日函数为:

$$\begin{aligned} \hat{\mathcal{L}}_\phi(\beta, m; r) &= \sum_{i=1}^N \|\alpha_i \circ \beta_i\|_1 + \delta_{\mathcal{M}_0}(m) + r_1^T (\hat{\Sigma}_1 \beta_1 - m - \hat{\mu}) + \sum_{i=2}^N r_i^T (\hat{\Sigma}_i \beta_i) \\ &\quad + \frac{\phi}{2} \left\| \hat{\Sigma}_1 \beta_1 - m - \hat{\mu} \right\|^2 + \frac{\phi}{2} \sum_{i=2}^N \left\| \hat{\Sigma}_i \beta_i \right\|^2. \end{aligned} \tag{3.3}$$

为了简化符号, 我们记

$$F = \text{Diag}(A_1, \dots, A_N), b = \hat{\mu},$$

$$G = (I_n, 0, \dots, 0)^T,$$

$$Q = I - H'(HH')^{-1}H,$$

$$H = \begin{pmatrix} I_n & I_n & \cdots & I_n \\ -I_n & 0 & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & -I_n & 0 \\ 0 & 0 & \cdots & -I_n \end{pmatrix},$$

并且

$$P_f = \rho_f I - \phi F' F, \rho_f > 2\phi \lambda_{\max}(F' F),$$

$$P_g = \rho_g I - \phi F' F, \rho_g > 2\phi \lambda_{\max}(G' G).$$

求解问题 (3.1) 等价问题的PA-AS-ADMM2具体框架如下:

步骤0 输入 $\phi > 0, 0 < \tau < 1, t_0 = 1, \beta^0, w^0, r^0$.

步骤1 当算法不满足终止条件时:

$$\text{通过 } \frac{1-t_k}{t_k} = \frac{1}{t_{k-1}} - \tau \text{ 求解 } t_k,$$

$$\text{计算外推项 } v^k = m^k + t_k \frac{(1-t_{k-1})}{t_{k-1}} (m^k - m^{k-1}).$$

步骤2 计算

$$\beta_1^{k+1} = S(\beta_{1j}^k - \frac{\phi}{\eta_1} \hat{\Sigma}_{1j}^T (\hat{\Sigma}_1 \beta_1^k - v^k - \hat{\mu} + \frac{z_1^k}{\phi}), \frac{\alpha_{1j}}{\eta_1})_{j=1, \dots, p_1},$$

$$\beta_i^{k+1} = S(\beta_{ij}^k - \frac{\phi}{\eta_i} \hat{\Sigma}_{ij}^T (\hat{\Sigma}_i \beta_i^k + \frac{z_i^k}{\phi}), \frac{\alpha_{ij}}{\eta_i})_{j=1, \dots, p_i}, i = 2, \dots, N,$$

$$m^{k+1} = \min(\max(\hat{\Sigma}_1 \beta_1^{k+1} - \hat{\mu} + \frac{z_1^k}{\phi}, -\lambda\alpha), \lambda\alpha),$$

$$r_1^{k+1} = r_1^k + \tau\phi Q(\hat{\Sigma}_1 \beta_1^{k+1} - m^{k+1} - \hat{\mu}),$$

$$r_i^{k+1} = r_i^k + \tau\phi Q \hat{\Sigma}_i \beta_i^{k+1}, i = 2, \dots, N.$$

转步骤3.

步骤3 令 $k \leftarrow k + 1$ 转步骤1.

接下来, 我们将提出下面的命题说明PA-AS-ADMM和PA-AS-ADMM2是等价的.

命题3.1. (PA-AS-ADMM算法和PA-AS-ADMM2算法等价性) 给定参数 $\phi > 0$ and $0 < \tau < 1$, $\beta^0 \in \text{dom}(f)$, $m^0 \in \text{dom}(g)$ 以及 $r^0 \in \text{Range}(Q)$, 选择 $z^0 = H'(HH')^{-1}b + r^0$. 对于给定的任意 $k \geq 0$, 我们给出以下结论:

(i) PA-AS-ADMM2算法产生的序列 $\{(\beta^k, m^k)\}$ 和PA-AS-ADMM算法产生的序列 $\{(\beta^k, m^k)\}$ 是一致的;

(ii) z^k 和 r^k 之间存在固定的等价关系: $z^k = r^k + H'(HH')^{-1}b$.

证明. 首先, λ^k 在 Q 的范围内. 根据文献 [20] 的结论, $H z^0 = b$ 并且 $H z^k = b$. 因为 $\gamma^{k+\frac{1}{3}} = (HH')^{-1}H(c - F'\beta^k - G'v^k)$, 可以得到

$$H'\gamma^{k+\frac{1}{3}} = PH'\gamma^{k+\frac{1}{3}} = P(c - F'\beta^k - G'v^k).$$

使用 PA-AS-ADMM2 中关于 β 的更新步骤, 可以得到

$$\begin{aligned} \beta^{k+1} &= \operatorname{argmin}_{\beta} \left\{ \hat{\mathcal{L}}_{\phi}(\beta, v^k; r^k) + \frac{\phi}{2} \left\| F'(\beta - \beta^k) \right\|_P^2 + \frac{\phi}{2} \left\| \beta - \beta^k \right\|_{P_f}^2 \right\} \\ &= \operatorname{argmin}_{\beta} \left\{ f(\beta) + g(v^k) + \langle \bar{b}, F'\beta + G'v^k - c \rangle + \langle r^k, Q(F'\beta + G'v^k - c) \rangle \right. \\ &\quad \left. + \frac{\phi}{2} \left\| Q(F'\beta + G'v^k - c) \right\|^2 + \frac{\phi}{2} \left\| P(F'\beta + G'v^k + H'\gamma^{k+\frac{1}{3}} - c) \right\|^2 + \frac{\phi}{2} \left\| \beta - \beta^k \right\|_{P_f}^2 \right\} \\ &= \operatorname{argmin}_{\beta} \left\{ f(\beta) + g(v^k) + \langle z^k, F'\beta + G'v^k - c \rangle + \frac{\phi}{2} \left\| Q(F'\beta + G'v^k + H'\gamma^{k+\frac{1}{3}} - c) \right\|^2 \right. \\ &\quad \left. + \frac{\phi}{2} \left\| P(F'\beta + G'v^k + H'\gamma^{k+\frac{1}{3}} - c) \right\|^2 + \frac{\phi}{2} \left\| \beta - \beta^k \right\|_{P_f}^2 \right\} \\ &= \operatorname{argmin}_{\beta} \left\{ f(\beta) + g(v^k) - \langle b, \gamma^{k+\frac{1}{3}} \rangle + \langle z^k, F'\beta + G'v^k + H'\gamma^{k+\frac{1}{3}} - c \rangle \right. \\ &\quad \left. + \frac{\phi}{2} \left\| F'\beta + G'v^k + H'\gamma^{k+\frac{1}{3}} - c \right\|^2 + \frac{\phi}{2} \left\| \beta - \beta^k \right\|_{P_f}^2 \right\} \\ &= \operatorname{argmin}_{\beta} \left\{ \mathcal{L}_{\phi}(\beta, v^k, \gamma^{k+\frac{1}{3}}; z^k) + \frac{\phi}{2} \left\| \beta - \beta^k \right\|_{P_f}^2 \right\}, \end{aligned}$$

可以看到, 最后一个等式是 PA-AS-ADMM 中关于 β 的更新步骤. 因此, 我们可以得到 PA-AS-ADMM2 产生的 β^k 与 PA-AS-ADMM 产生的 β^k 等价的结论. 因为

$$\begin{aligned} w^{k+\frac{2}{3}} &= \operatorname{argmin}_w \mathcal{L}_{\phi}(x^{k+1}, y^k, w; z^k) \\ &= (HH')^{-1}H(c - F'x^{k+1} - G'y^k), \end{aligned}$$

同样地, 使用 PA-AS-ADMM2 关于 m^k 的更新, 可以得到

$$\begin{aligned} m^{k+1} &= \operatorname{argmin}_m \left\{ \hat{\mathcal{L}}_{\phi}(\beta^{k+1}, m; r^k) + \frac{\phi}{2} \left\| G'(m - m^k) \right\|_P^2 + \frac{\phi}{2} \left\| m - m^k \right\|_{P_g}^2 \right\} \\ &= \operatorname{argmin}_m \left\{ \hat{\mathcal{L}}_{\phi}(\beta^{k+1}, m; r^k) + \frac{\phi}{2} \left\| P(F'\beta^{k+1} + G'm + H'\gamma^{k+\frac{2}{3}} - c) \right\|^2 + \frac{\phi}{2} \left\| m - m^k \right\|_{P_g}^2 \right\} \\ &= \operatorname{argmin}_m \left\{ \mathcal{L}_{\phi}(\beta^{k+1}, m, \gamma^{k+\frac{2}{3}}; z^k) + \frac{\phi}{2} \left\| m - m^k \right\|_{P_g}^2 \right\}, \end{aligned}$$

其中最后一个等式是 PA-AS-ADMM 中关于 m 的更新步骤. 根据这个关系, 我们可以证明命题的 (i) 部分.

根据PA-AS-ADMM和PA-AS-ADMM2关于对偶部分的更新, 我们有

$$\begin{aligned} r^{k+1} &= r^k + \tau\phi Q(F' \beta^{k+1} + G' m^{k+1} - c) \\ &= z^k - H'(HH')^{-1}b + \tau\phi(F' \beta^{k+1} + G' m^{k+1} + H' \gamma^{k+1} - c) \\ &= z^{k+1} - H'(HH')^{-1}b, \end{aligned}$$

其中 $\gamma^{k+1} = \operatorname{argmin}_{\gamma} \mathcal{L}_{\phi}(\beta^{k+1}, m^{k+1}, \gamma; z^{k+1}) = (HH')^{-1}H(c - F' \beta^{k+1} - G' m^{k+1})$, 因此我们证明完毕。□

利用我们提出的命题 (3.1), PA-AS-ADMM和PA-AS-ADMM2是等价的。因此, 我们只要证明出PA-AS-ADMM2的收敛速率, 也就能证明PA-AS-ADMM的收敛。根据参考文献 [19]中的定理3.11, 我们可以得到下面的引理。

引理3.2. 当 $0 < \tau < 1$ 时, PA-AS-ADMM2的非遍历收敛速率为 $O(\frac{1}{K})$ 。

结合命题3.1和引理1的结论, 我们可以得到下面的定理:

定理3.3. 当 $0 < \tau < 1$ 时, PA-AS-ADMM的非遍历收敛速率为 $O(\frac{1}{K})$ 。

4. 数值实验

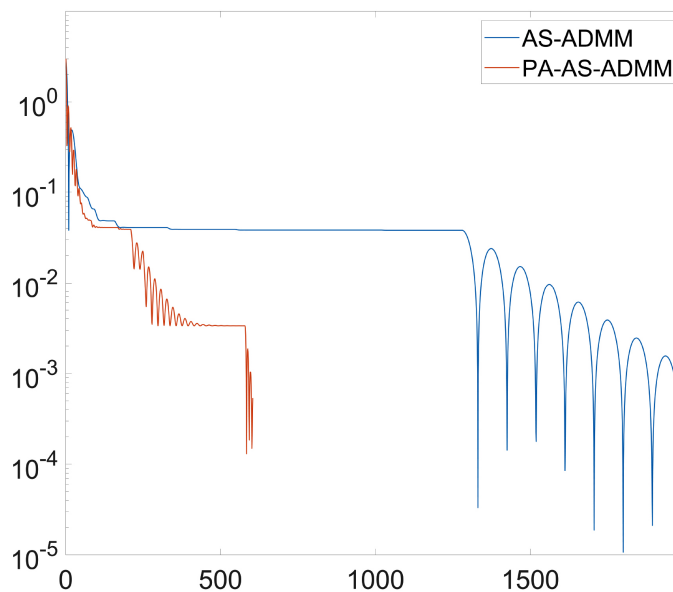


Figure 1. The relationship between the primal residuals and the number of iterations in AS-ADMM and PA-AS-ADMM when $p = 2000, n = 200, e^{abs} = 1e - 4, e^{rel} = 1e - 4$

图 1. $p = 2000, n = 200, e^{abs} = 1e - 4, e^{rel} = 1e - 4$ 时AS-ADMM和PA-AS-ADMM原始残差和迭代次数之间的关系

Table 1. The efficiency comparison between PA-AS-ADMM and AS-ADMM at $e^{abs} = 1e - 3$ and $e^{rel} = 1e - 3$
表 1. PA-AS-ADMM 和AS-ADMM在 $e^{abs} = 1e - 3$ 和 $e^{rel} = 1e - 3$ 条件下的运行效率

Algorithm	迭代次数		CPU 时间(秒)	
	AS-ADMM	PA-AS-ADMM	AS-ADMM	PA-AS-ADMM
p=500	831	418	4.37	2.11
p=1000	968	475	29.96	12.36
p=2000	1193	709	194.38	108.56

Table 2. The efficiency comparison between PA-AS-ADMM and AS-ADMM at $e^{abs} = 1e - 4$ and $e^{rel} = 1e - 4$
表 2. PA-AS-ADMM 和AS-ADMM在 $e^{abs} = 1e - 4$ 和 $e^{rel} = 1e - 4$ 条件下的运行效率

Algorithm	迭代次数		CPU 时间(秒)	
	AS-ADMM	PA-AS-ADMM	AS-ADMM	PA-AS-ADMM
p=500	1212	535	6.04	2.53
p=1000	1389	625	34.82	23.90
p=2000	1270	684	371.08	145.18

在数值实验中, 我们使用多元正态分布 $N(\mu, \Sigma)$ 来生成市场超额收益率模拟数据, 其中超额收益率均值向量为 $\mu = (0, \dots, 0, 1, \dots, 1)^T$, 并且我们设置后10个元素非零, 超额收益率协方差矩阵为 $\Sigma = (\rho_{ij})_{p \times p}$, 其中, $\rho_{ij} = 0.25^{|i-j|}$ 对任意 $1 \leq i, j \leq p$ 。我们设置样本量为 $n = 200$, 资产分组数量 $N = 8$, 资产数量 p 分别等于500, 1000, 2000。在算法的终止条件选取上, 我们参照了参考文献 [21]中的方法, 将在原始残差和对偶残差均低于一个足够小的数时终止算法, 并且这个数取决于 e^{abs} 和 e^{rel} 的选择。我们分别设置 $e^{abs} = 1e - 3, 1e - 4, e^{rel} = 1e - 3, 1e - 4$ 。此外, 为了避免算法过度迭代, 当迭代次数达到5000次时, 我们也会终止算法。

为了更加直观地看到AS-ADMM和PA-AS-ADMM在高维条件下以及高收敛精度下产生的原始残差与迭代次数之间的关系, 如图1所示, 我们绘制了当参数设置为 $e^{abs} = 1e - 4, e^{rel} = 1e - 4, n = 200, p = 2000$ 时两个算法的收敛过程。显然, PA-AS-ADMM在相同参数水平下能够更快地达到指定的收敛精度。

为了进一步比较AS-ADMM和PA-AS-ADMM的算法效率, 我们在上述设置的不同参数条件下各进行了100次随机实验。实验得到的结果分别示于表1和表2中。从表格数据可以初步看到, 在相同的问题规模下PA-AS-ADMM明显比AS-ADMM花费更少的CPU时间和迭代次数。更具体地说, 在更高维条件($p = 2000$)和更高精度要求($e^{abs} = 1e - 4, e^{rel} = 1e - 4$)下, PA-AS-ADMM比AS-ADMM的迭代次数可以减少一半左右, CPU运行时间减少三分之一左右。这说明了我们的算法比AS-ADMM具有更高的运行效率。

5. 结论

本文针对高维M-V模型下的算法效率不高的问题, 使用了部分外推的加速思想, 提出了部分加速资产分割ADMM算法 (PA-AS-ADMM)。不仅在理论上证明了该算法的非遍历收敛速率

为 $O(\frac{1}{K})$ ，还通过数值模拟实验验证了该算法的有效性。

参考文献

- [1] Markowitz, H.M. (1952) Portfolio Selection. *The Journal of Finance*, **7**, 77-91.
<https://doi.org/10.1111/j.1540-6261.1952.tb01525.x>
- [2] Bai, Z., Liu, H. and Wong, W.K. (2009) Enhancement of the Applicability of Markowitz's Portfolio Optimization by Utilizing Random Matrix Theory. *Mathematical Finance*, **19**, 639-667. <https://doi.org/10.1111/j.1467-9965.2009.00383.x>
- [3] Fan, J., Liao, Y. and Shi, X. (2015) Risks of Large Portfolios. *Journal of Econometrics*, **186**, 367-387. <https://doi.org/10.1016/j.jeconom.2015.02.015>
- [4] Lorenzo, D., Liuzzi, G., Rinaldi, F., Schoen, F. and Sciandrone, M. (2012) A Concave Optimization-Based Approach for Sparse Portfolio Selection. *Optimization Method and Software*, **27**, 983-10000. <https://doi.org/10.1080/10556788.2011.577773>
- [5] DeMiguel, V., Garlappi, L., Nogales, J. and Uppal, R. (2009) A Generalized Approach to Portfolio Optimization: Improving Performance by Constraining Portfolio Norms. *Management Science*, **55**, 798-812. <https://doi.org/10.1287/mnsc.1080.0986>
- [6] Brodie, J., Daubechies, I., DeMol, C., Giannone, D. and Loris, D. (2008) Sparse and Stable Markowitz Portfolios. ECB Working Paper Series.
- [7] Still, S. and Kondor, I. (2010) Regularizing Portfolio Optimization. *New Journal of Physics*, **12**, Article 075034. <https://doi.org/10.1088/1367-2630/12/7/075034>
- [8] Brandt, M.W., Pedro, S.C. and Rossen, V. (2009) Parametric Portfolio Policies: Exploiting Characteristics in the Cross Section of Equity Returns. *Review of Financial Studies*, **22**, 3411-3447. <https://doi.org/10.1093/rfs/hhp003>
- [9] At-Sahalia, Y., Mykland, P.A. and Zhang, L. (2011) Ultra High Frequency Volatility Estimation with Dependent Microstructure Noise. *Journal of Econometrics*, **160**, 160-175.
<https://doi.org/10.1016/j.jeconom.2010.03.028>
- [10] Frahm, G., *et al.* (2010) Dominating Estimators for Minimum-Variance Portfolios. *Journal of Econometrics*, **159**, 289-302. <https://doi.org/10.1016/j.jeconom.2010.07.007>
- [11] Engle, R.F., Ledoit, O., *et al.* (2019) Large Dynamic Covariance Matrices. *Journal of Business & Economic Statistics*, **37**, 363-375. <https://doi.org/10.1080/07350015.2017.1345683>
- [12] Boudt, K., *et al.* (2017) Positive Semidefinite Integrated Covariance Estimation, Factorizations and Asynchronicity. *Journal of Econometrics*, **196**, 347-367.
<https://doi.org/10.1016/j.jeconom.2016.09.016>
- [13] Cai, T.T., *et al.* (2020) High-Dimensional Minimum Variance Portfolio Estimation Based on High-Frequency Data. *Journal of Econometrics*, **214**, 482-494.
<https://doi.org/10.1016/j.jeconom.2019.04.039>

-
- [14] Ding, Y., *et al.* (2021) High Dimensional Minimum Variance Portfolio Estimation under Statistical Factor Models. *Journal of Econometrics*, **222**, 502-515.
<https://doi.org/10.1016/j.jeconom.2020.07.013>
- [15] So, M.K.P., Chan, T.W.C. and Chu, A.M.Y. (2022) Efficient Estimation of High-Dimensional Dynamic Covariance by Risk Factor Mapping: Applications for Financial Risk Management. *Journal of Econometrics*, **227**, 151-167. <https://doi.org/10.1016/j.jeconom.2020.04.040>
- [16] Cai, T. and Liu, W. (2011) A Direct Estimation Approach To Sparse Linear Discriminant Analysis. *Journal of the American Statistical Association*, **106**, 1566-1577.
<https://doi.org/10.1198/jasa.2011.tm11199>
- [17] Pun, C. and Wong, H. (2019) A Linear Programming Model for Selection of Sparse High-Dimensional Multiperiod Portfolios. *European Journal of Operational Research*, **273**, 754-771.
<https://doi.org/10.1016/j.ejor.2018.08.025>
- [18] Cai, Z., *et al.* (2024) Asset Splitting Algorithm for Ultrahigh Dimensional Portfolio Selection and Its Theoretical Property. *Journal of Econometrics*, **239**, Article 105291.
<https://doi.org/10.1016/j.jeconom.2022.04.004>
- [19] Li, H. and Lin, Z. (2019) Accelerated Alternating Direction Method of Multipliers: An Optimal $O(1/K)$ Nonergodic Analysis. *Journal of Scientific Computing*, **79**, 671-699.
<https://doi.org/10.1007/s10915-018-0893-5>
- [20] Sun, D., Toh, D.C. and Yang, L. (2015) A Convergent 3-Block Semiproximal Alternating Direction Method of Multipliers for Conic Programming with 4-Type Constraints. *SIAM Journal on Optimization*, **25**, 882-915. <https://doi.org/10.1137/140964357>
- [21] Boyd, S., Parikh, N. and Chu, E. (2011) Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, **3**, 1-122. <https://doi.org/10.1561/22000000016>