第七章 线性与非线性方程组的迭代解法

数值计算方法课程组

中国石油大学-华东

（课堂专属 切勿网传）

理论上, 直接法得到的解是准确的，但是我们可以看得出，它们的计算量都是 n^3 数量级，存储量为 n^2 量级，这在 n 比较小的时候还比较合适，但是对于现在的很多实际问题，往往要我们求解很大的 n 的矩阵，而且这些矩阵往往是系数矩阵就是这些矩阵含有大量的 0 元素。对于这类的矩阵，在用直接法时就会耗费大量的时间和存储单元. 因此我们有必要引入一类新的方法：迭代法.

迭代法具有的特点是速度快。与非线性方程的迭代方法一样，需要我们构造一个等价的方程，从而建立一个收敛序列，序列的极限值就是方程组的根. 例如，解方程

$$Ax = b$$

做等价变换

$$x = Mx + g$$

从而建立迭代格式

$$x^{(k+1)} = Mx^{(k)} + g, \quad k = 0, 1, \cdots$$

从初值 $x^{(0)}$ 出发，产生迭代序列 $x^{(k)}$, 若其收敛得到原问题的解。

迭代法的优点

- 程序简单, 存储量小;
- 适用于求解系数矩阵为大型稀疏矩阵问题.

目 录

目 录

§7.1.1 Jacobi迭代法

考虑非奇异线性代数方程组

$$Ax = b, \quad A = (a_{ij})_{n\times n}, \quad det(A) \neq 0 \tag{1.1}$$

令

$$A = D - L - U$$

其中

$$D = diag(a_{11}, a_{22}, \cdots, a_{nn}),$$

$$L = \begin{bmatrix} 0 & & & & \\ -a_{21} & 0 & & & \\ -a_{31} & -a_{32} & 0 & & \\ \vdots & \cdots & \ddots & \ddots & \\ -a_{11} & -a_{n2} & \cdots & -a_{n,n-1} & 0 \end{bmatrix}, U = \begin{bmatrix} 0 & -a_{12} & -a_{13} & \cdots & -a_{1n} \\ & 0 & -a_{23} & \cdots & -a_{2n} \\ & & 0 & \ddots & \vdots \\ & & & \ddots & -a_{n-1,n} \\ & & & & 0 \end{bmatrix}$$

可将 (1.1) 可写为如下等价形式

$$x = Bx + g, \quad B = D^{-1}(L+U), \quad g = D^{-1}b. \tag{1.2}$$

若给定初始向量 $x^{(0)} = (x_1^{(0)}, \cdots, x_n^{(0)})^T$ 从而可以建立如下迭代格式

$$x^{(k+1)} = Bx^{(k)} + g, \quad k = 0, 1, 2, 3, \cdots \tag{1.3}$$

此方法称为 Jacobi 迭代法，B 称为 Jacobi 迭代矩阵，g 称为常数项.
其分量形式为

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right), \quad i = 1, 2, \cdots, n \tag{1.4}$$

例：写出 Jacobi 迭代法求解下列方程组的迭代格式

$$\begin{bmatrix} 10 & 3 & 1 \\ 2 & -10 & 3 \\ 1 & 3 & 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 14 \\ -5 \\ 14 \end{bmatrix}$$

解：Jacobi 迭代公式为

$$\begin{aligned} x_1^{(k+1)} &= (14 - 3x_2^{(k)} - x_3^{(k)})/10 \\ x_2^{(k+1)} &= (-5 - 2x_1^{(k)} - 3x_3^{(k)})/(-10) \\ x_3^{(k+1)} &= (14 - x_1^{(k)} - 3x_2^{(k)})/10 \end{aligned}$$

Jacobi 迭代法算例

若取初值 $x^{(0)}=(0,0,0)^T$, 选取不同的精度要求计算，结果如下表

精度要求	迭代次数	近似解
0.001	9	$(1.0002507 \quad 1.0000694 \quad 1.0002507)$
0.0001	12	$(1.0000102 \quad 0.9999835 \quad 1.0000102)$
0.00001	14	$(0.9999981 \quad 1.0000020 \quad 0.9999981)$

§7.1.2 Gauss-Seidel 迭代法

在 Jacobi 迭代法 (1.4) 中，计算 $x_i^{(k+1)}$ 时，利用已经计算出来的 $x_1^{(k+1)}, x_2^{(k+1)}, \cdots, x_{i-1}^{(k+1)}$，可以得到 $\boxed{\text{Gauss-Seidel 迭代法}}$，即

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right), \quad i = 1, 2, \cdots, n \quad (1.5)$$

显然，Gauss-Seidel 迭代法是 Jacobi 迭代法的一种改进．其向量形式表示为

$$x^{(k+1)} = D^{-1} L x^{(k+1)} + D^{-1} U x^{(k)} + D^{-1} b, \quad k = 0, 1, 2, 3, \cdots \quad (1.6)$$

特别，矩阵 $D - L$ 可逆时，Gauss-Seidel 迭代法可表示为

$$x^{(k+1)} = (D - L)^{-1} U x^{(k)} + (D - L)^{-1} b, \quad k = 0, 1, 2, 3, \cdots \quad (1.7)$$

例：写出 Gauss-Seidel 迭代法求解下列方程组的迭代格式

$$\begin{bmatrix} 10 & 3 & 1 \\ 2 & -10 & 3 \\ 1 & 3 & 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 14 \\ -5 \\ 14 \end{bmatrix}$$

解：Gauss-Seidel 迭代公式为

$$\begin{aligned} x_1^{(k+1)} &= (14 - 3x_2^{(k)} - x_3^{(k)})/10 \\ x_2^{(k+1)} &= (-5 - 2x_1^{(k+1)} - 3x_3^{(k)})/(-10) \\ x_3^{(k+1)} &= (14 - x_1^{(k+1)} - 3x_2^{(k+1)})/10 \end{aligned}$$

Gauss-Seidel 迭代法算例

若取初值 $x^{(0)} = (0,0,0)^T$, 选取不同的精度要求计算，结果如下表

方法	精度要求	迭代次数	近似解
G-S	0.001	6	(1.0000390 1.0000277 0.9999878)
	0.0001	7	(0.9999929 0.9999949 1.0000022)
	0.00001	9	(0.9999998 0.9999998 1.0000001)
Jacobi	0.001	9	(1.0002507 1.0000694 1.0002507)
	0.0001	12	(1.0000102 0.9999835 1.0000102)
	0.00001	14	(0.9999981 1.0000020 0.9999981)

目录

§7.2.1 收敛的充分必要条件及误差估计

如前所述，Jacobi 迭代与 Gauss-Seidel 迭代都可以写成如下统一的形式

$$x^{(k+1)} = Mx^{(k)} + g, \quad k = 0, 1, \cdots$$

此格式称为单步定常线性迭代法. 当迭代产生的序列 $\{x^{(k)}\}_{k=0}^{\infty}$ 收敛到向量 x^*，则称迭代算法收敛，否则称为不收敛或发散.

问题

若迭代序列收敛，则

$$x^* = Mx^* + g \Longleftrightarrow Ax^* = b(???)$$

定义 (相容性)

如果方程组 $Ax = b$ 与 $x = Mx + g$ 等价，即存在可逆矩阵 G 使得

$$G(I - M) = A, \quad Gg = b$$

则称迭代法 $x^{(k+1)} = Mx^{(k)} + g$ 与已知方程组是相容的.

- Jacobi 迭代法:

$$M = D^{-1}(L + U) = I - D^{-1}A \Rightarrow D(I - M) = A$$

- Gauss-Seidel 迭代法:

$$M = (D - L)^{-1}U \Rightarrow (D - L)(I - M) = A$$

引理 (收敛的充分必要条件)

迭代算法 $x^{(k+1)} = Mx^{(k)} + g$ 收敛的充要条件是 $M^k \to 0$.

证明：假设迭代算法收敛，则有

$$\lim_{k\to\infty} x^{(k)} = x^* \Rightarrow x^* = Mx^* + g$$

设 x^* 为 $Ax = b$ 的解，由相容性可知

$$Ax^* = b \Leftrightarrow x^* = Mx^* + g$$

从而可知

$$x^{(k)} - x^* = M(x^{(k-1)} - x^*) = \cdots = M^k(x^{(0)} - x^*)$$

$$\lim_{k\to\infty} x^{(k)} = x^* \Leftrightarrow M^k \to 0.$$

定理 (充分必要条件)

迭代算法 $x^{(k+1)} = Mx^{(k)} + g$ 收敛的充要条件是 M 的谱半径满足 $\rho(M) < 1$.

注释

- 迭代法是否收敛取决于迭代矩阵的谱半径，与初始向量和常数项无关。
- 而对于同一个方程组，不同的迭代法对应的迭代矩阵的谱半径一般不会相同，因而收敛性也不同。

例如：若方程组系数矩阵满足

$$A_1 = \begin{bmatrix} 1 & 2 & -2 \\ 1 & 1 & 1 \\ 2 & 2 & 1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 2 & -1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -2 \end{bmatrix}$$

Jacobi 迭代法，$JM = D^{-1}(L+U)$， Gauss-Seidel 迭代法，$GM = (D-L)^{-1}U$.

对于 A_1 来说，

$$\rho(JM) = 0 < 1, \Rightarrow \text{Jacobi 迭代法收敛}$$

$$\rho(GM) = 2 > 1, \Rightarrow \text{Gauss-Seidel 迭代法不收敛}$$

对于 A_2 来说，

$$\rho(JM) = \sqrt{5}/2 > 1, \Rightarrow \text{Jacobi 迭代法不收敛}$$

$$\rho(GM) = \frac{1}{2} < 1, \Rightarrow \text{Gauss-Seidel 迭代法收敛}$$

注释

实际应用中，计算谱半径较为困难，因此判断迭代过程是否收敛需要考虑更易操作的条件.

定理 (误差估计 I)

若迭代矩阵 $\|M\| = q < 1$, 并假定范数 $\|I\| = 1$, 则迭代算法 $x^{(k+1)} = Mx^{(k)} + g$ 收敛，且其误差满足

$$\|x^{(k)} - x^*\| \leq \frac{q^k}{1-q}\|x^{(1)} - x^{(0)}\|.$$

证明:

$$\begin{aligned} x^{(k)} - x^* &= M^k(x^{(0)} - x^*) \\ \Rightarrow \|x^{(k)} - x^*\| &\leq \|M^k(x^{(0)} - x^*)\| \leq q^k\|x^{(0)} - x^*\| \end{aligned}$$

注意到

$$
\begin{aligned}
x^{(0)} - x^* &= x^{(0)} - (I-M)^{-1}g = (I-M)^{-1}((I-M)x^{(0)} - g) \\
&= (I-M)^{-1}(x^{(0)} - Mx^{(0)} - g) = (I-M)^{-1}(x^{(0)} - x^{(1)})
\end{aligned}
$$

由

$$\|(I-M)^{-1}\| \le \frac{1}{1-\|M\|} = \frac{1}{1-q},$$

所以

$$\|x^{(k)} - x^*\| \le q^k \|(I-M)^{-1}(x^{(0)} - x^{(1)})\| \le \frac{q^k}{1-q}\|x^{(1)} - x^{(0)}\|.$$

注释

- 利用矩阵的范数判定迭代收敛只是一个充分条件，通常采用矩阵的 1-范数、∞-范数来判定。
- 由上定理可知，可计算满足精度要求的近似解需要迭代次数。但此方法估计往往偏高，实际计算应用不便.

定理 (误差估计 II)

若迭代矩阵 $\|M\| = q < 1$, 并假定范数 $\|I\| = 1$, 则迭代算法 $x^{(k+1)} = Mx^{(k)} + g$ 收敛，且其误差满足

$$\|x^{(k)} - x^*\| \leq \frac{q}{1-q}\|x^{(k)} - x^{(k-1)}\|.$$

证明：

$$\begin{aligned} x^{(k)} - x^* &= Mx^{(k-1)} + g - (Mx^* + g) \\ &= Mx^{(k-1)} - Mx^* \\ &= Mx^{(k-1)} - M(I-M)^{-1}g \\ &= M(I-M)^{-1}(x^{(k-1)} - Mx^{(k-1)} - g) \\ &= M(I-M)^{-1}(x^{(k-1)} - x^{(k)}) \end{aligned}$$

两边取范数得结论.

定理 (收敛的充分条件 I)

设 B 是 *Jacobi* 迭代的迭代矩阵, 若 $\|B\|_\infty < 1$, 则 *Gauss-Seidel* 迭代算法收敛，且满足估计

$$\|x^{(k)} - x^*\|_\infty \le \frac{\mu^k}{1-\mu}\|x^{(1)} - x^{(0)}\|_\infty$$

其中

$$\mu = \max_i \Big(\sum_{j=i+1}^{n} |b_{ij}| / (1 - \sum_{j=1}^{i-1} |b_{ij}|) \Big)$$

且有 $\mu \le \|B\|_\infty < 1$，这里 b_{ij} 是矩阵 B 的元素.

定理 (收敛的充分条件 II)

设 B 是 *Jacobi* 迭代的迭代矩阵, 若 $\|B\|_1 < 1$, 则 *Gauss-Seidel* 迭代算法收敛，且满足估计

$$\|x^{(k)} - x^*\|_\infty \leq \frac{\tilde{\mu}^k}{(1-\tilde{\mu})(1-s)}\|x^{(1)} - x^{(0)}\|_\infty$$

其中

$$s = \max_j \sum_{i=j+1}^{n} |b_{ij}|, \quad \tilde{\mu} = \max_j (\sum_{i=1}^{j-1} |b_{ij}|/(1 - \sum_{i=j+1}^{n} |b_{ij}|)) \leq \|B\|_1 < 1.$$

定理（Jacobi 迭代收敛的充要条件）

若系数矩阵 A 对称，对角线元素 $a_{ii} > 0(i = 1, 2 \cdots, n)$, 则 $Jacobi$ 迭代算法收敛的充分必要条件是 A 和 $2D - A$ 都正定.

证明：对于 Jacobi 迭代算法，其迭代矩阵

$$B = I - D^{-1}A = D^{-1/2}(I - D^{-1/2}AD^{-1/2})D^{1/2}$$

故矩阵 B 与 $I - D^{-1/2}AD^{-1/2}$ 相似，故有相同的特征值；且 $D^{-1/2}AD^{-1/2}$, $I - D^{-1/2}AD^{-1/2}$, $2I - D^{-1/2}AD^{-1/2}$ 对称.
必要性. 假设 Jacobi 迭代算法收敛，即 $\rho(B) < 1$, 则矩阵 $I - D^{-1/2}AD^{-1/2}$ 特征值绝对值小于 1，即 $D^{-1/2}AD^{-1/2}$ 的特征值介于 $(0, 2)$ 之间，从而 A 正定. 另外 $2I - D^{-1/2}AD^{-1/2}$ 的特征值也全为正数，所以 $2I - D^{-1/2}AD^{-1/2}$ 也正定.

而
$$D^{-1/2}(2D-A)D^{-1/2}=2I-D^{-1/2}AD^{-1/2}$$
因此 $2D-A$ 都正定.
充分性. 因
$$D^{-1/2}AD^{-1/2}=D^{1/2}(I-B)AD^{-1/2}$$
由 A 正定，可知 $I-B$ 的特征值全大于零, 所以 B 的特征值均小于 1. 因为 $2D-A$ 正定，又
$$D^{-1/2}(2D-A)D^{-1/2}=D^{1/2}(I+B)D^{-1/2}$$
故 $I+B$ 的特征值均大于零, B 的特征值均大于-1，从而 $\rho(B)<1$, Jacobi 迭代法收敛.

定理

若系数矩阵 A 对称正定，则 *Gauss-Seidel* 迭代收敛.

证明：设 λ 为 Gauss-Seidel 迭代矩阵 $(D-L)^{-1}U$ 的一个特征值，v 为对于特征向量，则有

$$(D-L)^{-1}Uv=\lambda v\Rightarrow Uv=L^Tv=\lambda(D-L)v\Rightarrow v^*L^Tv=\lambda v^*Dv-\lambda v^*Lv$$

令 $\delta=v^*Dv, v^*Lv=\alpha+i\beta$, 则有

$$\begin{aligned}
v^*L^Tv &= \alpha-i\beta\\
\Rightarrow\quad & \lambda(\delta-(\alpha+i\beta))=\alpha-i\beta\\
\Rightarrow\quad & \lambda^2((\delta-\alpha)^2+\beta^2)=\alpha^2+\beta^2\\
\Rightarrow\quad & \lambda^2=\frac{\alpha^2+\beta^2}{(\delta-\alpha)^2+\beta^2}
\end{aligned}$$

因

$$0 < v^* A v = v^*(D - L - L^T)v = \delta - 2\alpha$$

从而可知

$$(\delta - \alpha)^2 + \beta^2 = \alpha^2 + \beta^2 + \delta(\delta - 2\alpha) > \alpha^2 + \beta^2$$

也即是

$$|\lambda| < 1.$$

注释

从上面定理可知，迭代算法收敛性的判别，可由方程系数矩阵性质来判别.

定义 (对角占优矩阵)

设矩阵 $A \in R^{n\times n}$, 若对所有的 $1 \le i \le n$ 都有

$$|a_{ii}| \ge \sum_{j=1, j\ne i}^{n} |a_{ij}|$$

并且上不等式至少对一个 i 有严格不等号成立，则称矩阵 A 为弱严格（行）对角占优的；若对所有的 i 都有严格不等式成立，则称矩阵 A 为严格（行）对角占优的或强对角占优的.

定义 (可约矩阵)

如果存在 n 阶排列矩阵 P，使

$$PAP^T = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix} \tag{2.8}$$

A_{11} 为 r 阶方阵，A_{22} 为 $n-r$ 阶方阵，则称 A 可约 *(可分)*；反之，如果不存在这样的排列矩阵，则称 A 不可约.

例如：下面矩阵 A 可约

$$A = \begin{bmatrix} -1 & 1 & 2 \\ 0 & 0 & 1 \\ 0 & -1 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 & -1 & 1 \\ 0 & 0 & 1 \\ -1 & 1 & 2 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & -1 & 0 \\ 1 & 0 & 0 \\ 2 & 1 & -1 \end{bmatrix}$$

注释

若系数矩阵 A 是可约的，则可通过行与列重排化为上面 (2.8) 式，从而可将方程组简化为低阶方程组。$A_{11}x_1 = b_1, A_{22}x_2 = b_2 - A_{21}x_1$

定义 (不可约对角占优)

若一个矩阵不可约，且弱严格对角占优，则称为不可约对角占优矩阵.

定理

若 A 是严格对角占优或不可约对角占优矩阵，则 A 非奇异的.

推论

若 A 是严格对角占优或不可约对角占优的对称矩阵，且对角线元素皆正，则 A 正定.

定理

若 A 是严格对角占优或不可约对角占优的矩阵，则 *Jacobi* 迭代和 *Gauss-Seidel* 迭代收敛.

证明：先证明 Jacobi 迭代收敛. 反证法. 因 A 是严格对角占优或不可约对角占优，则有 $|a_{ii}| > 0$, 故 D 可逆. 现假设 Jacobi 迭代矩阵 B 的某个特征值 $|\lambda| \geq 1$, 考察矩阵 $\lambda D - L - U$, 显然 $\lambda D - L - U$ 也是严格对角占优或不可约对角占优的，因此 $\lambda D - L - U$ 非奇异. 再由

$$\begin{aligned} \lambda I - B &= \lambda I - D^{-1}(L+U) \\ &= D^{-1}(\lambda D - L - U) \end{aligned}$$

可推得

$$det(\lambda I - B) = det(D^{-1})det(\lambda D - L - U) \neq 0$$

这与 λ 是 B 的特征值矛盾. 即 Jacobi 迭代矩阵 B 的所有特征值绝对值均小于 1，故 Jacobi 迭代收敛.

关于 Gauss-Seidel 迭代，只需要考虑矩阵 $\lambda D - \lambda L - U$，用同样的方法可证明 $(D - L)^{-1} U$ 的特征值的模均小于 1，即 Gauss-Seidel 迭代收敛.

收敛速度

一般的迭代格式为

$$x^{(k+1)} = Mx^{(k)} + g$$

通常用 $\|M^k\|$ 的大小来反映迭代算法的收敛速度.

定义 (平均收敛速度)

定义

$$R_k(M) = -\frac{\ln\|M^k\|}{k} = -\ln\|M^k\|^{\frac{1}{k}}$$

为 k 次迭代的平均收敛速度.

定义 (渐进收敛速度)

定义

$$R(M) = -\ln \rho(M)$$

为迭代算法的渐进收敛速度.

目 录

§7.3.1 超松弛迭代法概述

超松弛迭代法思想：类似于 G-S 迭代的改进方法，利用 k 次迭代值与 $k+1$ 次的 G-S 迭代值做加权平均。
由前面的知识可知，G-S 迭代算法

$$\bar{x}_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right), \quad i = 0, 1, 2, \cdots, n \tag{3.9}$$

做加权平均

$$x_i^{(k+1)} = \omega\bar{x}_i^{(k+1)} + (1-\omega)x_i^{(k)}$$

即得超松弛迭代法的分量形式

超松弛 (SOR) 迭代法:

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \omega\frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right) \quad (3.10)$$

其中 ω 称为松弛因子； $\omega = 1$ 时即为 G-S 迭代法. 记 $A = D - L - U$, 则超松弛 (SOR) 迭代法的矩阵形式为

$$x^{(k+1)} = (1-\omega)x^{(k)} + \omega D^{-1}(b + Lx^{(k+1)} + Ux^{(k)}) \quad (3.11)$$

或者

$$x^{(k+1)} = Sx^{(k)} + g, \quad g = \omega(D-\omega L)^{-1}b \quad (3.12)$$

迭代矩阵 $S = (D-\omega L)^{-1}[(1-\omega)D + \omega U]$.

例：写出 SOR 迭代法求解下列方程组的迭代格式

$$\begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 24 \\ 30 \\ -24 \end{bmatrix}$$

解：SOR 迭代公式为

$$\begin{aligned} x_1^{(k+1)} &= (1-\omega)x_1^{(k)} + \frac{\omega}{4}(24 - 3x_2^{(k)}) \\ x_2^{(k+1)} &= (1-\omega)x_2^{(k)} + \frac{\omega}{4}(30 - 3x_1^{(k+1)} + x_3^{(k)}) \\ x_3^{(k+1)} &= (1-\omega)x_3^{(k)} + \frac{\omega}{4}(-24 + x_2^{(k+1)}) \end{aligned}$$

若取初值 $x^{(0)} = (1,1,1)^T$, 选取不同的 ω 值计算，结果如下表

ω	Tol	迭代次数	近似解
$\omega = 1$	0.001	12	(3.0012790 3.9989342 -5.0002665)
	0.0001	16	(3.0001952 3.9998374 -5.0000407)
	0.00001	21	(3.0000186 3.9999845 -5.0000039)
$\omega = 0.95$	0.001	12	(3.0020191 3.9982705 -5.0004444)
	0.0001	18	(3.0001673 3.9998567 -5.0000368)
	0.00001	23	(3.0000210 3.9999820 -5.0000046)
$\omega = 1.25$	0.001	8	(2.9997451 4.0000653 -4.9998924)
	0.0001	10	(2.9999853 4.0000031 -4.9999935)
	0.00001	12	(2.9999993 4.0000001 -4.9999996)
$\omega = 1.5$	0.001	13	(3.0006104 4.0001741 -5.0007434)
$\omega = 1.95$	0.001	151	(2.9995106 4.0017780 -5.0027919)

其他格式

- 向后超松弛 (BSOR) 迭代格式

$$x^{(k+1)} = (1-\omega)x^{(k)} + \omega D^{-1}(b + Lx^{(k)} + Ux^{(k+1)}) \tag{3.13}$$

- 对称超松弛 (SSOR) 迭代格式

$$\begin{aligned} \bar{x}^{(k+1)} &= (1-\omega)x^{(k)} + \omega D^{-1}(b + Lx^{(k+1)} + Ux^{(k)}) \\ x^{(k+1)} &= (1-\omega)\bar{x}^{(k)} + \omega D^{-1}(b + L\bar{x}^{(k)} + Ux^{(k+1)}) \end{aligned}$$

例：写出 BSOR 及 SSOR 迭代法求解下列方程组的迭代格式

$$\begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 24 \\ 30 \\ -24 \end{bmatrix}$$

解： BSOR 迭代公式为

$$
\begin{aligned}
x_3^{(k+1)} &= (1-\omega)x_3^{(k)} + \frac{\omega}{4}(-24 + x_2^{(k)}) \\
x_2^{(k+1)} &= (1-\omega)x_2^{(k)} + \frac{\omega}{4}(30 - 3x_1^{(k)} + x_3^{(k+1)}) \\
x_1^{(k+1)} &= (1-\omega)x_1^{(k)} + \frac{\omega}{4}(24 - 3x_2^{(k+1)})
\end{aligned}
$$

SSOR 迭代公式为

$$
\begin{aligned}
\bar{x}_1^{(k+1)} &= (1-\omega)x_1^{(k)} + \frac{\omega}{4}(24 - 3x_2^{(k)}) \\
\bar{x}_2^{(k+1)} &= (1-\omega)x_2^{(k)} + \frac{\omega}{4}(30 - 3\bar{x}_1^{(k+1)} + x_3^{(k)}) \\
\bar{x}_3^{(k+1)} &= (1-\omega)x_3^{(k)} + \frac{\omega}{4}(-24 + \bar{x}_2^{(k+1)})
\end{aligned}
$$

$$
\begin{aligned}
x_3^{(k+1)} &= (1-\omega)\bar{x}_3^{(k)} + \frac{\omega}{4}(-24 + \bar{x}_2^{(k)}) \\
x_2^{(k+1)} &= (1-\omega)\bar{x}_2^{(k)} + \frac{\omega}{4}(30 - 3\bar{x}_1^{(k)} + x_3^{(k+1)}) \\
x_1^{(k+1)} &= (1-\omega)\bar{x}_1^{(k)} + \frac{\omega}{4}(24 - 3x_2^{(k+1)})
\end{aligned}
$$

若取初值 $x^{(0)} = (1,1,1)^T$, 选取 $\omega = 1.25$ 进行计算，结果如下表

所用方法	Tol	迭代次数	近似解
BSOR	0.001	8	(2.9998426 4.0003635 -4.9995660)
	0.0001	10	(2.9999991 4.0000051 -4.9999831)
	0.00001	11	(3.0000012 3.9999988 -5.0000027)
SSOR	0.001	18	(3.0008900 3.9985916 -5.0003161)
	0.0001	23	(3.0000939 3.9998514 -5.0000334)
	0.00001	28	(3.0000099 3.9999843 -5.0000035)

举例

考虑前面的问题

举例

考虑前面的问题

$$\begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 24 \\ 30 \\ -24 \end{bmatrix}$$

举例

考虑前面的问题

$$\begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 24 \\ 30 \\ -24 \end{bmatrix}$$

从而可知系数矩阵 A 及其 Jacobi 迭代矩阵 B

$$A = \begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & -3/4 & 0 \\ -3/4 & 0 & 1/4 \\ 0 & 1/4 & 0 \end{bmatrix}.$$

不难看出，矩阵 A 是对称正定的三对角矩阵，可以用定理结论. 可计算出

$$\rho(B) = \sqrt{0.625} \approx 0.790, \quad \rho(G-S) = 0.625.$$

而 SOR 方法的最优松弛因子

$$\omega_b = \frac{2}{1-\sqrt{1-0.625}} \approx 1.24,$$

从而可得

$$\rho(S_{\omega_b}) \approx 0.24 < \rho(G-S) < \rho(B).$$

显然，可知采用最优松弛因子的 SOR 法比 G-S 方法和 Jacobi 方法收敛的快得多.

§7.3.2 SOR 迭代格式的收敛性

定理

对任意的矩阵 $A \in R^{n\times n}$, 设其对角线元皆非零，则对所有实数 ω, 有 $\rho(S) \geq |\omega - 1|$, 其中 $S = (D - \omega L)^{-1}[(1-\omega)D + \omega U]$.

证明：令 $A = D - L - U$. 设迭代矩阵 S 的特征值为 $\lambda_1, \cdots, \lambda_n$

$$
\begin{aligned}
det(S) &= \lambda_1 \cdots \lambda_n \\
det(S) &= det(D - \omega L)^{-1} det([(1-\omega)D + \omega U]) \\
&= detD^{-1} det[(1-\omega)D] = (1-\omega)^n \\
\Rightarrow |1-\omega|^n &\leq |\lambda_1| \cdots |\lambda_n| \leq (\max_i |\lambda_i|)^n \\
\Rightarrow |\omega - 1| &\leq \rho(S)
\end{aligned}
$$

推论

如果 $Ax=b$ 的 SOR 迭代收敛，则

$$|\omega-1|<1 \quad or \quad 0<\omega<2.$$

定理

对任意的矩阵 $A\in R^{n\times n}$ 为对称正定矩阵，且 $0<\omega<2$，则解线性方程组 $Ax=b$ 的 SOR 迭代收敛.

证明：记 $A=D-L-U$. 因矩阵 A 对称正定，则 $A^T=A, L^T=U$, SOR 迭代矩阵为

$$S=(D-\omega L)^{-1}[(1-\omega)D+\omega U]$$

设 SOR 迭代矩阵的任意特征值为 λ, 其相应的特征向量为 x, 则

$$Sx=\lambda x \Rightarrow [(1-\omega)D+\omega U]x=\lambda(D-\omega L)x$$

上式两边与 x 做内积

$$(1-\omega)(Dx,x)+\omega(Ux,x)=\lambda[(Dx,x)-\omega(Lx,x)] \tag{3.14}$$

因 A 正定，故可知 D 正定，从而可知 $p=(Dx,x)>0$. 记 $(Lx,x)=\alpha+i\beta$, 则取共轭复数有 $(Ux,x)=\alpha-i\beta$, 代入 (3.14), 可以求得特征值

$$\lambda=\frac{(1-\omega)p+\omega\alpha-i\omega\beta}{p-\omega\alpha-i\omega\beta}, \quad |\lambda|^2=\frac{[p-(\omega p-\alpha)]^2+\omega^2\beta^2}{(p-\omega\alpha)^2+\omega^2\beta^2}$$

注意到上面分子减分母

$$[p-(\omega p-\alpha)]^2-(p-\omega\alpha)^2=p\omega(2-\omega)(2\alpha-p).$$

因为 A 正定，

$$(Ax,x)=(Dx,x)-(Lx,x)-(Ux,x)=p-2\alpha>0$$

利用 $0<\omega<2$, 可知 $|\lambda|^2<1$, 从而 SOR 迭代法收敛.

定理

设矩阵 $A \in R^{n\times n}$ 对称，其对角线元 $a_{ii} > 0$，若求解方程组 $Ax = b$ 的 SOR 迭代法收敛，则 A 正定，且 $0 < \omega < 2$.

证明：记第 k 次的迭代误差向量为

$$e^{(k)} = x^{(k)} - x^*, \quad \delta^{(k)} = e^{(k)} - e^{(k+1)} = x^{(k)} - x^{(k+1)}.$$

由 SOR 迭代算法可知 $e^{(k+1)} = Se^{(k)}$, 从而可得

$$\begin{aligned}
(D-\omega L)e^{(k+1)} &= [(1-\omega)D + \omega U]e^{(k)} \\
&= (D-\omega L)e^{(k)} - \omega(D-L-U)e^{(k)} \\
&= (D-\omega L)e^{(k)} - \omega A e^{(k)} \\
\Rightarrow (D-\omega L)\delta^{(k)} &= \omega A e^{(k)}
\end{aligned}$$

考察二次型

$$
\begin{aligned}
(Ae^{(k+1)}, e^{(k+1)}) &= (A(e^{(k)} - \delta^{(k)}), e^{(k)} - \delta^{(k)}) \\
&= (Ae^{(k)}, e^{(k)}) - 2(Ae^{(k)}, \delta^{(k)}) + (A\delta^{(k)}, \delta^{(k)})
\end{aligned}
$$

注意到 $L^T = U$, 所以 $((L - U)\delta^{(k)}, \delta^{(k)}) = 0$, 所以

$$
\begin{aligned}
& \omega[(Ae^{(k)}, e^{(k)}) - (Ae^{(k+1)}, e^{(k+1)})] \\
= \; & 2\omega(Ae^{(k)}, \delta^{(k)}) - \omega(A\delta^{(k)}, \delta^{(k)}) \\
= \; & 2((D - \omega L)\delta^{(k)}, \delta^{(k)}) - \omega(A\delta^{(k)}, \delta^{(k)}) \\
= \; & ((2D - \omega L - \omega D + \omega L + \omega U)\delta^{(k)}, \delta^{(k)}) \\
= \; & (2 - \omega)(D\delta^{(k)}, \delta^{(k)})
\end{aligned}
$$

因 SOR 算法收敛，故而 $0 < \omega < 2$, 又 $a_{ii} > 0$, 从而可知

$$(2-\omega)(D\delta^{(k)}, \delta^{(k)}) > 0 \quad \Rightarrow (Ae^{(k)}, e^{(k)}) > (Ae^{(k+1)}, e^{(k+1)})$$

说明上面的二次型单调递减趋于零. 若矩阵 A 不正定，则存在非零向量 $e^{(0)}$ 使得

$$(Ae^{(0)}, e^{(0)}) \leq 0$$

且有 $\delta^{(0)} = e^{(0)} - e^{(1)} = (I-S)e^{(0)} \neq 0$

$$(Ae^{(1)}, e^{(1)}) < (Ae^{(0)}, e^{(0)}) \leq 0$$

这与二次型单调递减趋于零矛盾.

定理 (SSOR 迭代法的收敛性)

设矩阵 $A \in R^{n\times n}$ 对称正定，且 $0 < \omega < 2$, 则解方程组 $Ax = b$ 的 SSOR 迭代法收敛.

§7.3.3 最佳松弛因子

定义 (2-循环矩阵)

设矩阵 $A \in R^{n\times n}$，$A = D - L - U$. 若存在排列矩阵 P 使得

$$PAP^T = \begin{bmatrix} D_1 & M_1 \\ M_2 & D_2 \end{bmatrix}$$

其中 D_1, D_2 为对角阵，则称矩阵 A 为 2-循环矩阵.

定义 (相容次序矩阵)

设矩阵 $A \in R^{n\times n}$，$A = D - L - U$. 若 $\alpha \neq 0$ 时，$\alpha D^{-1}L + \alpha^{-1}D^{-1}U$ 的特征值都与 α 无关，则称矩阵 A 为相容次序矩阵.

定理

设矩阵 $A \in R^{n\times n}$ 的所有对角元非零，且是 2-循环矩阵和相容次序矩阵，记 $B = D^{-1}(L+U)$ 为求解线性方程组 $Ax=b$ 的 Jacobi 迭代矩阵，且 B^2 的特征值均在 [0,1) 上；并记

$$\mu = \rho(B) \quad \omega_b = \frac{2}{1+\sqrt{1-\mu^2}}$$

则求解方程组 $Ax=b$ 的 SOR 迭代法的迭代矩阵 S 的谱半径满足

$$\rho(S) = \begin{cases} \frac{(\omega\mu+\sqrt{\omega^2\mu^2-4(\omega-1)})^2}{4} & 0<\omega<\omega_b \\ \omega-1 & \omega_b \le \omega \le 2 \end{cases}$$

$$\rho(S_{\omega_b}) = \min_{0\le\omega\le 2} \rho(S_\omega)$$

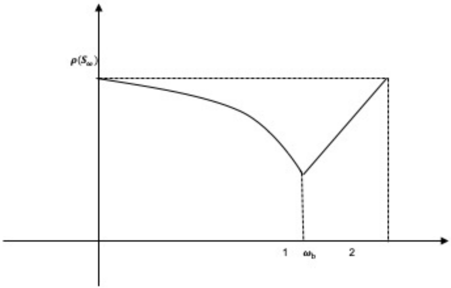
且当 $0<\omega<\omega_b$ 时，$\rho(S_\omega)$ 是 ω 的单调递减函数，并有

$$\lim_{\omega \to \omega_b} \frac{d}{d\omega} \rho(S_\omega) = -\infty.$$

注释

关于最佳松弛因子的进一步讨论，可参考文献《数值计算方法》（下册），林成森著，《矩阵迭代分析》，R.S.Varga 著，蒋尔雄等译.

谱半径与松弛因子关系图



知识小结

Jacobi、Gauss-Seidel、SOR 迭代算法收敛性判别

算法	收敛性条件
Jacobi 迭代 $B = D^{-1}(L+U)$	$\rho(B) < 1$ $A, 2D - A$ 对称正定 $a_{ii} > 0$ A 严格对角占优或不可约对角占优
G-S 迭代 $GS = (D-L)^{-1}U$	$\rho(GS) < 1$ Jacobi 迭代矩阵 $\\|B\\|_1 < 1$ Jacobi 迭代矩阵 $\\|B\\|_\infty < 1$ A 对称正定 A 严格对角占优或不可约对角占优
SOR 迭代 $S = (D-\omega L)^{-1}[(1-\omega)D + \omega U]$	A 对称正定且 $0 < \omega < 2$

目 录

共轭梯度法（Conjugate Gradient Methods）是一种求解对称正定线性方程组的方法.

它是 20 世纪 50 年代初期 Hestenes 和 Stiefel 首先提出.

它是一种变分方法，其思想是将求解方程组问题等价转化为一个二次泛函的极值问题.

§7.4.1 最速下降法

考虑如下的线性方程组

$$Ax = b, \quad A\text{对称正定}, x = (x_1, \cdots, x_n)^T, \quad b = (b_1, \cdots, b_n)^T$$

为了说明最速下降法的思想，定义二次型

$$\varphi(x) = (Ax, x) - 2(b, x) = x^T Ax - 2b^T x.$$

此二次型有如下性质:

1. 对任意的 $x \in R^n$, $\nabla\varphi(x) = 2(Ax - b)$;
2. 设 $x^* = A^{-1}b$ 为 $Ax = b$ 的解，则

$$\varphi(x^*) = -(Ax^*, x^*)$$

$$
\begin{aligned}
\varphi(x)-\varphi(x^*) &= (Ax,x)-2(b,x)+(Ax^*,x^*)\\
&= (Ax,x)-2(Ax^*,x)+(Ax^*,x^*)\\
&= (A(x-x^*),x-x^*)
\end{aligned}
$$

定理

设矩阵 A 对称正定，则求线性方程组 $Ax=b$ 等价于求二次型 $\varphi(x)$ 的极小值点

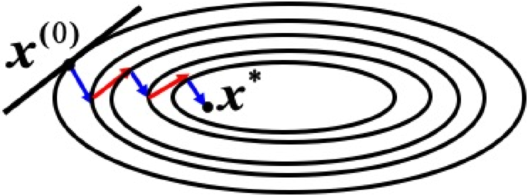
$$\varphi(x^*)=\min_{x\in R^n}\varphi(x)$$

证明： 必要性. 由二次型的性质可以直接得到. 下面证明充分性. 如果 $\varphi(x^*)=\min_{x\in R^n}\varphi(x)$, 则由极值的必要条件得 $\nabla\varphi(x^*)=2(Ax^*-b)=0$, 从而 $Ax^*=b$.

最速下降法的思想

最速下降法是指每次沿着函数值下降最快的方向寻找最小值点. 而函数值下降最快的方向是函数的负梯度方向.

其几何意义如下图所示



最速下降法的实现过程

选取初始向量 $x^{(0)}$，由二次泛函 $\varphi(x)$ 的基本性质 (1)

$$-\nabla\varphi(x^{(0)})/2 = b - Ax^{(0)} =: r^{(0)}$$

如果 $r^{(0)} = 0$, 则 $x^{(0)}$ 为方程组的解；如果 $r^{(0)} \neq 0$, 则沿 $r^{(0)}$ 方向进行一维极小搜索：求步长 α 使得

$$\min_{\alpha} \varphi(x^{(0)} + \alpha r^{(0)})$$

由

$$\frac{d}{d\alpha}\varphi(x^{(0)} + \alpha r^{(0)}) = 0$$

$$\Rightarrow \quad 2\alpha(Ar^{(0)}, r^{(0)}) - 2(r^{(0)}, r^{(0)}) = 0 \quad \Rightarrow \alpha_0 = \frac{(r^{(0)}, r^{(0)})}{(Ar^{(0)}, r^{(0)})}$$

令 $x^{(1)} = x^{(0)} + \alpha_0 r^{(0)}$, 完成第一次迭代；依次类推，重复上述过程.

最速下降法的算法

选取初始向量 $x^{(0)}$, 计算
$r^{(0)} = b - Ax^{(0)}$; $k = 0$;

while $r^{(0)} < \varepsilon$

$k = k + 1$;
$\alpha_{k-1} = \frac{(r^{(k-1)}, r^{(k-1)})}{(Ar^{(k-1)}, r^{(k-1)})}$;
$x^{(k)} = x^{(k-1)} + \alpha_{k-1} r^{(k-1)}$;
$r^{(k)} = b - Ax^{(k)}$;

end

搜索方向是正交的：

$$(r^{(k+1)}, r^{(k)}) = 0$$

注意到

$$\begin{aligned} r^{(k+1)} &= b - Ax^{(k+1)} \\ &= b - A(x^{(k)} + \alpha_k r^{(k)}) \\ &= r^{(k)} - \alpha_k A r^{(k)} \end{aligned}$$

缺陷：收敛速度慢.

引理

设矩阵 A 的特征值为 $0 < \lambda_1 \leq \cdots \lambda_n$, $P(t)$ 是关于 t 的一个实系数多项式，则

$$\|P(A)x\|_A \leq \max_{1\leq i\leq n} |P(\lambda_i)|\|x\|_A, \quad x \in R^n$$

其中 $\|x\|_A = \sqrt{x^T Ax}$

证明：设 $Ay_i = \lambda_i y_i$, 且满足

$$(y_i, y_j) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

则对任意的 x, $x = \sum_{i=1}^{n} \beta_i y_i$

所以

$$\begin{aligned}(P(A)x)^T A P(A)x &= (\sum_{i=1}^{n} \beta_i P(\lambda_i) y_i)^T A (\sum_{i=1}^{n} \beta_i P(\lambda_i) y_i) \\ &= \sum_{i=1}^{n} \lambda_i \beta_i^2 P^2(\lambda_i) \\ &\le \max_{1\le i\le n} P^2(\lambda_i) \sum_{i=1}^{n} \lambda_i \beta_i^2 \\ &= \max_{1\le i\le n} P^2(\lambda_i) x^T A x\end{aligned}$$

从而可得

$$\|P(A)x\|_A \le \max_{1\le i\le n} |P(\lambda_i)| \|x\|_A$$

定理（最速下降法的收敛性定理）

设矩阵 A 的特征值为 $0 < \lambda_1 \leq \cdots \leq \lambda_n$，则由最速下降法参产生的迭代序列 $\{x^{(k)}\}$ 满足

$$\|x^{(k)} - x^*\|_A \leq \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}\right)^k \|x^{(0)} - x^*\|_A, \quad x^* = A^{-1}b.$$

证明： 由最速下降法思想可知

$$\varphi(x^{(k)}) \leq \varphi(x^{(k-1)} + \alpha r^{(k-1)})$$

又注意到

$$\begin{aligned} \varphi(x) + {x^*}^T A x^* &= \varphi(x) - \varphi(x^*) \\ &= (x - x^*)^T A (x - x^*) \end{aligned}$$

$$
\begin{aligned}
(x^{(k)} - x^*)^T A (x^{(k)} - x^*) &= \varphi(x^{(k)}) + x^{*T} A x^* \\
&\le \varphi(x^{(k-1)} + \alpha r^{(k-1)}) + x^{*T} A x^* \\
&= (x^{(k-1)} + \alpha r^{(k-1)} - x^*)^T A (x^{(k-1)} + \alpha r^{(k-1)} - x^*) \\
(r^{(k-1)} = b - A x^{(k-1)}) \quad &= [(I - \alpha A)(x^{(k-1)} - x^*)]^T A [(I - \alpha A)(x^{(k-1)} - x^*)]
\end{aligned}
$$

记 $P_\alpha(t) = 1 - \alpha t$, 则由引理可知

$$
\|x^{(k)} - x^*\|_A \le \|P_\alpha(A)(x^{(k-1)} - x^*)\|_A \le \max_{1 \le i \le n} |P_\alpha(\lambda_i)| \|x^{(k-1)} - x^*\|_A
$$

注意到

$$
\min_{\alpha} \max_{\lambda_1 \le t \le \lambda_n} |1 - \alpha t| = \frac{\lambda_n - \lambda_1}{\lambda_1 + \lambda_n}
$$

$$
\Rightarrow \|x^{(k)} - x^*\|_A \le \frac{\lambda_n - \lambda_1}{\lambda_1 + \lambda_n} \|x^{(k-1)} - x^*\|_A.
$$

§7.4.2 共轭梯度法

备注

- 上定理告诉我们，当 $\lambda_1 \lll \lambda_n$ 时，最速下降法收敛非常慢.
- 改进最速下降法，得共轭梯度法.

定义（A-共轭向量组）

设 $A \in R^{n\times n}$ 为对称正定矩阵，若 R^n 中向量组 $\{p^{(k)}\}_{k=1}^{l}$ 满足 $(Ap^{(i)}, p^{(j)}) = 0, i \neq j$，则称其为 R^n 中一组 A-共轭向量组.

共轭梯度法的思想

利用一维极小搜索方法确定一组 A-共轭方向代替最速下降法中的正交方向来进行迭代.

共轭梯度法的实现过程

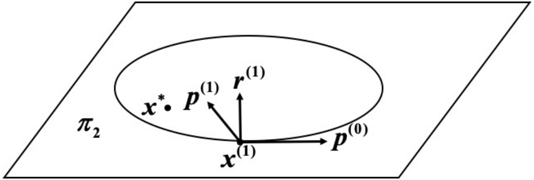
选取初始向量 $x^{(0)}$, 第一步取负梯度方向下山，即

$$
\begin{aligned}
p^{(0)} &= r^{(0)} = b - Ax^{(0)}, \\
\alpha^{(0)} &= \frac{\left(r^{(0)}, r^{(0)}\right)}{\left(Ap^{(0)}, p^{(0)}\right)}, \\
x^{(1)} &= x^{(0)} + \alpha^{(0)} p^{(0)}, \\
r^{(1)} &= b - Ax^{(1)},
\end{aligned}
$$

如何确定下一个搜索方向呢？ 在过点 $x^{(1)}$ 的由向量 $r^{(1)}$ 和 $p^{(0)}$ 所张成的下列二维平面内找出函数值下降最快的方向作为搜索方向 $p^{(1)}$

$$
\pi_2 = \{x = x^{(1)} + \xi r^{(1)} + \eta p^{(0)} : \xi, \eta \in R\}
$$

向量 $r^{(1)}$, $p^{(0)}$ 和 $p^{(1)}$ 的几何意义



由此可知 $\varphi(x)$ 在平面 π_2 可表示为

$$\begin{aligned}\varphi(x^{(1)}+\xi r^{(1)}+\eta p^{(0)}) &= (x^{(1)}+\xi r^{(1)}+\eta p^{(0)})^T A(x^{(1)}+\xi r^{(1)}+\eta p^{(0)}) \\ &\quad -2b^T(x^{(1)}+\xi r^{(1)}+\eta p^{(0)}) \\ &= (A(x^{(1)}+\xi r^{(1)}+\eta p^{(0)}),(x^{(1)}+\xi r^{(1)}+\eta p^{(0)})) \\ &\quad -2(b,(x^{(1)}+\xi r^{(1)}+\eta p^{(0)}))\end{aligned}$$

从而

$$\begin{aligned}\frac{\partial\varphi}{\partial\xi} &= 2(A(x^{(1)}+\xi r^{(1)}+\eta p^{(0)}), r^{(1)}) - 2\xi(b, r^{(1)}) \\ &= 2[\xi(Ar^{(1)}, r^{(1)}) + \eta(Ap^{(0)}, r^{(1)}) - (r^{(1)}, r^{(1)})] = 0 \\ \frac{\partial\varphi}{\partial\eta} &= 2[\xi(Ar^{(1)}, p^{(0)}) + \eta(Ap^{(0)}, p^{(0)})] = 0\end{aligned}$$

从而可得

$$\widetilde{x} = x^{(1)} + \xi_0 r^{(1)} + \eta_0 p^{(0)},$$

其中 ξ_0, η_0 满足

$$\begin{aligned}\xi_0(Ar^{(1)}, r^{(1)}) + \eta_0(Ap^{(0)}, r^{(1)}) &= (r^{(1)}, r^{(1)}) \\ \xi_0(Ar^{(1)}, p^{(0)}) + \eta_0(Ap^{(0)}, p^{(0)}) &= 0\end{aligned}$$

取搜索方向 $p^{(1)}$ 作为下一步下山方向

$$p^{(1)} = \frac{1}{\xi_0}(\widetilde{x} - x^{(1)}) = r^{(1)} + \frac{\eta_0}{\xi_0}p^{(0)}$$

沿该方向进行一维搜索得步长为

$$\alpha_1 = \frac{(r^{(1)}, p^{(1)})}{(Ap^{(1)}, p^{(1)})}.$$

记

$$\beta_0 = \frac{\eta_0}{\xi_0} = -\frac{(Ar^{(1)}, p^{(0)})}{(Ap^{(0)}, p^{(0)})}$$
$$x^{(2)} = x^{(1)} + \alpha_1 p^{(1)}, \quad p^{(1)} = r^{(1)} + \beta_0 p^{(0)}$$

重复上面过程，得到共轭梯度迭代序列.

一般计算公式

若已知 $p^{(k)}$, 则第 $k+1$ 步迭代计算如下

$$\begin{aligned}
\alpha_k &= \frac{(r^{(k)}, p^{(k)})}{(Ap^{(k)}, p^{(k)})} \\
x^{(k+1)} &= x^{(k)} + \alpha_k p^{(k)} \\
r^{(k+1)} &= b - Ax^{(k+1)} \\
\beta_k &= -\frac{(Ar^{(k+1)}, p^{(k)})}{(Ap^{(k)}, p^{(k)})} \\
p^{(k+1)} &= r^{(k+1)} + \beta_k p^{(k)}
\end{aligned}$$

注意到

$$(r^{(k+1)}, p^{(k)}) = (b - Ax^{(k+1)}, p^{(k)}) = (r^{(k)}, p^{(k)}) - \alpha_k(Ap^{(k)}, p^{(k)}) = 0$$

$$(r^{(k)}, p^{(k)}) = (r^{(k)}, r^{(k)} + \beta_{k-1} p^{(k-1)}) = (r^{(k)}, r^{(k)})$$

从而

$$\begin{aligned}
\alpha_k &= \frac{(r^{(k)}, r^{(k)})}{(Ap^{(k)}, p^{(k)})} \\
\beta_k &= -\frac{(r^{(k+1)}, Ap^{(k)})}{(Ap^{(k)}, p^{(k)})} = -\frac{(r^{(k+1)}, \alpha_k^{-1}(r^{(k)} - r^{(k+1)}))}{(Ap^{(k)}, p^{(k)})} \\
&= \alpha_k^{-1} \frac{(r^{(k+1)}, r^{(k+1)})}{(Ap^{(k)}, p^{(k)})} = \frac{(r^{(k+1)}, r^{(k+1)})}{(r^{(k)}, r^{(k)})}
\end{aligned}$$

算法

选取初始向量 $x^{(0)}$, 计算 $p^{(0)} = r^{(0)} = b - Ax^{(0)},\ k = 0$
while $r^{(k)} \neq 0$

$$
\begin{aligned}
\alpha_k &= \frac{\left(r^{(k)}, r^{(k)}\right)}{\left(Ap^{(k)}, p^{(k)}\right)} \\
x^{(k+1)} &= x^{(k)} + \alpha_k p^{(k)} \\
r^{(k+1)} &= r^{(k)} - \alpha_k A p^{(k)} \\
\beta_k &= \frac{\left(r^{(k+1)}, r^{(k+1)}\right)}{\left(r^{(k)}, r^{(k)}\right)} \\
p^{(k+1)} &= r^{(k+1)} + \beta_k p^{(k)} \\
k &= k + 1;
\end{aligned}
$$

end

定理

由共轭梯度法的序列 $\{r^{(i)}, p^{(i)}\}$ 满足如下性质:

$$
\begin{aligned}
&(1) \quad (r^{(j)}, p^{(i)}) = 0, \quad 0 \le i < j \le k; \\
&(2) \quad (r^{(j)}, r^{(i)}) = 0, \quad i \ne j, \quad 0 \le i, j \le k; \\
&(3) \quad (Ap^{(j)}, p^{(i)}) = 0, \quad i \ne j, \quad 0 \le i, j \le k; \\
&(4) \quad span\{r^{(0)}, \cdots, r^{(k)}\} = span\{p^{(0)}, \cdots, p^{(k)}\} =: K(A, r^{(0)}, k+1),
\end{aligned}
$$

其中

$$K(A, r^{(0)}, k+1) = span\{r^{(0)}, Ar^{(0)}, \cdots, A^k r^{(0)}\}.$$

称为 *Krylov* 子空间.

注释

- 上面定理表明 $\{r^{(0)}, \cdots, r^{(k)}\}$, $\{p^{(0)}, \cdots, p^{(k)}\}$ 分别为 Krylov 子空间 $K(A, r^{(0)}, k+1)$ 的正交基和共轭正交基.
- 共轭梯度法最多用 n 步便可得到方程组的精确解.
- 理论上讲，共轭梯度法是直接算法.

定理

用共轭梯度法得到的近似解 $x^{(k)}$ 满足

$$\varphi(x^{(k)}) = \min\{\varphi(x) : x \in x^{(0)} + K(A, r^{(0)}, k)\}$$

或

$$\|x^{(k)} - x^*\|_A = \min\{\|x^{(k)} - x^*\|_A : x \in x^{(0)} + K(A, r^{(0)}, k)\}.$$

例题

用 CG 方法求解下列方程组：$x^{(0)}=(0,0,0)^T$

$$\begin{bmatrix} 2 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 3 \\ 1 \\ 3 \end{bmatrix}$$

解： 易证系数矩阵对称正定。
第一步： 计算 $p^{(0)}=r^{(0)}=b-Ax^{(0)}=(3,1,3)^T$,

$$\alpha_0=\frac{(r^{(0)},r^{(0)})}{(Ap^{(0)},p^{(0)})}=\frac{19}{55}$$

$$x^{(1)}=x^{(0)}+\alpha_0 p^{(0)}=\frac{19}{55}(3,1,3)^T$$

第二步：计算

$$
\begin{aligned}
r^{(1)} &= r^{(0)} - \alpha_0 A p^{(0)} = -\frac{6}{55}(1, -6, 1)^T \\
\beta_0 &= \frac{(r^{(1)}, r^{(1)})}{(r^{(0)}, r^{(0)})} = \frac{72}{3025}, \alpha_1 = \frac{(r^{(1)}, r^{(1)})}{(Ap^{(1)}, p^{(1)})} = \frac{55}{57} \\
p^{(1)} &= r^{(1)} + \beta_0 p^{(0)} = -\frac{114}{3025}(-1, 18, -1)^T \\
x^{(2)} &= x^{(1)} + \alpha_1 p^{(1)} = (1, 1, 1)^T \\
r^{(2)} &= b - Ax^{(2)} = (0, 0, 0)^T
\end{aligned}
$$

§7.4.3 实用的共轭梯度法及其收敛性

理论上，共轭梯度法为直接法，最多 n 步可得精确解；

实际上，因舍入误差的出现，使得 $r^{(k)}$ 的正交性损失，不能有限步终止迭代计算；

实际计算，将 CG 方法看作迭代方法使用，通过 $\|r^{(k)}\|$ 的大小或最大迭代步数决定是否终止计算.

实用迭代算法:

x=初值;
$k=0, r=b-Ax, \rho=r^Tr$;
while $(\sqrt{\rho}>\varepsilon\|b\|_2)\&\ (k\le k_{\max})$
  $k=k+1$;
  if $k=1$
    $p=r$;
  else
    $\beta=\rho/\tilde{\rho}, p=r+\beta p$;
  end

  $w=A\rho; \alpha=\rho/p^Tw; x=x+\alpha p$;
  $r=r-\alpha w; \tilde{\rho}=\rho; \rho=r^Tr$;
end

共轭梯度法的优点

- 算法中，矩阵 A 仅用于求向量 $w = Ap$, 可充分运用矩阵稀疏性质；
- 不需要预先估计参数，可以直接计算；
- 每次迭代所需的计算，主要是向量之间的运算，便于并行化.

定理

如果矩阵 $A = I + B$ 且 $rank(B) = r$，则共轭梯度法最多迭代 $r + 1$ 步可得方程组的精确解.

证明：注意到 $rank(B) = r$ 意味着

$$\text{span}\{r^{(0)}, Ar^{(0)}, \cdots, A^k r^{(0)}\} = \text{span}\{r^{(0)}, Br^{(0)}, \cdots, B^k r^{(0)}\}$$

的维数不会超过 $r + 1$，固有前面的定理可知，结论成立.

上面定理说明，当线性方程组系数矩阵与单位矩阵相差一个秩为 r 的矩阵且 r 很小时，CG 算法收敛很快.

定理

用共轭梯度法得到方程组的近似解 $x^{(k)}$ 满足

$$\|x^{(k)} - x^*\|_A \leq 2\left(\frac{\sqrt{\kappa_2}-1}{\sqrt{\kappa_2}+1}\right)^k \|x^{(0)} - x^*\|_A, (\kappa_2 = \kappa_2(A)).$$

- 其证明可参考北京大学出版社《数值线性代数》.
- 定理显示 CG 法比最速下降法收敛得快.

目 录

§7.5 非线性方程组的迭代解法

考虑 n 个方程的 n 元非线性方程组的一般形式

$$\begin{cases} f_1(x_1,\cdots,x_n)=0 \\ f_2(x_1,\cdots,x_n)=0 \\ \vdots \\ f_n(x_1,\cdots,x_n)=0 \end{cases}$$

f_i 为定义在区域 D 的 n 元实值函数，且至少一个为非线性函数. 如

$$\begin{cases} xy=z^3+1 \\ xyz+y^2=x^2+2 \\ e^x+z=e^y+3 \end{cases}$$

本节主要介绍牛顿迭代法.

§7.5.1 牛顿迭代法

Newton 迭代法的迭代格式：利用多元函数 Taylor 展开公式可得

$$f(x^*) = f(x^{(k)}) + \nabla f(x^{(k)})(x^* - x^{(k)}) + R(x^*, x^{(k)}),$$

其中 $R(x, x^{(k)}) = O((x - x^{(k)})^2)$，

$$\nabla f(x^{(k)}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

为函数 f 的 Jacobi 矩阵．忽略高阶无穷小项，可得近似

$$\nabla f(x^{(k)})(x^* - x^{(k)}) = -f(x^{(k)})$$

若 $\nabla f(x^{(k)})$ 可逆，则可构造迭代格式

$$x^{(k+1)} = x^{(k)} - \nabla f(x^{(k)})^{-1} f(x^{(k)}), \quad k = 0, 1, 2 \cdots$$

称上述公式为 Newton 迭代格式。

存在的问题

在实际迭代时，Newton 迭代方法计算量庞大，既要计算 Jacobi 矩阵，还要求逆.

实际计算时，分两步完成：

$$\nabla f(x^{(k)}) y^{(k)} = -f(x^{(k)}), \quad x^{(k+1)} = x^{(k)} + y^{(k)}.$$

§7.5.2 Broyden 秩 1 方法

Broyden 秩 1 方法是 Newton 迭代方法一种改进. 其优点如下

- 仅需第一步计算 Jacobi 矩阵并求逆，以后每步只需作矩阵和向量相乘;
- 其运算量 $O(n^2)$, 大大降低工作量.

Broyden 秩 1 方法的原理如下:
由多元 Taylor 展开公式可知

$$f(x^{(k-1)}) = f(x^{(k)}) + \nabla f(x^{(k)})(x^{(k-1)} - x^{(k)}) + R(x^{(k-1)}, x^{(k)}),$$

其近似公式为

$$\nabla f(x^{(k)})(x^{(k)} - x^{(k-1)}) = f(x^{(k)}) - f(x^{(k-1)}),$$

记

$$A^{(k)} = \nabla f(x^{(k)}), y^{(k-1)} = (x^{(k)} - x^{(k-1)}), g^{(k-1)} = f(x^{(k)}) - f(x^{(k-1)})$$

从而有牛顿方程

$$A^{(k)} y^{(k-1)} = g^{(k-1)}.$$

令

$$\begin{aligned} A^{(k)} &= A^{(k-1)} + \Delta A^{(k)} \longrightarrow \text{秩1矩阵} \\ &= A^{(k-1)} + u^{(k-1)} y^{(k-1)\,T} \longrightarrow u^{(k-1)}\text{为待定向量} \end{aligned}$$

将其代入牛顿迭代方程得

$$g^{(k-1)} = A^{(k)} y^{(k-1)} = A^{(k-1)} y^{(k-1)} + u^{(k-1)} y^{(k-1)\,T} y^{(k-1)}$$

从而可得

$$u^{(k-1)} = \frac{g^{(k-1)} - A^{(k-1)} y^{(k-1)}}{y^{(k-1)T} y^{(k-1)}}$$

$$A^{(k)} = A^{(k-1)} + \frac{g^{(k-1)} - A^{(k-1)} y^{(k-1)}}{y^{(k-1)T} y^{(k-1)}} (y^{(k-1)})^T$$

引理

如果 A 非奇异，$x, y \in R^n$, 若 $y^T A x \neq -1$, 则 $A + xy^T$ 可逆，且

$$(A + xy^T)^{-1} = A^{-1} - \frac{A^{-1} x y^T A^{-1}}{1 + y^T A^{-1} x}.$$

由上面的引理可得 $A^{(k)}$ 的逆与 $A^{(k-1)}$ 的逆之间的关系

$$(A^{(k)})^{-1} = (A^{(k-1)})^{-1} + \frac{[(A^{(k-1)})^{-1}g^{(k-1)} - y^{(k-1)}](y^{(k-1)})^T(A^{(k-1)})^{-1}}{(y^{(k-1)})^T(A^{(k-1)})^{-1}g^{(k-1)}} \tag{5.15}$$

Broyden 秩 1 算法

选初值 $x^{(0)}$, 计算 $(A^{(0)})^{-1} = (\nabla f(x^{(0)}))^{-1}$, $x^{(1)} = x^{(0)} - (A^{(0)})^{-1}f(x^{(0)})$,
for k=1,2,...,

$y = x^{(k)} - x^{(k-1)}$,$g = f(x^{(k)}) - f(x^{(k-1)})$;

运用公式 (5.15) 计算 $(A^{(k)})^{-1}$;

$x^{(k+1)} = x^{(k)} - (A^{(k)})^{-1}f(x^{(k)})$;

若迭代条件满足，终止循环并输出 $x^{(k+1)}$.

end

知识小结

主要内容

Jacobi 迭代法、Gauss-Seidel 迭代法、SOR 迭代法、CG 算法
收敛性分析
非线性方程组求解的迭代算法：Newton 迭代法、Broyden 算法

重点及难点

重点：Jacobi 迭代法、Gauss-Seidel 迭代法、SOR 迭代法、CG 算法
难点：CG 算法、收敛性

Many thanks for your attention !

中國石油大學(华东)
CHINA UNIVERSITY OF PETROLEUM