

求解大型线性最小二乘问题的贪婪 Gauss-Seidel 方法

李寒宇, 张彦钧

(重庆大学 数学与统计学院, 重庆 401331)

摘要: 基于一种选择系数矩阵 A 的工作列的策略, 提出了求解大型线性最小二乘问题的一种不同的贪婪 Gauss-Seidel 方法, 并对该方法进行了收敛性分析。数值实验表明, 在相同的精度下, 所提方法在计算时间上优于文献提出的贪婪随机坐标下降方法。

关键词: 贪婪 Gauss-Seidel 方法; 贪婪随机坐标下降方法; 随机的 Gauss-Seidel 方法; 大型线性最小二乘问题

中图分类号: O241.6

文献标志码: A

A Greedy Gauss-Seidel Method for Solving the Large Linear Least Squares Problem

LI Hanyu, ZHANG Yanjun

(College of Mathematics and Statistics, Chongqing University, Chongqing 401331, China)

Abstract: An alternative greedy Gauss-Seidel method for solving the large linear least squares problem is proposed based on a novel strategy for choosing the working columns of the coefficient matrix A , and the convergence of the new method is analyzed. The numerical experiments show that for the same accuracy, the method proposed in this paper outperforms the greedy randomized coordinate descent method proposed recently in term of the computing time.

Key words: greedy Gauss-Seidel method; greedy randomized coordinate descent method; randomized Gauss-Seidel method; large linear least squares problem

线性最小二乘问题是数值代数与科学计算中一个经典问题, 经常出现于参数估计、反演与预测等问

题之中。常见的直接法包括带旋转的 QR 分解法和奇异值分解法^[1-2]等。然而, 当矩阵规模较大时, 此类方法通常需要很大的存储量并且计算成本较高。因此, 迭代法成为求解大型线性最小二乘问题的重要选择, 如著名的 Gauss-Seidel 方法^[3]。

众所周知, Gauss-Seidel 方法与 Kaczmarz 方法密切相关, 如它们均可视为坐标下降法对应于特殊半正定线性系统的变形。具体地, 对于线性系统 $Ax = b$, Kaczmarz 方法的迭代格式与坐标下降法应用于半正定系统 $AA^T u = b$, 并结合标准的原始-对偶映射 $x = A^T u$ 的迭代格式一致, Gauss-Seidel 方法的迭代格式则等同于坐标下降法应用于半正定系统 $A^T Ax = A^T b$ 的迭代格式^[4]。2009年, Strohmer 和 Vershynin^[5]首次证明了随机 Kaczmarz 方法具有线性收敛性。随后, Leventhal 和 Lewis^[6]证明了随机 Gauss-Seidel (RGS) 方法也具有类似的结果, 该方法也称为随机坐标下降方法。它是根据适当的概率分布随机投影于矩阵 A 的列, 达到最小化 $\|b - Ax\|_2^2$ 的目的, 因而具有很好的性能, 从而引起了广泛的关注, 可参见文献[7-14]及其参考文献。

Bai 和 Wu^[15]提出了贪婪随机坐标下降 (GRCD) 方法, 该方法引入了一种贪婪的概率准则, 避免了 RGS 方法的一些缺点, 从而使得其在迭代次数和计算时间上都优于 RGS 方法。文献[15]提出的贪婪思想具有广泛的应用, 参见文献[16-24]及其参考文献。

本文提出一种求解大型线性最小二乘问题的贪婪 Gauss-Seidel (GGS) 方法, 与 GRCD 方法相比, 它采用了完全不同的方式来确定矩阵 A 的工作列, 使得其每次迭代所需的计算时间更少。从理论上证明

收稿日期: 2021-05-10

基金项目: 国家自然科学基金(11671060); 重庆市自然科学基金(cstc2019jcyj-msxmX0267)

第一作者: 李寒宇(1981—), 男, 教授, 博士生导师, 理学博士, 主要研究方向为随机数值代数与张量计算。

E-mail: hlyl@cqu.edu.cn



论文
拓展
介绍

GGs 方法的收敛性;在数值实验中,利用文献[15]中的例子比较了 GGS 和 GRCD 这 2 种方法的性能。

1 预备知识

对于向量 $z \in \mathbb{R}^n$, $z^{(j)}$ 表示它的第 j 个分量。对于矩阵 $G = (g_{ij}) \in \mathbb{R}^{m \times n}$, $G_{(j)}$, $\|G\|_2$ 以及 $\|G\|_F$ 分别表示它的第 j 列、谱范数以及 F 范数。此外,如果 $G \in \mathbb{R}^{n \times n}$ 是一个正定矩阵,那么任意一个向量 $x \in \mathbb{R}^n$ 的能量范数定义为 $\|x\|_G = \sqrt{x^T G x}$, 其中 $(\cdot)^T$ 表示一个向量或者一个矩阵的转置。另外,用 I 表示单位矩阵,用 e_j 表示它的第 j 列,用 $\lambda_{\min}(G^T G)$ 表示 $G^T G$ 的最小正特征值,用 $|W|$ 表示集合 W 的元素个数。

类似文献[15],本文以 $x^* = A^\dagger b$, 其中 $A^\dagger = (A^T A)^{-1} A^T$ 表示 A 的 Moore-Penrose 广义逆,表示如下线性最小二乘问题的唯一最小范数最小二乘解:

$$\min_{x \in \mathbb{R}^n} \|b - Ax\|_2^2 \quad (1)$$

其中矩阵 $A \in \mathbb{R}^{m \times n}$ 列满秩,向量 $b \in \mathbb{R}^m$. 众所周知, $x^* := \arg \min_{x \in \mathbb{R}^n} \|b - Ax\|_2^2$ 是关于式(1)的如下正规方程组^[25]的解:

$$A^T A x = A^T b \quad (2)$$

基于正规方程组(2),文献[15]提出了如下贪婪随机坐标下降方法,即算法 1,其中 $r_k = b - Ax_k$ 表示残差向量。

算法 1 贪婪随机坐标下降方法。①置 $k := 0$ 。计算 $\delta_k = \frac{1}{2} \left(\frac{1}{\|A^T r_k\|_2^2} \max_{1 \leq j \leq n} \left\{ \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2} \right\} + \frac{1}{\|A\|_F^2} \right)$ 。②定义正整数指标集 $v_k = \{j \mid |A_{(j)}^T r_k|^2 \geq \delta_k \|A^T r_k\|_2^2 \|A_{(j)}\|_2^2\}$ 。③令 $s_k = A^T r_k$, 计算 $\tilde{s}_k^{(j)} = \begin{cases} s_k^{(j)}, & \text{如果 } j \in v_k, \\ 0, & \text{其他情形。} \end{cases}$ ④根据概率 $\Pr(c = j_k) = \frac{|\tilde{s}_k^{(j_k)}|^2}{\|\tilde{s}_k\|_2^2}$ 选取指标 $j_k \in v_k$ 。⑤计算 $x_{k+1} = x_k + \frac{s_k^{(j_k)}}{\|A_{(j_k)}\|_2^2} e_{j_k}$ 。⑥置 $k = k + 1$, 转步骤①。

由贪婪随机坐标下降方法中 δ_k 和 v_k 的定义可知,如果 $l \in v_k$, 那么

$$\frac{|A_{(l)}^T r_k|^2}{\|A_{(l)}\|_2^2} \geq \frac{1}{2} \left(\max_{1 \leq j \leq n} \left\{ \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2} \right\} + \frac{\|A^T r_k\|_2^2}{\|A\|_F^2} \right)$$

注意到

$$\max_{1 \leq j \leq n} \left\{ \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2} \right\} \geq \sum_{j=1}^n \frac{\|A_{(j)}\|_2^2 |A_{(j)}^T r_k|^2}{\|A\|_F^2 \|A_{(j)}\|_2^2} = \frac{\|A^T r_k\|_2^2}{\|A\|_F^2}$$

因此,不能得到如下结论:如果 $l \in v_k$, 那么

$$\frac{|A_{(l)}^T r_k|^2}{\|A_{(l)}\|_2^2} \geq \max_{1 \leq j \leq n} \left\{ \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2} \right\}, \text{ 也即}$$

$$\frac{|A_{(l)}^T r_k|^2}{\|A_{(l)}\|_2^2} = \max_{1 \leq j \leq n} \left\{ \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2} \right\}$$

由此可知,可能存在一些 $l \in v_k$ 使得

$$\frac{|A_{(l)}^T r_k|^2}{\|A_{(l)}\|_2^2} < \max_{1 \leq j \leq n} \left\{ \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2} \right\} \quad (3)$$

与此同时,对于任意的 $j_k \in v_k$, 由迭代格式可以得到

$$\|Ax_{k+1} - Ax_k\|_2^2 = \frac{|A_{(j_k)}^T r_k|^2}{\|A_{(j_k)}\|_2^2} \quad (4)$$

结合式(3)和式(4)可以发现,在更新得到 x_{k+1} 的时候,指标集合 v_k 中的列指标并不能保证 Ax_{k+1} 与 Ax_k 之间的距离最大。此外,在计算 δ_k 的时候需要计算矩阵 A 的所有列范数。

2 基于残差-距离的贪婪 Gauss-Seidel 方法

考虑到贪婪随机坐标下降方法里指标集合 v_k 中的列指标并不能保证 Ax_{k+1} 与 Ax_k 之间的距离最大,并且计算 δ_k 的时候需要计算矩阵 A 的所有列范数;同时,考虑到最近很多工作按照最大残差^[26-30]策略选择迭代指标,提出了基于残差-距离的贪婪 Gauss-Seidel 方法,即算法 2。

算法 2 贪婪 Gauss-Seidel 方法。①置 $k := 0$ 。确定正整数指标集

$$R_k = \{ \tilde{j}_k \mid \tilde{j}_k = \arg \max_{1 \leq j \leq n} |A_{(j)}^T r_k| \}$$

②计算

$$j_k = \arg \max_{\tilde{j}_k \in R_k} \left\{ \frac{|A_{(\tilde{j}_k)}^T r_k|^2}{\|A_{(\tilde{j}_k)}\|_2^2} \right\}$$

③计算

$$x_{k+1} = x_k + \frac{A_{(j_k)}^T r_k}{\|A_{(j_k)}\|_2^2} e_{j_k}$$

④置 $k = k + 1$, 转步骤①。

贪婪 Gauss-Seidel 方法主要包括 2 步: ①通过正规方程组(2)的残差向量 s_k 的最大元素确定指标集合 R_k 。②按照 Ax_{k+1} 与 Ax_k 之间的距离最大准则从集合 R_k 中选择迭代所需列指标。粗略看来, 该方法似乎改变了贪婪随机坐标下降方法的 2 个主要步骤的顺序。然而, 与贪婪随机坐标下降方法相比, 贪婪 Gauss-Seidel 方法除了在求 x_{k+1} 时使得 Ax_{k+1} 与 Ax_k 之间的距离最大之外, 也不再需要计算矩阵 A 每一列的范数。此外, 由于 R_k 是由向量 s_k 的最大项决定的, 还可以发现集合 R_k 中的元素个数可能小于集合 v_k 中的元素个数, 即 $|R_k| < |v_k|$ 。因此, 该方法可以减少每次迭代的计算成本, 从而可期待在计算时间上会有更好的表现, 这一点将在第 3 节中的数值实验中进行验证。

注 1 如果 $|A_{(j_k)}^T r_k| = \max_{1 \leq j \leq n} |A_{(j)}^T r_k|$, 则 $j_k \in R_k$ 。

由此可知贪婪 Gauss-Seidel 方法中的集合 R_k 非空。

注 2 类似于贪婪随机坐标下降方法, 贪婪

Gauss-Seidel 方法可利用 $\frac{|A_{(\tilde{j}_k)}^T r_k|^2}{\|A_{(\tilde{j}_k)}\|_2^2}$, 其中 $\tilde{j}_k \in R_k$, 作

为概率选择标准得到相应的随机算法。在这种情况下, 它的收敛因子可能比贪婪 Gauss-Seidel 方法稍

差, 因为贪婪 Gauss-Seidel 方法是根据 $\frac{|A_{(\tilde{j}_k)}^T r_k|^2}{\|A_{(\tilde{j}_k)}\|_2^2}$ 最

大的值来选择指标的, 其使得 Ax_{k+1} 与 Ax_k 之间的距离最大。

注 3 在贪婪 Gauss-Seidel 方法中确定了指标集合 R_k 之后, 类似文献[31-34]中的思想, 可以得到关于贪婪 Gauss-Seidel 方法的分块算法。

接下来, 给出贪婪 Gauss-Seidel 方法的收敛性分析。

定理 1 由贪婪 Gauss-Seidel 方法生成的迭代

序列 $\{x_k\}_{k=0}^{\infty}$, 从初始向量 $x_0 \in \mathbf{R}^n$ 开始, 线性收敛于唯一的最小范数最小二乘解 $x_* = A^\dagger b$, 并且

$$\|x_1 - x_*\|_{A^T A}^2 \leq \left(1 - \frac{1}{|R_0|} \cdot \sum_{j_0 \in R_0} \frac{1}{\|A_{(j_0)}\|_2^2} \cdot \frac{1}{n} \cdot \lambda_{\min}(A^T A)\right) \|x_0 - x_*\|_{A^T A}^2 \quad (5)$$

对于 $k=1, 2, \dots$, 有

$$\|x_{k+1} - x_*\|_{A^T A}^2 \leq \left(1 - \frac{1}{|R_k|} \cdot \sum_{j_k \in R_k} \frac{1}{\|A_{(j_k)}\|_2^2} \cdot \frac{1}{n-1} \cdot \lambda_{\min}(A^T A)\right) \|x_k - x_*\|_{A^T A}^2 \quad (6)$$

此外, 令 $\beta = \min\left\{\frac{1}{|R_k|} \cdot \sum_{j_k \in R_k} \frac{1}{\|A_{(j_k)}\|_2^2}\right\}$, $k=1, 2, \dots$, 则对于 $k=1, 2, \dots$, 有

$$\|x_k - x_*\|_{A^T A}^2 \leq \left(1 - \frac{\beta \cdot \lambda_{\min}(A^T A)}{n-1}\right)^{k-1} \sum_{j_0 \in R_0} \frac{\lambda_{\min}(A^T A)}{|R_0| \cdot \|A_{(j_0)}\|_2^2 \cdot n} \|x_0 - x_*\|_{A^T A}^2 \quad (7)$$

证明 由贪婪 Gauss-Seidel 方法的迭代格式可知

$$A(x_{k+1} - x_k) = \frac{A_{(j_k)}^T r_k}{\|A_{(j_k)}\|_2^2} A_{(j_k)}$$

这意味着 $A(x_{k+1} - x_k)$ 平行于 $A_{(j_k)}$ 。与此同时

$$A(x_{k+1} - x_*) = A\left(x_k - x_* + \frac{A_{(j_k)}^T r_k}{\|A_{(j_k)}\|_2^2} e_{j_k}\right) =$$

$$A(x_k - x_*) + \frac{A_{(j_k)}^T r_k}{\|A_{(j_k)}\|_2^2} A_{(j_k)}$$

结合 $A^T A x_* = A^T b$ 可以得到

$$A(x_{k+1} - x_*) = \left(I - \frac{A_{(j_k)} A_{(j_k)}^T}{\|A_{(j_k)}\|_2^2}\right) A(x_k - x_*)$$

则

$$A_{(j_k)}^T A(x_{k+1} - x_*) = A_{(j_k)}^T \left(I - \frac{A_{(j_k)} A_{(j_k)}^T}{\|A_{(j_k)}\|_2^2}\right) A(x_k - x_*) = 0$$

故 $A(x_{k+1} - x_*)$ 垂直于 $A_{(j_k)}$ 。因此, 向量 $A(x_{k+1} - x_k)$ 垂直于向量 $A_{(j_k)}$ 。由勾股定理可得

$$\|A(x_{k+1} - x_*)\|_2^2 = \|A(x_k - x_*)\|_2^2 - \|A(x_{k+1} - x_k)\|_2^2$$

其等价形式为

$$\|x_{k+1} - x_*\|_{A^T A}^2 = \|x_k - x_*\|_{A^T A}^2 - \|x_{k+1} - x_k\|_{A^T A}^2 \quad (8)$$

另一方面, 由贪婪 Gauss-Seidel 方法可知

$$|A_{(j_k)}^T r_k| = \max_{1 \leq j \leq n} |A_{(j)}^T r_k|$$

并且

$$\frac{|A_{(j_k)}^T r_k|^2}{\|A_{(j_k)}\|_2^2} = \max_{j \in R_k} \left\{ \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2} \right\}$$

则

$$\begin{aligned} \|x_{k+1} - x_k\|_{A^T A}^2 &= \|A(x_{k+1} - x_k)\|_2^2 = \\ &= \frac{|A_{(j_k)}^T r_k|^2}{\|A_{(j_k)}\|_2^2} \geq \sum_{j_k \in R_k} \frac{|A_{(j_k)}^T r_k|^2}{\|A_{(j_k)}\|_2^2} \geq \\ &\geq \frac{|A_{(j_k)}^T r_k|^2}{\sum_{j \in R_k} \frac{|A_{(j)}^T r_k|^2}{\|A_{(j)}\|_2^2}} \geq \\ &\sum_{j_k \in R_k} \frac{1}{|R_k|} \cdot \frac{|A_{(j_k)}^T r_k|^2}{\|A_{(j_k)}\|_2^2} = \sum_{j_k \in R_k} \frac{1}{|R_k|} \cdot \frac{\max_{1 \leq j \leq n} |A_{(j)}^T r_k|^2}{\|A_{(j_k)}\|_2^2} \end{aligned} \quad (9)$$

因此, 将式(9)代入式(8)可得

$$\|x_{k+1} - x_*\|_{A^T A}^2 \leq \|x_k - x_*\|_{A^T A}^2 - \sum_{j_k \in R_k} \frac{1}{|R_k|} \cdot \frac{\max_{1 \leq j \leq n} |A_{(j)}^T r_k|^2}{\|A_{(j_k)}\|_2^2} \quad (10)$$

当 $k=0$ 时, 有

$$\begin{aligned} \max_{1 \leq j \leq n} |A_{(j)}^T r_0|^2 &= \max_{1 \leq j \leq n} |A_{(j)}^T r_0|^2 \cdot \frac{\|A^T r_0\|_2^2}{\sum_{j=1}^n |A_{(j)}^T r_0|^2} \geq \\ &\frac{1}{n} \cdot \|A^T r_0\|_2^2 \end{aligned}$$

结合文献[16]中的结果:

$$\|A^T x\|_2^2 \geq \lambda_{\min}(A^T A) \|x\|_2^2 \quad (11)$$

其中向量 x 属于 A 的列空间, 可得到

$$\begin{aligned} \max_{1 \leq j \leq n} |A_{(j)}^T r_0|^2 &\geq \frac{1}{n} \cdot \lambda_{\min}(A^T A) \cdot \|Ax_* - Ax_0\|_2^2 = \\ &\frac{1}{n} \cdot \lambda_{\min}(A^T A) \cdot \|x_0 - x_*\|_{A^T A}^2 \end{aligned} \quad (12)$$

因此, 将式(12)代入式(10)可得

$$\begin{aligned} \|x_1 - x_*\|_{A^T A}^2 &\leq \|x_0 - x_*\|_{A^T A}^2 - \\ &\sum_{j_0 \in R_0} \frac{1}{|R_0|} \cdot \frac{1}{\|A_{(j_0)}\|_2^2} \cdot \frac{1}{n} \cdot \lambda_{\min}(A^T A) \cdot \|x_0 - x_*\|_{A^T A}^2 \\ &= \left(1 - \frac{1}{|R_0|} \cdot \sum_{j_0 \in R_0} \frac{1}{\|A_{(j_0)}\|_2^2} \cdot \frac{1}{n} \cdot \lambda_{\min}(A^T A)\right) \|x_0 - x_*\|_{A^T A}^2 \end{aligned}$$

这即是式(5)。

当 $k \geq 1$ 时, 有

$$\max_{1 \leq j \leq n} |A_{(j)}^T r_k|^2 = \max_{1 \leq j \leq n} |A_{(j)}^T r_k|^2 \cdot \frac{\|A^T r_k\|_2^2}{\sum_{j=1}^n |A_{(j)}^T r_k|^2}$$

根据贪婪 Gauss-Seidel 方法的迭代格式, 立刻可得

$$\begin{aligned} A_{(j_{k-1})}^T r_k &= A_{(j_{k-1})}^T (r_{k-1} - \frac{A_{(j_{k-1})}^T r_{k-1}}{\|A_{(j_{k-1})}\|_2^2} A_{(j_{k-1})}) = \\ &A_{(j_{k-1})}^T r_{k-1} - A_{(j_{k-1})}^T r_{k-1} = 0 \end{aligned}$$

则

$$\max_{1 \leq j \leq n} |A_{(j)}^T r_k|^2 = \max_{1 \leq j \leq n} |A_{(j)}^T r_k|^2 \cdot \frac{\|A^T r_k\|_2^2}{\sum_{\substack{j=1 \\ j \neq j_{k-1}}}^n |A_{(j)}^T r_k|^2} \geq$$

$$\frac{1}{n-1} \cdot \|A^T r_k\|_2^2$$

结合式(11)得到

$$\begin{aligned} \max_{1 \leq j \leq n} |A_{(j)}^T r_k|^2 &\geq \\ &\frac{1}{n-1} \cdot \lambda_{\min}(A^T A) \|Ax_* - Ax_k\|_2^2 \\ &= \frac{1}{n-1} \cdot \lambda_{\min}(A^T A) \|x_k - x_*\|_{A^T A}^2 \end{aligned} \quad (13)$$

因此, 将式(13)代入式(10)可得

$$\begin{aligned} \|x_{k+1} - x_*\|_{A^T A}^2 &\leq \|x_k - x_*\|_{A^T A}^2 - \\ &\sum_{j_k \in R_k} \frac{1}{|R_k|} \cdot \frac{1}{\|A_{(j_k)}\|_2^2} \cdot \frac{1}{n-1} \cdot \lambda_{\min}(A^T A) \|x_k - x_*\|_{A^T A}^2 \\ &= \left(1 - \frac{1}{|R_k|} \cdot \sum_{j_k \in R_k} \frac{1}{\|A_{(j_k)}\|_2^2} \cdot \frac{1}{n-1} \cdot \lambda_{\min}(A^T A)\right) \|x_k - x_*\|_{A^T A}^2 \end{aligned}$$

这即是式(6)。关于迭代指标 k 递归可得到式(7)。

注 4 运用下面不等式

$$1 - \frac{\lambda_{\min}(A^T A)}{\|A\|_F^2 - \min_{1 \leq j \leq n} \|A_{(j)}\|_2^2} < 1 - \frac{\lambda_{\min}(A^T A)}{\|A\|_F^2} < 1$$

可以得到

$$0 < \frac{\lambda_{\min}(A^T A)}{\max_{1 \leq j \leq n} \|A_{(j)}\|_2^2 \cdot (n-1)} \leq \frac{\lambda_{\min}(A^T A)}{\|A\|_F^2 - \min_{1 \leq j \leq n} \|A_{(j)}\|_2^2} < 1$$

再结合

$$\frac{1}{\max_{1 \leq j \leq n} \|A_{(j)}\|_2^2} \leq \beta \leq \frac{1}{n} \cdot \sum_{j_k=1}^n \frac{1}{\|A_{(j_k)}\|_2^2}$$

可得到

$$1 - \frac{\beta \cdot \lambda_{\min}(A^T A)}{n-1} \leq 1 - \frac{\lambda_{\min}(A^T A)}{\max_{1 \leq j \leq n} \|A_{(j)}\|_2^2 \cdot (n-1)} < 1$$

因此,贪婪 Gauss-Seidel 方法的收敛因子确实是小于1的。

注5 贪婪随机坐标下降方法^[15]的误差估计如下:

$$E_k[\|x_{k+1} - x_*\|_{A^T A}^2] \leq \left(1 - \frac{1}{2} \left(\frac{1}{\|A\|_F^2 - \min_{1 \leq j \leq n} \|A_{(j)}\|_2^2} + \frac{1}{\|A\|_F^2} \lambda_{\min}(A^T A) \right) \|x_k - x_*\|_{A^T A}^2 \right)$$

其中 $k=1, 2, \dots$ 。由此可知贪婪随机坐标下降方法的收敛因子为

$$1 - \frac{1}{2} \left(\frac{1}{\|A\|_F^2 - \min_{1 \leq j \leq n} \|A_{(j)}\|_2^2} + \frac{1}{\|A\|_F^2} \lambda_{\min}(A^T A) \right)$$

由于直接比较贪婪 Gauss-Seidel(GGS)和贪婪随机坐标下降(GRCD)2种方法的收敛因子并不容易,所以采用2个数值例子进行比较。此外,还绘制了2种方法的实际收敛速率,其定义^[17]如下:

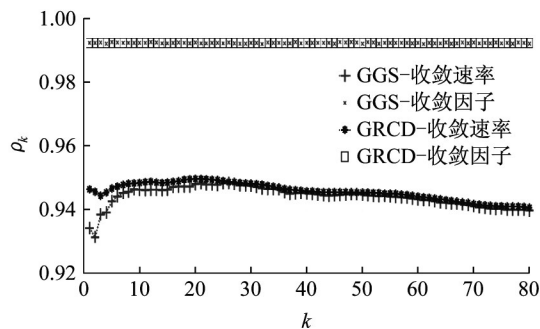
$$\rho_k = \left(\frac{E[\|x_k - x_*\|_2^2]}{\|x_0 - x_*\|_2^2} \right)^{1/k}, k \geq 1$$

图1表明,贪婪 Gauss-Seidel方法和贪婪随机坐标下降方法的收敛因子基本相同,但前者的实际收敛速率略好于后者。

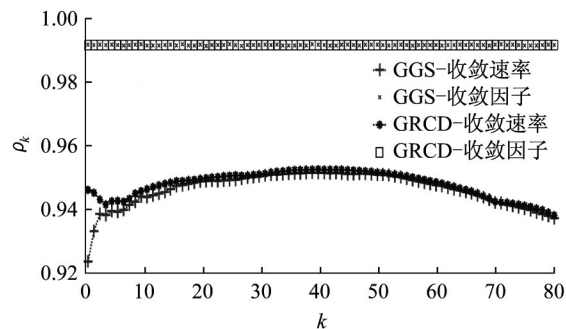
3 数值实验

通过数值实验比较贪婪 Gauss-Seidel (GGS)方法和贪婪随机坐标下降(GRCD)方法在求解线性最小二乘问题上的表现,其中矩阵 A 来自2类集合,一类是由 Matlab 函数 randn 随机生成,一类是来自不同应用的稀疏矩阵^[35]。为了公正直接地比较 GGS 和 GRCD 方法,采用的例子都来自文献^[15]。

主要依据迭代次数(IT)和 CPU 计算时间比较这2种方法。数值结果中的迭代次数和计算时间表示的是50次重复运行相应方法的平均值。此外,为了直观地比较2种方法,给出了 GGS 方法相对于 GRCD 方法的迭代次数加速比(IT speed-up),其定义为 GRCD 方法所需的迭代次数除以 GGS 方法所需的迭代次数,同时给出了 GGS 方法相对于 GRCD 方法的计算时间加速比(CPU speed-up),其定义为



a 随机矩阵 $A \in \mathbb{R}^{5000 \times 100}$



b 随机矩阵 $A \in \mathbb{R}^{10000 \times 100}$

图1 GGS和GRCD方法的收敛因子与实际收敛速率

Fig. 1 Convergence factors and rates of the GGS and GRCD methods

GRCD 方法所需的计算时间除以 GGS 方法所需的计算时间。对于来自文献[35]的稀疏矩阵,其稠密度定义为矩阵非零元素个数除以矩阵元素个数。

在数值实验中,解向量 x_* 由 Matlab 函数 randn 随机生成。对于相容的系统,令右边 $b = Ax_*$ 。对于不相容的系统,取 $b = Ax_* + r_0$,其中 r_0 是属于 A^T 的零空间的一个非零向量,由 Matlab 函数 null 生成。本节所有的数值实验,均取初始向量 $x_0 = 0$,停机准则为近似解的相对误差

$$\frac{\|x_k - x_*\|_2^2}{\|x_*\|_2^2} < 10^{-6}, \text{ 或者迭代次数超过 } 20 \text{ 万步。}$$

对于第一类随机生成的矩阵,线性系统相容时,2 类迭代法的迭代次数和计算时间的数值结果如表 1 所示;当线性系统不相容时,其数值结果如表 2 所示。从表 1 和表 2 可以看出,GGS 方法的迭代次数与 GRCD 方法几乎相同,但在计算时间上,GGS 方法的效率更高,计算时间加速比至少可达到 1.626。

表 1 GGS 和 GRCD 2 种方法在随机数据矩阵时求解相容系统的数值结果

Tab. 1 Numerical results of solving consistent systems with random data matrices by using the GGS and the GRCD methods

阶数	IT			CPU 计算时间		
	GGS	GRCD	IT speed-up	GGS	GRCD	CPU speed-up
1 000×50	126.000 0	128.240 0	1.017 8	0.013 8	0.063 1	4.590 9
1 000×100	374.000 0	361.500 0	0.966 6	0.046 6	0.170 3	3.657 7
1 000×150	603.000 0	600.560 0	0.996 0	0.104 4	0.319 4	3.059 9
2 000×50	108.000 0	106.260 0	0.983 9	0.012 5	0.052 5	4.200 0
2 000×100	246.000 0	245.720 0	0.998 9	0.046 6	0.131 3	2.818 8
2 000×150	439.000 0	445.680 0	1.015 2	0.109 4	0.269 1	2.460 0
3 000×50	105.000 0	104.960 0	0.999 6	0.017 2	0.055 6	3.236 4
3 000×100	231.000 0	236.880 0	1.025 5	0.061 9	0.144 4	2.333 3
3 000×150	409.000 0	409.040 0	1.000 1	0.140 0	0.283 4	2.024 6
4 000×50	96.000 0	99.740 0	1.039 0	0.019 4	0.057 2	2.951 6
4 000×100	205.000 0	209.120 0	1.020 1	0.067 8	0.138 8	2.046 1
4 000×150	337.000 0	343.660 0	1.019 8	0.163 8	0.266 2	1.626 0
5 000×50	96.000 0	95.380 0	0.993 5	0.025 0	0.060 0	2.400 0
5 000×100	195.000 0	203.080 0	1.041 4	0.072 8	0.156 9	2.154 5
5 000×150	340.000 0	337.020 0	0.991 2	0.181 9	0.297 8	1.637 5

表 2 GGS 和 GRCD 2 种方法在随机数据矩阵时求解不相容系统的数值结果

Tab. 2 Numerical results of solving inconsistent systems with random data matrices by using the GGS and the GRCD methods

阶数	IT			CPU 计算时间		
	GGS	GRCD	IT speed-up	GGS	GRCD	CPU speed-up
1 000×50	120.000 0	124.860 0	1.040 5	0.012 5	0.059 1	4.725 0
1 000×100	329.000 0	321.380 0	0.976 8	0.040 0	0.159 1	3.976 6
1 000×150	589.000 0	579.560 0	0.984 0	0.099 4	0.300 9	3.028 3
2 000×50	113.000 0	110.200 0	0.975 2	0.011 9	0.056 6	4.763 2
2 000×100	245.000 0	250.060 0	1.020 7	0.053 1	0.132 2	2.488 2
2 000×150	434.000 0	444.720 0	1.024 7	0.111 3	0.266 6	2.396 1
3 000×50	107.000 0	105.080 0	0.982 1	0.019 4	0.055 3	2.854 8
3 000×100	235.000 0	232.360 0	0.988 8	0.060 9	0.141 2	2.317 9
3 000×150	399.000 0	401.460 0	1.006 2	0.140 3	0.276 9	1.973 3
4 000×50	95.000 0	97.480 0	1.026 1	0.019 4	0.053 7	2.774 2
4 000×100	220.000 0	216.740 0	0.985 2	0.069 4	0.144 4	2.081 1
4 000×150	348.000 0	356.800 0	1.025 3	0.152 5	0.277 2	1.817 6
5 000×50	87.000 0	91.940 0	1.056 8	0.018 7	0.055 9	2.983 3
5 000×100	212.000 0	215.960 0	1.018 7	0.086 2	0.156 6	1.815 2
5 000×150	336.000 0	339.260 0	1.009 7	0.164 1	0.305 0	1.859 0

对于第二类矩阵,即文献[35]中的稀疏列满秩矩阵,线性系统相容时,2类迭代方法的迭代次数和计算时间的数值结果如表3所示;当线性系统不相容时,其数值结果如表4所示。从这2个表中可见,

除了非常病态的矩阵 Trefethen_300外, GGS方法和 GRCD方法的迭代次数几乎相同。但对于所有矩阵, GGS方法的计算时间都小于 GRCD方法,计算时间加速比至少可达到1.5315。

表3 GGS和GRCD 2种方法在真实数据矩阵时求解相容系统的数值结果

Tab. 3 Numerical results of solving consistent systems with real-world data matrices by using the GGS and the GRCD methods

矩阵名称	阶数	稠密度/%	条件数	IT			CPU计算时间		
				GGS	GRCD	IT speed-up	GGS	GRCD	CPU speed-up
abtaha1	14 596×209	1.68	12.23	14 888	13 966	0.938 0	8.255 0	12.642 8	1.531 5
Cities	55×46	53.04	207.15	29 181	40 937	1.402 9	0.174 7	1.849 7	10.588 6
divorce	50×9	50.00	19.39	634	647	1.020 0	0.002 8	0.031 6	11.222 2
WorldCities	315×100	23.87	66.00	5 011	5 011	1.000 0	0.077 2	0.291 6	3.777 3
Trefethen_300	300×300	5.20	1 772.69	3 210	1 374	0.428 0	0.041 6	0.073 4	1.766 9
cake5	37×37	17.02	15.42	1 477	1 624.4	1.099 8	0.006 6	0.070 0	10.666 7

表4 GGS和GRCD 2种方法在真实数据矩阵时求解不相容系统的数值结果

Tab. 4 Numerical results of solving inconsistent systems with real-world data matrices by using the GGS and the GRCD methods

矩阵名称	阶数	稠密度/%	条件数	IT			CPU计算时间		
				GGS	GRCD	IT speed-up	GGS	GRCD	CPU speed-up
abtaha1	14 596×209	1.68	12.23	11 264	12 571	1.116 0	6.275 0	11.303 4	1.801 3
Cities	55×46	53.04	207.15	28 449	39 752	1.397 3	0.171 6	1.827 8	10.653 9
divorce	50×9	50.00	19.39	552	497	0.899 8	0.002 8	0.021 3	7.555 6
WorldCities	315×100	23.87	66.00	3 532	3 576	1.012 5	0.055 0	0.205 0	3.727 3

因此,数值实验显示 GGS方法的迭代次数与 GRCD方法几乎相同,但在所有情形中,贪婪 Gauss-Seidel方法在计算时间上总是优于贪婪随机坐标下降方法。

4 结论与展望

针对大型线性最小二乘问题,提出了一类新的贪婪 Gauss-Seidel方法,理论分析了新方法的收敛性。数值实验表明,本文方法虽然与贪婪随机坐标下降方法的迭代次数几乎相同,但所需计算时间更少。主要原因在于本文方法不仅避免了矩阵A所有列范数的计算,而且指标集合的元素个数更少,从而减少了每次迭代的计算时间。

众所周知,对于 Gauss-Seidel方法而言,工作列的选取准则在整个迭代过程都有着十分重要的影响。如何设计一个随机的列选取准则改进贪婪 Gauss-Seidel方法需要进行进一步的研究。

作者贡献声明:

李寒宇:主要负责本文的指导和修订工作,包括文章框

架结构、方法分析和验证等。

张彦钧:主要负责本文的初稿撰写工作,包括算法推导、理论证明与数值实现。

参考文献:

- [1] BJÖRCK Å. Numerical methods for least squares problems [M]. Philadelphia: Society for Industrial and Applied Mathematics, 1996.
- [2] HIGHAM N J. Accuracy and stability of numerical algorithms [M]. Philadelphia: Society for industrial and applied mathematics, 2002.
- [3] SAAD Y. Iterative methods for sparse linear systems [M]. Philadelphia: Society for Industrial and Applied Mathematics, 2003.
- [4] HEFNY A, Needell D, Ramdas A. Rows versus columns: Randomized Kaczmarz or Gauss-Seidel for ridge regression [J]. SIAM Journal on Scientific Computing, 2017, 39(5): S528.
- [5] STROHMER T, VERSHYNIN R. A randomized Kaczmarz algorithm with exponential convergence [J]. Journal of Fourier Analysis and Applications, 2009, 15(2): 262.
- [6] LEVENTHAL D, LEWIS A S. Randomized methods for linear constraints: Convergence rates and conditioning [J]. Mathematics of Operations Research, 2010, 35(3): 641.

- [7] MA A, NEEDELL D, RAMDAS A. Convergence properties of the randomized extended Gauss-Seidel and Kaczmarz methods [J]. *SIAM Journal on Matrix Analysis and Applications*, 2015, 36(4): 1590.
- [8] EDALATPOUR V, HEZARI D, SALKUYEH D K. A generalization of the Gauss-Seidel iteration method for solving absolute value equations [J]. *Applied Mathematics and Computation*, 2017, 293: 156.
- [9] TU S, VENKATARAMAN S, WILSON A C, *et al.* Breaking locality accelerates block Gauss-Seidel [C]// *International Conference on Machine Learning*. Sydney: PMLR, 2017: 3482-3491.
- [10] CHEN L, SUN D, TOH K C. An efficient inexact symmetric Gauss-Seidel based majorized ADMM for high-dimensional convex composite conic programming [J]. *Mathematical Programming*, 2017, 161(1/2): 237.
- [11] TIAN Z, TIAN M, LIU Z, *et al.* The Jacobi and Gauss-Seidel-type iteration methods for the matrix equation $AXB=C$ [J]. *Applied Mathematics and Computation*, 2017, 292: 63.
- [12] XU Y. Hybrid Jacobian and Gauss-Seidel proximal block coordinate update methods for linearly constrained convex programming [J]. *SIAM Journal on Optimization*, 2018, 28(1): 646.
- [13] DU K. Tight upper bounds for the convergence of the randomized extended Kaczmarz and Gauss-Seidel algorithms [J]. *Numerical Linear Algebra with Applications*, 2019, 26(3): e2233.
- [14] RAZAVIYAYN M, HONG M, REYHANIAN N, *et al.* A linearly convergent doubly stochastic Gauss-Seidel algorithm for solving linear equations and a certain class of over-parameterized optimization problems [J]. *Mathematical Programming*, 2019, 176(1/2): 465.
- [15] BAI Z Z, WU W T. On greedy randomized coordinate descent methods for solving large linear least-squares problems [J]. *Numerical Linear Algebra with Applications*, 2019, 26(4): e2237.
- [16] BAI Z Z, WU W T. On greedy randomized Kaczmarz method for solving large sparse linear systems [J]. *SIAM Journal on Scientific Computing*, 2018, 40(1): A592.
- [17] BAI Z Z, WU W T. On relaxed greedy randomized Kaczmarz methods for solving large sparse linear systems [J]. *Applied Mathematics Letters*, 2018, 83: 21.
- [18] 杜亦疏,殷俊锋,张科. 求解大型稀疏线性方程组的贪婪距离随机Kaczmarz方法[J]. *同济大学学报(自然科学版)*, 2020, 48(8): 1224.
DU Yishu, YIN Junfeng, ZHANG Ke. Greedy randomized-distance Kaczmarz method for solving large sparse linear systems [J]. *Journal of Tongji University (Natural Science)*, 2020, 48(8): 1224.
- [19] NUTINI J. Greed is good: Greedy optimization methods for large-scale structured problems [D]. Vancouver: University of British Columbia, 2018.
- [20] ZHANG J J. A new greedy Kaczmarz algorithm for the solution of very large linear systems [J]. *Applied Mathematics Letters*, 2019, 91: 207.
- [21] DU K, GAO H. A new theoretical estimate for the convergence rate of the maximal weighted residual Kaczmarz algorithm [J]. *Numerical Mathematics: Theory, Methods and Applications*, 2019, 12(2): 627.
- [22] LIU Y, GU C Q. Variant of greedy randomized Kaczmarz for ridge regression [J]. *Applied Numerical Mathematics*, 2019, 143: 223.
- [23] NIU Y Q, ZHENG B. A greedy block Kaczmarz algorithm for solving large-scale linear systems [J]. *Applied Mathematics Letters*, 2020, 104: 106294.
- [24] ZHANG J, GUO J. On relaxed greedy randomized coordinate descent methods for solving large linear least-squares problems [J]. *Applied Numerical Mathematics*, 2020, 157: 372.
- [25] OSBORNE E E. On least squares solution of linear equations [J]. *Journal of the Association for Computing Machinery*, 1961, 8: 628.
- [26] HADDOCK J, NEEDELL D. On Motzkin's method for inconsistent linear systems [J]. *BIT Numerical Mathematics*, 2019, 59(2): 387.
- [27] REBROVA E, NEEDELL D. Sketching for Motzkin's iterative method for linear systems [C]// *53rd Asilomar Conference on Signals, Systems, and Computers*. Pacific Grove: IEEE, 2019: 271-275.
- [28] MOTZKIN T S, SCHOENBERG I J. The relaxation method for linear inequalities [J]. *Canadian Journal of Mathematics*, 1954, 6: 393.
- [29] NUTINI J, SEPEHRY B, LARADJI I, *et al.* Convergence rates for greedy Kaczmarz algorithms, and faster randomized Kaczmarz rules using the orthogonality graph [J]. *arXiv preprint arXiv*, 2016: 1612.07838.
- [30] PETRA S, POPA C. Single projection Kaczmarz extended algorithms [J]. *Numerical Algorithms*, 2016, 73(3): 791.
- [31] DU K, SI W, SUN X. Randomized extended average block Kaczmarz for solving least squares [J]. *SIAM Journal on Scientific Computing*, 2020, 42(6): A3541.
- [32] DU K, SUN X. A doubly stochastic block Gauss-Seidel algorithm for solving linear equations [J]. *Applied Mathematics and Computation*, 2021, 408: 126373.
- [33] NECOARA I. Faster randomized block Kaczmarz algorithms [J]. *SIAM Journal on Matrix Analysis and Applications*, 2019, 40(4): 1425.
- [34] LI H, ZHANG Y. Greedy block Gauss-Seidel methods for solving large linear least squares problem [J]. *arXiv preprint arXiv*, 2020: 2004.02476.
- [35] DAVIS T A, HU Y. The University of Florida sparse matrix collection [J]. *ACM Transactions on Mathematical Software (TOMS)*, 2011, 38(1): 1.