

求解约束最小二乘半正定规划问题的 L-BFGS 方法

樊长幸, 沈春根, 王云龙

(上海理工大学 理学院, 上海 200093)

摘要: 对带等式和不等式约束的最小二乘半正定规划问题的求解进行了研究。在 Slater 约束规范条件下, 对偶问题的最优解与原问题最优解相等。因此, 考虑将最小二乘半正定规划问题转化为相应的对偶问题, 通过求解对偶问题达到求解原问题的目的。针对最小二乘半正定规划问题的对偶问题, 首先构造相应的二次模型, 沿负梯度方向最小化该二次模型得到柯西点, 在此基础上, 利用积极约束技巧, 划分积极约束集与非积极约束集, 然后应用 L-BFGS 技巧对自由变量进行加速, 从而求得对偶问题的最优解。最后, 从理论上证明了算法的全局收敛性, 并进行了初步的数值实验, 将该算法与光滑化牛顿法作对比, 结果表明该算法在计算时间上有一定的优势。

关键词: 对偶问题; 梯度投影法; L-BFGS 算法; 柯西点; 全局收敛性

中图分类号: O 221.2

文献标志码: A

L-BFGS Method for Constrained Least Squares Semidefinite Programming

FAN Changxing, SHEN Chungen, WANG Yunlong

(College of Science, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: The solution of the least squares semidefinite programming problem with equality and inequality constraints was studied. Under the Slater constraint specification, the optimum solutions for the dual and original problems are the same. Therefore, the least squares semidefinite programming problem was transformed into its corresponding dual problem, and the original problem was solved by solving the dual problem. For the solution of the dual problem of the least squares semidefinite programming problem, the quadratic model was constructed, and Cauchy point was obtained by minimizing the quadratic model along the negative gradient direction. On this basis, the positive constraint set and the non-active constraint set were divided by using positive constraint technique. Then, the L-BFGS technique was applied to accelerate the free variables to obtain the optimum solution of the dual problem. Finally, the global convergence of the algorithm was proved theoretically, and a preliminary numerical experiment was carried out to compare the algorithm with the smoothed Newton method. The results show that the algorithm has certain advantages in computing time.

Keywords: dual problem; gradient projection method; L-BFGS algorithm; Cauchy point; global convergence

收稿日期: 2018-05-24

第一作者: 樊长幸(1991-), 男, 硕士研究生. 研究方向: 最优化理论. E-mail: fcxwork2018@163.com

通信作者: 沈春根(1981-), 男, 副教授. 研究方向: 最优化理论. E-mail: shenchungen@usst.edu.cn

Frobenius 范数矩阵逼近问题是近年来学者们研究的热门课题，在图像处理、统计、金融与保险等领域有着十分广泛的应用。多年以来许多国内外学者一直致力于研究矩阵逼近问题，并取得了丰硕的成果。Higham^[1]使用一种加权 Frobenius 范数研究了相关系数矩阵最佳逼近问题，并提出使用修改的交替投影法来计算该问题。Malick^[2]提出了一种基于拉格朗日对偶的算法来解决等式约束最小二乘问题。Boyd 等^[3]研究了 Frobenius 范数意义下的最小二乘协方差矩阵逼近问题，并通过对偶的方法来求解此问题。Gao 等^[4]考虑具有等式和不等式约束的最小二乘半正定规划，通过将偶问题转化为带有投影算子的半光滑方程组来解决。他们设计一个非精确的光滑牛顿法来解决产生的半光滑系统，在每次迭代中，使用 BiCGStab 迭代求解器来获得光滑牛顿线性系统的近似解，数值实验验证了该方法的高效性。He 等^[5]证明了交替方向法在解决最小二乘半定规划问题上的适用性，并检验了交替方向法的有效性。Li 等^[6]提出了一种利用投影半光滑牛顿法求解带有等式不等式约束的最佳相关系数矩阵逼近问题，该方法在适当的假设下具有全局收敛性和二次收敛速度。Savas 等^[7]提出了一种新的 Frobenius 范数下的聚类矩阵逼近框架，该框架非常灵活，可以按各种方式进行修改以适应特定应用的需要。

本文考虑应用对偶技术研究 Frobenius 范数意义下的矩阵逼近问题。对偶问题是带非负约束的凸优化问题，在 Slater 的条件下，对偶问题的最优解与原问题最优解相等。因此，考虑将原问题转化为相应的对偶问题，通过求解对偶问题可以达到求解原问题的目的。为了处理大规模对偶问题，本文利用积极约束技巧估计非负约束，运用 L-BFGS 方法对自由变量进行加速，并证明了算法的全局收敛性。

1 算法构造

考虑 Frobenius 范数意义下的矩阵逼近问题

$$\begin{cases} \min \left(\frac{1}{2} \|X - G\|_F^2 \right) \\ \text{s.t. } \langle A_i, X \rangle \geq b_i, i \in I \\ \langle A_j, X \rangle = b_j, j \in \mathcal{E} \\ X \in S_+^n \end{cases}$$

式中: $I=\{1,\cdots,q\}$; $\mathcal{E}=\{q+1,\cdots,m\}$; $A_i \in S^n$; $m=p+q$, $S^n = \{M \in \mathbb{R}^{n \times n} | M \text{ 是对称矩阵}\}$; $S_+^n = \{M \in \mathbb{R}^{n \times n} | M \text{ 是对称半正定矩阵}\}$;

$b = (b_1, \cdots, b_m)^T \in \mathbb{R}^{n \times n}$ 是给定的向量, $\|\cdot\|_F$ 表示由标准内积 $\langle \cdot, \cdot \rangle$ 诱导的 Frobenius 范数。

根据文献[3]可知上述问题的对偶问题为

$$\begin{cases} \max \left(-\frac{1}{2} \|(G + \mathcal{A}(u) + \mathcal{B}(v))_+\|_F^2 + \right. \\ \left. \frac{1}{2} \|G\|_F^2 + (u^T, v^T)b \right) \\ \text{s.t. } (u; v) \in \Omega \end{cases} \tag{1}$$

式中: $\mathcal{A}(u) = \sum_{i=1}^q u_i A_i$; $\mathcal{B}(v) = \sum_{j=1}^p v_j A_{j+q}$; $(u^T, v^T)b = \sum_{i=1}^q u_i b_i + \sum_{j=1}^p v_j b_{j+q}$; $\Omega = \{(u; v) | u \geq 0, u \in \mathbb{R}^q, v \in \mathbb{R}^p\}$ 。

$$\varphi(u, v) = \frac{1}{2} \|(G + \mathcal{A}(u) + \mathcal{B}(v))_+\|_F^2 - \frac{1}{2} \|G\|_F^2 - (u^T, v^T)b$$

于是，问题(1)可转换为极小化对偶模型

$$\begin{cases} \min \varphi(u, v) \\ \text{s.t. } (u; v) \in \Omega \end{cases} \tag{2}$$

令 $z = (u^T, v^T)^T$, $z^k = ((u^k)^T, (v^k)^T)^T$ 表示第 k 步迭代点, z_c^k 表示第 k 步柯西点^[8]。令 $\varphi(z) = \varphi(u, v)$, $\varphi^k = \varphi(z^k)$, $g^k = (\nabla_u \varphi(z^k)^T, \nabla_v \varphi(z^k)^T)^T$, $g_c^k = \nabla \varphi(z_c^k)$ 。设 B^k 表示 φ 在迭代点 z^k 处 Hessian 矩阵的 L-BFGS 近似, H^k 表示 B^k 的逆矩阵。此外，用下标表示向量的分量，例如： z_i^k 表示 z^k 的第 i 个分量， $[z_c^k]_i$ 表示 z_c^k 的第 i 个分量。

对于该问题的算法构造，分 5 步进行。

步骤 1 计算柯西点。

Byrd 等^[8]采用柯西点估计积极约束集，结果显示柯西点能够有效探测积极约束。据此，采用柯西点处信息确定积极约束集。在 z^k 处的柯西点 z_c^k 是通过利用投影技术，沿着负梯度方向最小化如下二次模型而得：

$$m^k(z) = \varphi(z) + (g^k)^T (z - z^k) + \frac{1}{2} (z - z^k)^T B^k (z - z^k)$$

步骤 2 更新 B^k 和 H^k 。

利用 L-BFGS 算法间接产生 B^k 和 H^k ，详情参见文献[9-10]。

步骤 3 确定积极约束集与非积极约束集。

首先，先给出投影误差估计，对于给定的充分小的 $\varepsilon > 0$ ，令

$$\omega^k = \|z_c^k - P_\Omega(z_c^k - g_c^k)\| \text{ 且 } \varepsilon^k = \min\{\varepsilon, \omega^k\} \tag{3}$$

式中， $P_\Omega(x)$ 表示 x 在 Ω 上的投影。

其次，类似文献[11]，可以根据 ε^k 定义指标集 $C(z_c^k)$ 和 $D(z_c^k)$ ：

$$C(z_c^k) = \{i | 0 \leq [z_c^k]_i \leq \varepsilon^k, i = 1, \dots, q\} \quad (4)$$

$$D(z_c^k) = \{1, \dots, m\} \setminus C(z_c^k) = \{i | [z_c^k]_i > \varepsilon^k, i = 1, \dots, m\} \quad (5)$$

那么 $C(z_c^k)$ 中指标对应的变量被称作积极变量, $D(z_c^k)$ 中指标对应的变量被称作非积极变量。为了获得积极变量的搜索方向, 将积极集 $C(z_c^k)$ 划分为 3 个部分:

$$C_1(z_c^k) = \{i | [z_c^k]_i = 0 \text{ 且 } [g_c^k]_i \geq 0, i = 1, \dots, q\} \quad (6)$$

$$C_2(z_c^k) = \{i | 0 \leq [z_c^k]_i \leq \varepsilon^k \text{ 且 } [g_c^k]_i < 0, i = 1, \dots, q\} \quad (7)$$

$$C_3(z_c^k) = \{i | 0 < [z_c^k]_i \leq \varepsilon^k \text{ 且 } [g_c^k]_i \geq 0, i = 1, \dots, q\} \quad (8)$$

注意到 $C_1(z_c^k)$ 中的变量符合相应的最优性条件, 所以它是合理的; $C_3(z_c^k)$ 中的变量沿着最速下降方向达到或指向边界, 因此, 最速下降方向在子空间中应当回溯以保证其可行; 而 $C_2(z_c^k)$ 中的变量沿着最速下降方向在可行域内部移动。

步骤 4 确定搜索方向。

用 N_c^k 表示列由 $\{e_i \in \mathbb{R}^m | i \in D(z_c^k)\}$ 构成的矩阵。类似地, P_1^k 和 P_2^k 分别表示列由 $\{e_i \in \mathbb{R}^m | i \in C_3(z_c^k)\}$ 和 $\{e_i \in \mathbb{R}^m | i \in C_2(z_c^k)\}$ 构成的矩阵。为了方便起见, 简记 $\bar{H}_c^k = (N_c^k)^T H^k N_c^k$, $D_c^k = D(z_c^k)$, $C_{ic}^k = C_i(z_c^k)$, $i = 1, 2, 3$ 。

定义搜索方向如下:

$$[d_c^k]_i = \begin{cases} 0, & i \in C_{1c}^k \\ -[P_1^k(P_1^k)^T T_c^k g_c^k]_i, & i \in C_{3c}^k \\ -[P_2^k(P_2^k)^T g_c^k]_i, & i \in C_{2c}^k \\ -[N_c^k \bar{H}_c^k (N_c^k)^T g_c^k]_i, & i \in D_c^k \end{cases} \quad (9)$$

$$\text{式中, } T_c^k = \text{diag}([o_c^k]_1, \dots, [o_c^k]_q) \\ [o_c^k]_i = \begin{cases} 0, & i \notin [C_c^k]_3 \\ \frac{[u_c^k]_i}{[g_c^k]_i}, & 0 < [u_c^k]_i \leq \min\{\varepsilon^k, [g_c^k]_i\}, i \in [C_c^k]_3 \\ 1, & \text{其他} \end{cases} \quad (10)$$

步骤 5 运用非单调线搜索确定步长。

在确定好搜索方向后, 还需确定搜索步长。Zhang 等^[12]提出并分析了一种新的非单调线搜索算法, 证明了非凸光滑函数的全局收敛性和强凸函数的线性收敛性。与单调或传统的非单调方法相比, 该非单调线搜索技术一般需要较少的函数和梯度计算次数。因此, 本文采用文献^[12]中非单调线搜索技术确定迭代步的步长。

算法 1

步骤 1 给定初始点 $z^0 \in \Omega$, $0 < \sigma < 1$, $0 < \tau < 1$, $0 < \delta < \frac{1}{2}$, $\varepsilon > 0$, 初始矩阵 H^0 , $c_0 = \varphi(z^0)$, $Q_0 = 1$, $k = 0$;

步骤 2 根据式(4)~(8), 确定 D_c^k , $C_{ic}^k, i = 1, 2, 3$;

步骤 3 参照文献^[8]中方法计算出柯西点 z_c^k , 如果 $\varphi(z_c^k) > \varphi(z^k)$, 则令 $z_c^k = z^k$;

步骤 4 根据式(9)计算搜索方向 d_c^k , 并令 $\alpha = 1$;

步骤 5 如果 $d_c^k = 0$, 那么停止迭代;

步骤 6 如果步长因子 α 满足非单调线搜索条件

$$\varphi(P_\Omega(z_c^k + \alpha d_c^k)) \leq c^k + \alpha \delta (g_c^k)^T d_c^k \quad (11)$$

那么, 令 $\alpha_c^k = \alpha$, 否则令 $\alpha = \sigma \alpha$, 返回步骤 6;

步骤 7 令 $z^{k+1} = P_\Omega(z_c^k + \alpha_c^k d_c^k)$,

$$Q_{k+1} = \tau Q_k + 1, \quad c^{k+1} = \frac{\tau Q_k c^k + \varphi(z_c^{k+1})}{Q_{k+1}} \quad (12)$$

采用文献^[9]中算法 7.4 更新生成 H^{k+1} 的信息;

步骤 8 $k = k + 1$, 返回步骤 2。

2 收敛性分析

引理 1 假定 $\{d_c^k\}$ 和 $\{z_c^k\}$ 由算法 1 生成, H^k 正定, 则 z_c^k 是问题(2)的 KKT 点, 当且仅当 $d_c^k = 0$ 。

证明 首先假定 z_c^k 是式(2)的 KKT 点, 则存在拉格朗日乘子 $[\lambda_c^k]_i \geq 0, i = 1, \dots, q$, 使得

$$\begin{cases} [g_c^k]_i - [\lambda_c^k]_i = 0, & i = 1, \dots, q \\ [g_c^k]_i = 0, & i = q + 1, \dots, m \\ [\lambda_c^k]_i \geq 0, [z_c^k]_i \geq 0, [\lambda_c^k]_i [z_c^k]_i = 0, & i = 1, \dots, q \end{cases}$$

这个方程等价于如下方程:

$$\begin{cases} [g_c^k]_i \geq 0, \text{ 如果 } [z_c^k]_i = 0, i = 1, \dots, q \\ [g_c^k]_i = 0, i = q + 1, \dots, m \\ [g_c^k]_i = 0, \text{ 如果 } [z_c^k]_i > 0, i = 1, \dots, q \end{cases} \quad (13)$$

由于假定 z_c^k 是式(2)的 KKT 点, 则 $[g_c^k]_i = 0, i = q + 1, \dots, m$, 或 $[g_c^k]_i \geq 0, i = 1, \dots, q$ 。由式(6)~(8)及式(13)可知, $C_{2c}^k = \emptyset$ 。由式(9)可知, $[g_c^k]_i = 0, i \in D_c^k \cup C_{3c}^k$, 因此, $d_c^k = 0$ 。

反过来, 假定 $d_c^k = 0$, 由式(9)则有

$$\begin{cases} -\left[\mathbf{P}_1^k(\mathbf{P}_1^k)^T \mathbf{T}_c^k \mathbf{g}_c^k\right]_i = 0, & i \in C_{3c}^k \\ -\left[\mathbf{P}_2^k(\mathbf{P}_2^k)^T \mathbf{g}_c^k\right]_i = 0, & i \in C_{2c}^k \\ -\left[\mathbf{N}_c^k \bar{\mathbf{H}}_c^k (\mathbf{N}_c^k)^T \mathbf{g}_c^k\right]_i = 0, & i \in D_c^k \end{cases}$$

由于 $\bar{\mathbf{H}}_c^k, \mathbf{H}^k$ 正定, 且 $[\mathbf{o}_c^k]_i \neq 0$, 那么

$$\begin{aligned} \left[(\mathbf{P}_1^k)^T \mathbf{g}_c^k\right]_i &= 0, i \in C_{3c}^k \\ \left[(\mathbf{P}_2^k)^T \mathbf{g}_c^k\right]_i &= 0, i \in C_{2c}^k \\ \left[(\mathbf{N}_c^k)^T \mathbf{g}_c^k\right]_i &= 0, i \in D_c^k \end{aligned}$$

当 $i \notin C_{1c}^k$ 时, $[\mathbf{g}_c^k]_i = 0$. 因此式(13)成立, $\mathbf{z}_c^k$ 是式(2)的 KKT 点。

引理 2 如果 $\bar{\mathbf{H}}_c^k$ 是正定的, $\mathbf{d}_c^k$ 是由式(9)定义的, 则它满足 $(\mathbf{g}_c^k)^T \mathbf{d}_c^k \leq 0$, 等号成立当且仅当 $\mathbf{d}_c^k = 0$ 。

证明 由式(9)有

$$\begin{aligned} (\mathbf{g}_c^k)^T \mathbf{d}_c^k &= - \sum_{i \in C_{3c}^k} [\mathbf{g}_c^k]_i \left[\mathbf{P}_1^k(\mathbf{P}_1^k)^T \mathbf{T}_c^k \mathbf{g}_c^k\right]_i - \\ &\sum_{i \in C_{2c}^k} [\mathbf{g}_c^k]_i \left[\mathbf{P}_2^k(\mathbf{P}_2^k)^T \mathbf{g}_c^k\right]_i - \sum_{i \in D_c^k} [\mathbf{g}_c^k]_i \left[\mathbf{N}_c^k \bar{\mathbf{H}}_c^k (\mathbf{N}_c^k)^T \mathbf{g}_c^k\right]_i \leq 0 \end{aligned}$$

于是, $(\mathbf{g}_c^k)^T \mathbf{d}_c^k = 0$, 当且仅当 $\mathbf{d}_c^k = 0$ 时等号成立。

定理 1 假设 $\{\mathbf{d}_c^k\}$ 和 $\{\mathbf{z}_c^k\}$ 由算法 1 产生, 则 $\mathbf{z}_c^k$ 是式(2)的 KKT 点当且仅当 $(\mathbf{g}_c^k)^T \mathbf{d}_c^k = 0$ 。

证明 由引理 1 和引理 2 证得。

引理 3 假设 $\{\mathbf{d}_c^k\}$ 和 $\{\mathbf{z}_c^k\}$ 由算法 1 产生, ε^k 是由式(3)定义, 且 $\mathbf{d}_c^k \neq 0$, 则有

$$\beta_c^k \geq \min \left\{ 1, \frac{\varepsilon^k}{\|\mathbf{d}_c^k\|_\infty} \right\} \quad (14)$$

这里, $\beta_c^k = \sup_{0 \leq \eta \leq 1} \{\eta | 0 \leq \mathbf{z}_c^k + \eta \mathbf{d}_c^k\}$ 。

证明 根据定义, $\mathbf{z}_c^k, \mathbf{z}_c^k + \beta_c^k \mathbf{d}_c^k$ 必须满足式(2)的约束条件, 只需证明下面式子即可:

$$[\mathbf{z}_c^k]_i + [\bar{\beta}_c^k]_i [\mathbf{d}_c^k]_i \in [0, +\infty), i = 1, \dots, q \quad (15)$$

其中, $\bar{\beta}_c^k = \min \left\{ 1, \frac{\varepsilon^k}{\|\mathbf{d}_c^k\|_\infty} \right\}$ 。

如果 $i \in C_{1c}^k$, 则 $[\mathbf{d}_c^k]_i = 0$, 而 $[\mathbf{z}_c^k]_i \geq 0, \bar{\beta}_c^k > 0$, 则 $[\mathbf{z}_c^k]_i + [\bar{\beta}_c^k]_i [\mathbf{d}_c^k]_i \geq 0$; 若 $i \in C_{3c}^k, [\mathbf{d}_c^k]_i \leq 0$, 且 $[\mathbf{z}_c^k]_i + [\mathbf{d}_c^k]_i = 0$, 则 $[\mathbf{z}_c^k]_i + [\bar{\beta}_c^k]_i [\mathbf{d}_c^k]_i \geq [\mathbf{z}_c^k]_i + [\mathbf{d}_c^k]_i = 0$; 若 $i \in C_{2c}^k, [\mathbf{d}_c^k]_i > 0$, 则 $[\mathbf{z}_c^k]_i + [\bar{\beta}_c^k]_i [\mathbf{d}_c^k]_i \geq 0$; 若 $i \in D_c^k, [\mathbf{d}_c^k]_i \neq 0$, 当 $[\mathbf{d}_c^k]_i > 0$ 时, 则 $[\mathbf{z}_c^k]_i + [\bar{\beta}_c^k]_i [\mathbf{d}_c^k]_i \geq 0$; 当 $[\mathbf{d}_c^k]_i < 0$, 则

$$[\mathbf{z}_c^k]_i + \bar{\beta}^k [\mathbf{d}_c^k]_i \geq \varepsilon^k + \bar{\beta}^k [\mathbf{d}_c^k]_i \geq \varepsilon^k + \frac{\varepsilon^k}{\|\mathbf{d}_c^k\|_\infty} [\mathbf{d}_c^k]_i \geq 0$$

因此, 对任意的 $i = 1, 2, \dots, m$, 式(15)都成立, 所以式(14)成立。

引理 4 假设 $\{\mathbf{z}_c^k\}$ 是有界的, $\{\bar{\mathbf{H}}_c^k\}$ 一致正定有界, $\{\mathbf{d}_c^k\}$ 由算法 1 生成, ε^k 由式(3)定义, 则存在常数 $\bar{\beta} \in (0, 1)$, 使得对一切正整数 k , 都有 $\beta_c^k \geq \bar{\beta} \varepsilon^k$ 。

证明 由于 $\{\bar{\mathbf{H}}_c^k\}$ 一致正定有界, 则存在 $m_1, m_2 > 0$, 使得对一切正整数 k , 都有

$$\begin{cases} m_2 \left\| (\mathbf{N}_c^k)^T \mathbf{g}_c^k \right\|^2 \leq (\mathbf{g}_c^k)^T \mathbf{N}_c^k \bar{\mathbf{H}}_c^k (\mathbf{N}_c^k)^T \mathbf{g}_c^k \\ \left\| \bar{\mathbf{H}}_c^k \right\| \leq m_1 \end{cases} \quad (16)$$

因为 $\nabla \varphi$ Lipschitz 连续且 $\{\mathbf{z}_c^k\}$ 有界, 所以 $\{\mathbf{g}_c^k\}$ 有界, 即存在 $m_3 > 0$, 使得对一切正整数 k , 都有 $\|\mathbf{g}_c^k\| \leq m_3$. 由式(9)可知

$$\begin{aligned} \|\mathbf{d}_c^k\|^2 &= \left\| \bar{\mathbf{H}}_c^k (\mathbf{N}_c^k)^T \mathbf{g}_c^k \right\|^2 + \left\| (\mathbf{P}_1^k)^T \mathbf{T}_c^k \mathbf{g}_c^k \right\|^2 + \left\| (\mathbf{P}_2^k)^T \mathbf{g}_c^k \right\|^2 \leq \\ &\left\| \bar{\mathbf{H}}_c^k \right\|^2 \|\mathbf{g}_c^k\|^2 + \|\mathbf{T}_c^k\|^2 \|\mathbf{g}_c^k\|^2 + \|\mathbf{g}_c^k\|^2 \leq \\ &(m_1^2 + 2)m_3^2 \end{aligned} \quad (17)$$

因此, 由式(14)及式(17)可知, 存在 $\bar{\beta} \in (0, 1)$, 使得对一切正整数 k , 都有 $\beta_c^k \geq \bar{\beta} \varepsilon^k$ 。

引理 5 假设 $\{\bar{\mathbf{H}}_c^k\}$ 一致正定有界, $\{\mathbf{d}_c^k\}$ 由算法 1 产生, 则对一切正整数 k , 都有

$$\|\mathbf{d}_c^k\|^2 \leq -m_3 (\mathbf{g}_c^k)^T \mathbf{d}_c^k \quad (18)$$

其中, m_1, m_2 来自式(16)。

证明 由式(9)可知:

$$\begin{aligned} (\mathbf{g}_c^k)^T \mathbf{d}_c^k &\leq - \left(\left[(\mathbf{g}_c^k)^T \right]_i \left[\mathbf{P}_1^k (\mathbf{P}_1^k)^T \mathbf{T}_c^k \mathbf{g}_c^k \right]_i + \right. \\ &\left. \left\| (\mathbf{P}_2^k)^T \mathbf{g}_c^k \right\|^2 + m_2 \left\| (\mathbf{N}_c^k)^T \mathbf{g}_c^k \right\|^2 \right) \end{aligned}$$

由 $\mathbf{o}_i^k \in [0, 1]$, 以及式(16)和式(17)可以得到

$$\|\mathbf{d}_c^k\|^2 \leq -\frac{m_1^2}{m_2} (\mathbf{g}_c^k)^T \mathbf{d}_c^k$$

即式(18)成立。

引理 6 设 $\mathbf{c}^k$ 和 $\mathbf{d}_c^k$ 由算法 1 产生, 则对任意 k 有 $\varphi(\mathbf{z}^k) \leq \mathbf{c}^k$ (19)

证明参见文献[11]中引理 1。

定理 2 假设 $\{\mathbf{z}_c^k\}$ 是有界的, $\{\bar{\mathbf{H}}_c^k\}$ 一致正定有界, $\{\mathbf{d}_c^k\}$ 由算法 1 产生, 则 $\{\mathbf{z}_c^k\}$ 的极限点都是问题(2)的 KKT 点。

证明 分两步证明该结论, 第一步, 先证明 α_c^k 是有下界的。

假定 α_c^k 满足式 (11), 如果 $\sigma\alpha_c^k \geq \bar{\beta}\varepsilon^k$, 则 $\alpha_c^k \geq \frac{\bar{\beta}\varepsilon^k}{\sigma}$, 否则, $\alpha_c^k < \frac{\bar{\beta}\varepsilon^k}{\sigma}$.

由式(19), $\varphi(z^k) \leq c^k$, 有

$$\begin{aligned} \varphi(z^{k+1}) &= \varphi(P_\Omega[z_c^k + \sigma\alpha_c^k d_c^k]) > c^k + \sigma\alpha_c^k \delta(g_c^k)^T d_c^k \geq \\ &\varphi(z^k) + \sigma\alpha_c^k \delta(g_c^k)^T d_c^k \end{aligned} \quad (20)$$

由于 $\nabla\varphi$ 满足 Lipschitz 连续, 则

$$\varphi(P_\Omega[z_c^k + \sigma\alpha_c^k d_c^k]) - \varphi(z_c^k) = \varphi(z_c^k + \sigma\alpha_c^k d_c^k) - \varphi(z_c^k) =$$

$$\sigma\alpha_c^k \delta(g_c^k)^T d_c^k + \int_0^{\sigma\alpha_c^k} [\nabla\varphi(z_c^k + td_c^k) - \nabla\varphi(z_c^k)]^T d_c^k dt \leq$$

$$\sigma\alpha_c^k \delta(g_c^k)^T d_c^k + \frac{1}{2} L \sigma^2 (\alpha_c^k)^2 \|d_c^k\|^2$$

由式(20)可得

$$\begin{aligned} \sigma\alpha_c^k \delta(g_c^k)^T d_c^k &\leq \varphi(z^{k+1}) - \varphi(z^k) \leq \\ \varphi(z^{k+1}) - \varphi(z_c^k) &\leq \sigma\alpha_c^k \delta(g_c^k)^T d_c^k + \\ \frac{1}{2} L \sigma^2 (\alpha_c^k)^2 \|d_c^k\|^2 \end{aligned} \quad (21)$$

由引理 5 和式(21)可得

$$\alpha_c^k \geq \frac{-2(1-\delta)(g_c^k)^T d_c^k}{\sigma L \|d_c^k\|^2} \geq \frac{2(1-\delta)}{\sigma L m_4}$$

因此

$$\alpha_c^k \geq \min\left\{\frac{\bar{\beta}\varepsilon^k}{\sigma}, \frac{2(1-\delta)}{\sigma L m_4}\right\} \quad (22)$$

式中, $m_4 = \frac{m_1^2}{m_2}$.

第二步, 证明 $\lim_{k \rightarrow \infty} (g_c^k)^T d_c^k = 0$ 。

由式(11)和式(12)可得

$$c^{k+1} = \frac{\tau Q_k c^k + \varphi(z^{k+1})}{Q_{k+1}} \leq c^k + \frac{\alpha_c^k \delta(g_c^k)^T d_c^k}{Q_{k+1}} \quad (23)$$

由于 φ 有下界, 由引理 6 知 $\varphi(z^k) \leq c^k$, 对任意的正整数 k 都成立, 结合引理 2 可知 $\{c^k\}$ 单调减小且有下界。由式(23)可得

$$\sum_{k=0}^{\infty} \frac{\alpha_c^k \delta(g_c^k)^T d_c^k}{Q_{k+1}} < \infty \quad (24)$$

假设 z^* 是 $\{z_c^k\}$ 的任一聚点。于是, 存在无限指标集 $\mathcal{K}$ 使得 $\lim_{k \rightarrow \infty} z_c^k = z^*$ 。不妨设 $\lim_{k \rightarrow \infty} w^k = \bar{w}$, 若 $\bar{w} = 0$, 则 z^* 为问题(2)的 KKT 点, 否则 $\bar{w} > 0$ 。由式(22)可知, $\{\alpha_c^k\}_{\mathcal{K}}$ 下界不会趋于零, 结合式(24)得

$\lim_{k \in \mathcal{K}, k \rightarrow \infty} \frac{(g_c^k)^T d_c^k}{Q_{k+1}} = 0$ 。又因为 $0 < \tau < 1$, 则有

$$Q_{k+1} = 1 + \sum_{i=1}^k \prod_{m=0}^i \tau \leq 1 + \sum_{j=0}^k \tau^{j+1} \leq \sum_{j=0}^{\infty} \tau^j = \frac{1}{1-\tau}$$

所以 $1 < Q_{k+1} \leq \frac{1}{1-\tau}$, 即 Q_{k+1} 不会趋于零, 因

此, $\lim_{k \in \mathcal{K}, k \rightarrow \infty} (g_c^k)^T d_c^k = 0$ 。

由定理 2 知, z^* 为问题(2)的 KKT 点。综上, $\{z_c^k\}$ 的极限点均为问题(2)的 KKT 点。

3 数值结果

测试如下形式优化问题:

$$\begin{aligned} \min \quad & \frac{1}{2} \|X - G\|_F^2 \\ \text{s.t.} \quad & \begin{cases} X_{ij} = e_{ij}, & (i, j) \in \mathcal{B}_e \\ X_{ij} \geq l_{ij}, & (i, j) \in \mathcal{B}_l \\ X_{ij} \leq u_{ij}, & (i, j) \in \mathcal{B}_u \\ X \succeq 0 \end{cases} \end{aligned}$$

式中, $\mathcal{B}_e, \mathcal{B}_l, \mathcal{B}_u$ 为 $\{(i, j) | 1 \leq i \leq j \leq n\}$ 的 3 个指标子集, 满足: $\mathcal{B}_e \cap \mathcal{B}_l = \emptyset, \mathcal{B}_e \cap \mathcal{B}_u = \emptyset, l_{ij} \leq u_{ij}, \forall (i, j) \in \mathcal{B}_l \cap \mathcal{B}_u$ 。用 p_e, q_l, q_u 分别指代 $\mathcal{B}_e, \mathcal{B}_l, \mathcal{B}_u$ 的基数, 所以总约束个数 $m = p_e + q_l + q_u$ 。

对于上述类型测试问题, 通过例 1、例 2 分别进行了数值实验, 并将算法 1 与文献[4]中光滑牛顿法(用 ISNM 表示)从迭代次数和 CPU 时间两个方面进行比较。采用 Matlab 语言 (Matlab 8.6.0) 编程实现算法 1, 并在个人电脑(配置为 CPU 2.30 GHZ, RAM 8.00 GB)上运行测试问题。算法 1 的初始值设置为: 给定初始点 $z^0 = 0, \sigma = 0.2, \tau = 0.85, \delta = 10^{-4}$, 初始矩阵 $H^0 = I, \varepsilon = 10^{-5}$ 。对 ISNM 算法, 采用默认设置。

测试问题(参见文献[4])

例 1 数据矩阵 C 来自文献[4], 它是一个随机生成的 $n \times n$ 对称矩阵, 其中 C_{ij} 按照 $[-1, 1]$ 上均匀分布随机生成, 且令 $C_{ii} = 1, i, j = 1, \dots, n$ 。生成数据矩阵 C , 细节由如下的 Matlab 代码具体给出:

```
C=2.0*rand(n, n)-ones(n, n); C=triu(C)+triu(C, 1)'; C=C*C'; for i=1:n; C(i, i)=1; end.
```

指标集

$$\mathcal{B}_e = \{(i, i) | i = 1, \dots, n\}$$

$$\mathcal{B}_l = \mathcal{B}_u = \{(i, i+j) | i = 1, \dots, n-j, j = 1, 2, \dots, n_z\}$$

式中, n_z 是 X 中每行设定的上(下)界约束的个数。当 $(i, i) \in \mathcal{B}_e$ 时, $e_{ii} = 1$; 当 $(i, j) \in \mathcal{B}_l$ 时, $l_{ij} = -0.1$; 当 $(i, j) \in \mathcal{B}_u$ 时, $u_{ij} = 0.1$ 。分别测试 $n = 1\ 000, 1\ 500$ 和 $2\ 000$ 时的实验结果。

例 1 的数值实验结果见表 1，从表 1 可以看出，当矩阵维数固定时，增加约束个数，两种方法迭代次数都有所增加，但算法 1 迭代次数增加较慢，CPU 时间比 ISNM 算法明显少很多。当约束条件固定时，矩阵维数增加，两种方法迭代次数都有所增加，但算法 1 算法迭代次数增加较慢，而且 CPU 时间比 ISNM 算法仍然少很多。这样的结果符合预期，因为算法 1 只需要储存有限个修正对，而 ISNM 需要求解牛顿方程，因此，算法 1 在 CPU 时间上要比 ISNM 好一些。另外，ISNM 算法默认设置的最大迭代次数是 500，表 1 给出了 ISNM 在迭代 500 次时所需 CPU 时间，对于超过最大迭代次数的情况在表中用*进行标识。算法 1 成功地解决了例 1 的所有情况，而 ISNM 由于达到最大迭代次数 500 而未能解决例 1 的 3 个实例（标记为*），且 ISNM 的 CPU 时间已经远超过算法 1 所需时间。

表 1 例 1 的数值实验结果

Tab.1 Numerical test results of example 1

<i>n</i>	<i>m</i>	<i>n_z</i>	ISNM		算法 1	
			迭代次数	CPU 时间/s	迭代次数	CPU 时间/s
1 000	95 950	100	106	206	137	65
1 000	255 850	300	236	1 668	264	191
1 000	375 750	500	362	4 455	355	321
1 500	146 450	100	165	787	138	189
1 500	406 350	300	455	5 834	258	453
1 500	626 250	500	500*	10 901	372	786
2 000	196 950	100	215	2 104	137	448
2 000	556 850	300	500*	11 633	258	982
2 000	876 750	500	500*	17 036	361	1 615

例 2 矩阵 C 及 e_{ii} 同例 1 中定义的一样，指标 $\mathcal{B}_e = \{(i, i) | i = 1, \dots, n\}$ ，指标集 $\mathcal{B}_l, \mathcal{B}_u \subset \{(i, j) | 1 \leq i < j \leq n\}$ ，由 X 的第 i 行随机生成的 $\min(n_z, n - i)$ 个元素组成。

分别测试 $n = 1\,000, 1\,500, 2\,000$ 的情形，数值实验结果如表 2 所示。从表 2 可以看出当矩阵维数固定时，增加约束个数，两种方法迭代次数都有所增加，但算法 1 迭代次数增加较慢，CPU 时间比 ISNM 算法明显少很多。可能的原因是，在 m 很大的情况下，ISNM 需要更多的计算工作来解决广义牛顿方程。当约束条件固定时，矩阵维数增加，两种方法迭代次数都有所增加，但算法 1

表 2 例 2 的数值实验结果

Tab.2 Numerical test results of example 2

<i>n</i>	<i>m</i>	<i>n_z</i>	ISNM		算法 1	
			迭代次数	CPU 时间/s	迭代次数	CPU 时间/s
1 000	95 950	100	112	282	158	77
1 000	255 850	300	301	2 516	289	209
1 000	375 750	500	376	4 777	350	313
1 500	146 450	100	167	970	167	230
1 500	406 350	300	433	6 231	317	563
1 500	626 250	500	500*	11 759	392	818
2 000	196 950	100	228	2 718	151	536
2 000	556 850	300	500*	13 096	300	1 129
2 000	876 750	500	500*	17 030	398	1 694

算法迭代次数增加较慢，而且 CPU 时间比 ISNM 算法仍然少很多，这个结果是和表 1 有点相似的。但表 2 的迭代次数略微比表 1 多一些，这和测试问题有关。例 1 中的约束条件位置呈现带状，而例 2 约束条件位置是随机生成的，这或许是迭代多一些的原因。

4 结 论

对于该最小二乘半正定规划问题，提出了一种基于柯西点信息的 L-BFGS 算法。首先将最小二乘半正定规划问题转化为对偶问题，然后构造相应二次模型，再沿着负梯度方向最小化该二次模型得到柯西点，并在此基础上，利用积极约束技巧，划分积极约束集与非积极约束集，然后再运用 L-BFGS 方法对自由变量进行加速。对于步长的选取，采用非单调搜索技巧确定步长，与单调或传统的非单调方案相比，非单调线搜索算法平均使用较少的函数和梯度计算次数。对于算法 1，在一定的条件下，从理论上证明了算法的全局收敛性。最后，进行了初步的数值实验，数值实验表明所提算法 1 在处理较多约束优化问题时更为有效，与 ISNM 算法相比，在 CPU 时间上有一定的优势。

参考文献:

[1] HIGHAM N J. Computing the nearest correlation matrix-a problem from finance[J]. *IMA Journal of Numerical Analysis*, 2002, 22(3): 329–343.