最优化理论与方法

第13讲

共轭梯度法

第5章 无约束问题算法(III)
— 共轭梯度法

第一节 二次函数极小值问题的共轭方向法

第二节 非线性共轭梯度法

前面介绍的Newton法收敛较快，但须存储n阶矩阵，不适合求解大规模问题，而最速下降法虽然存储小但收敛慢。在这一章我们介绍存储小且收敛快的共轭梯度法。

首先介绍　　共轭方向

及求解二次函数的共轭方向法.

$5.1 二次函数极小值问题的共轭方向法

背景

将下降算法用于求解二元二次函数

$$f(x)=\frac{1}{2}xQx+b^{T}x+c$$

的极小点，这里$x\in R^{2},Q\in R^{2\times 2}$对称正定，

$b\in R^{2},c\in R.$

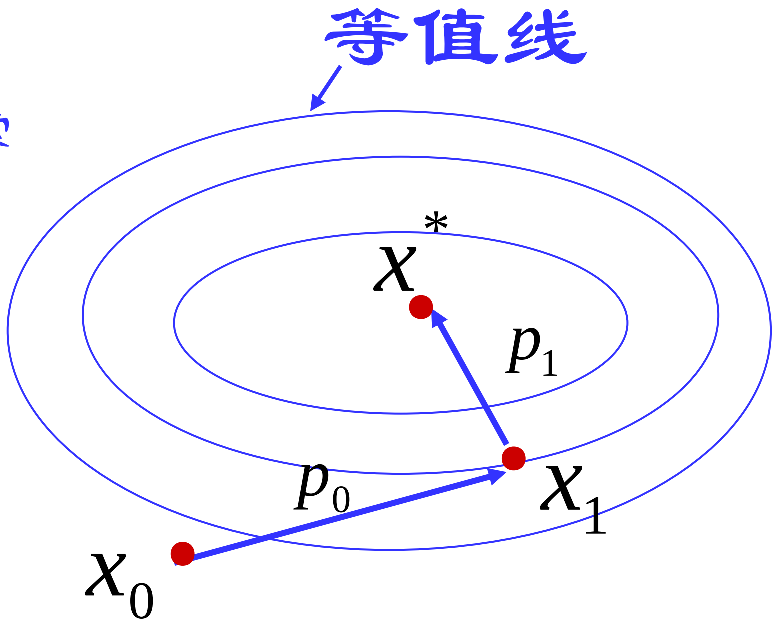
给定初始点x_0及下降方向p_0.线性搜索

后得点x_1满足

$$\nabla f(x_1)^{T}p_0=0$$

现在找方向p_1直接指向最优解x^{*},

如何找呢？



(1) 方向p_1满足的条件

设从x_1沿p_1到x^*的最佳步长为$\alpha_1 (\neq 0)$,

则有

$$x^* = x_1 + \alpha_1 p_1$$

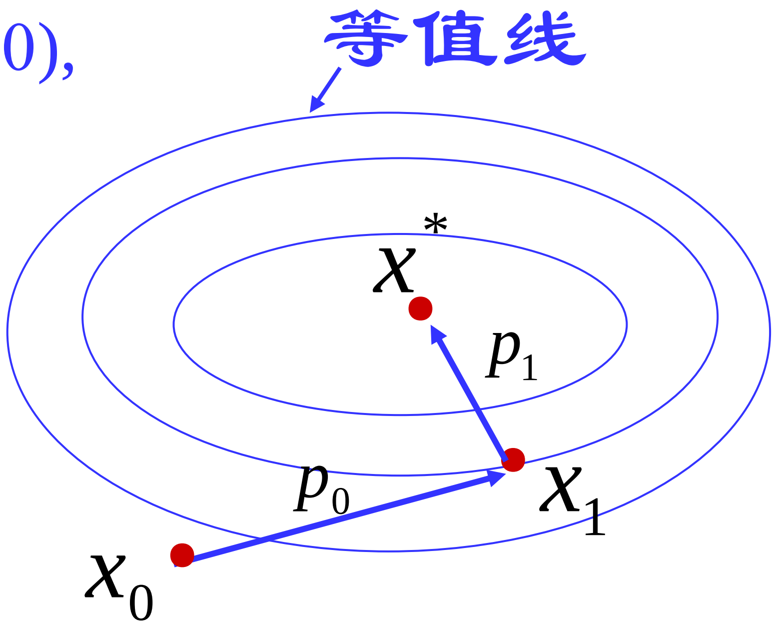
因此 $$\nabla f(x^*) = \nabla f(x_1) + \alpha_1 Q p_1$$

由于 $$\nabla f(x^*) = Q x^* + b = 0$$

所以 $$\nabla f(x_1) + \alpha_1 Q p_1 = 0$$

等式左乘 p_0^T，并注意到$\alpha_1 \neq 0$,得到

$$p_0^T Q\, p_1 = 0$$



这就是本章的核心：p_0和p_1关于Q共轭

返回

(2) 方向p_1的计算

由右图可知，p_1可以表示成p_0与$-\nabla f(x_1)$的线性组合.令

$$p_1 = -\nabla f(x_1) + \beta p_0$$

由上面的条件得到

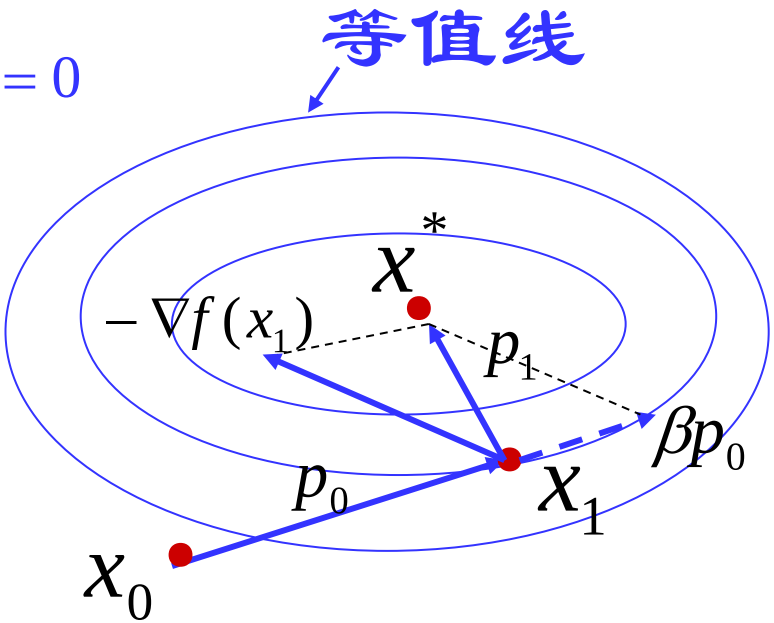
$$p_0^T Q p_1 = -p_0^T Q \nabla f(x_1) + \beta p_0^T Q p_0 = 0$$

由此解得

$$\beta = \frac{p_0^T Q \nabla f(x_1)}{p_0^T Q p_0}$$

所以

$$p_1 = -\nabla f(x_1) + \frac{p_0^T Q \nabla f(x_1)}{p_0^T Q p_0} p_0$$



从上面的(1)和(2)可以得出稍后要介绍的共轭方向法或共轭梯度法的基本原理

返回

共轭方向的定义

考察二次函数极小化问题

$$\min f(x) = \frac{1}{2}x^T Qx + q^T x \qquad (5.2)$$

其中Q是n阶对称正定矩阵，q为n维常量

定义5.1.1 设$A \in R^{n \times n}$对称正定，$d_1, d_2, \cdots, d_m$是R^n中非零向量. 若

$$d_i^T A d_j = 0, \qquad \forall i, j, i \neq j$$

则称向量组$d_1, d_2, \cdots, d_m$关于矩阵A相互共轭.

若$Q=I$，向量共轭就是正交，所以共轭是正交的推广.

并且共轭向量组也是线性无关的. 即有

定理5.1.1 设$A \in R^{n\times n}$对称正定，$d_1, d_2, \cdots, d_n$关于矩阵A相互共轭，则向量组 $d_1, d_2, \cdots, d_n$ 线性无关.

推论 在R^n中，相互共轭的向量组的向量个数不超过n

利用共轭方向,我们提出求二次函数(5.2)的极小值问题的共轭方向法

思想

从某个初始点 x_0 出发,依次沿关于 Q 共轭的 n 个方向 $d_0, d_1, \cdots, d_{n\text{-}1}$ 进行精确搜索

$$x_0 \xrightarrow[\alpha_0]{d_0} x_1 \xrightarrow[\alpha_1]{d_1} x_2 \xrightarrow[\alpha_2]{d_2} \cdots \xrightarrow[\alpha_{n-1}]{d_{n-1}} x_n$$

即令

$$x_{k+1} = x_k + \alpha_k d_k,\ k = 0,\ 1,\ 2,\ \cdots, n\text{-}1$$

其中 α_k 满足:

$$f(x_k + \alpha_k d_k) = \min_{\alpha \in R} f(x_k + \alpha d_k)$$

注意:这里的搜索是在整个实数轴上进行,因为共轭方向不一定是下降方向

下面的定理说明了共轭方向法如何求出解

定理5.1.2 设函数f由(5.2)给出，非零向量组$d_0, d_1, \cdots, d_{n-1}$关于矩阵$Q$相互共轭. $\forall x_0 \in R^n$，设迭代格式

$$x_{k+1} = x_k + \alpha_k d_k, k = 0,\ 1,\ \cdots,\ n-1$$

其中步长α_k由精确搜索计算. 则

$$\nabla f(x_{k+1})^T d_j = 0, \qquad \forall 0 \le j < k+1$$

且x_{k+1}是 f 在线性流形

$$S_k = \left\{ x = x_0 + \sum_{i=0}^{k} \beta_i d_i \,\middle|\, \beta_i \in R,\ i = 0, 1, 2, \cdots, k \right\}$$

中的极小点. 特别，$x_n = x^* = -Q^{-1}q$是问题(5.2)的唯一极小点.

该定理称为子空间扩展定理

定理表明共轭方向法具有二次终止性

定理的证明：分两步

先证第一部分：$\boxed{\nabla f(x_{k+1})^T d_j = 0, \forall j < k+1}$

若 $j = k$,由精确搜索,显然有：$\nabla f(x_{k+1})^{\mathrm{T}} d_k = 0$

$0 \le \forall j < k$,由于

$$\nabla f(x_{k+1}) = \nabla f(x_k) + \alpha_k Q d_k = \nabla f(x_{k-1}) + \alpha_k Q d_k + \alpha_{k-1} Q d_{k-1}$$

$$= \cdots\cdots$$

$$= \nabla f(x_{j+1}) + \sum_{i=j+1}^{k} \alpha_i Q d_i,$$

故由共轭性得

$$d_j^T \nabla f(x_{k+1}) = \underbrace{d_j^T \nabla f(x_{j+1})}_{\substack{0 \\ \text{精确搜索}}} + \sum_{i=j+1}^{k} \alpha_i \underbrace{d_j^T Q d_i}_{\substack{0 \\ \text{共轭性}}} = 0 + 0 = 0$$

x_{k+1}是 f 在线性流形

$$S_k = \{x = x_0 + \sum_{i=0}^{k} \beta_i d_i \mid \beta_i \in R, i = 0,1,2,\cdots,k\}$$

中的极小点.

再证第二部分

$\forall x \in S_k$ 且 $x \neq x_{k+1}$, 则

$$x = x_0 + \sum_{j=0}^{k} \beta_j d_j$$

由于 $$x_{k+1} = x_0 + \sum_{j=0}^{k} \alpha_j d_j$$

所以 $$x - x_{k+1} = \sum_{j=0}^{k} (\beta_j - \alpha_j) d_j$$

由泰勒公式得:

$$f(x) = f(x_{k+1}) + \nabla f(x_{k+1})^T (x - x_{k+1}) + \frac{1}{2}(x - x_{k+1})^T Q (x - x_{k+1})$$

$$= f(x_{k+1}) + \frac{1}{2}(x - x_{k+1})^T Q (x - x_{k+1}) > f(x_{k+1})$$

所以 x_{k+1} 是 f 在 S_k 上极小点

共轭方向的计算

共轭方向可用类似于 Gram - Schmit 正交化过程来产生

算法5.1 (Gram - Schmit 共轭化过程)

步0 给定线性无关向量组：$p_0, p_1, \cdots, p_{n-1} \in R^n$,

令 $d_0 = p_0, k := 0$;

步1 计算

$$d_{k+1} = p_{k+1} - \sum_{j=0}^{k} \frac{d_j^T Q p_{k+1}}{d_j^T Q p_j} d_j, \tag{5.5}$$

步2 若 $k = n - 2$, 则停止, 否则, 令 $k := k + 1$, 转步1.

容易验证, 由算法5.1产生的向量组 $d_0, d_1, \cdots, d_{n-1} \in R^n$

关于矩阵 Q 相互共轭

考虑 Ax=b

1 $$r_k = Ax_k - b.$$

为保证 p_k与 p_{k-1}共轭, 该如何选取β_k ?

3 $$\boxed{p_k = -r_k + \beta_k p_{k-1},}$$

上式两边同乘以 $p_{k-1}^T A$, 得到

$$\boxed{\beta_k = \frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}}.}$$ 2

采用下降迭代方法

5 $$\boxed{x_{k+1} = x_k + \alpha_k p_k,}$$

步长采用精确线搜索法，即

$$\boxed{\alpha_k = -\frac{r_k^T p_k}{p_k^T A p_k};}$$ 4

CG算法1

Given x_0;

Set $r_0 \leftarrow Ax_0 - b$, $p_0 \leftarrow -r_0$, $k \leftarrow 0$;

while $r_k \neq 0$

$$\alpha_k \leftarrow -\frac{r_k^T p_k}{p_k^T A p_k};$$

$$x_{k+1} \leftarrow x_k + \alpha_k p_k;$$

$$r_{k+1} \leftarrow Ax_{k+1} - b;$$

$$\beta_{k+1} \leftarrow \frac{r_{k+1}^T A p_k}{p_k^T A p_k};$$

$$p_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} p_k;$$

$$k \leftarrow k + 1;$$

end (while)

CG算法1的收敛性和性质

- 假设该算法第k步迭代得到的不是方程的解 x^*，则有下面性质，并且迭代序列 $\{x_k\}$ 最多n步收敛到解 x^*

$$
\begin{aligned}
r_k^T r_i &= 0, \qquad \text{for } i = 0, 1, \ldots, k-1, \\
\operatorname{span}\{r_0, r_1, \ldots, r_k\} &= \operatorname{span}\{r_0, Ar_0, \ldots, A^k r_0\}, \\
\operatorname{span}\{p_0, p_1, \ldots, p_k\} &= \operatorname{span}\{r_0, Ar_0, \ldots, A^k r_0\}, \\
p_k^T A p_i &= 0, \qquad \text{for } i = 0, 1, \ldots, k-1.
\end{aligned}
$$

CG算法2（实际问题常使用）

Algorithm 5.2 (CG).

Given x_0;

Set $r_0 \leftarrow Ax_0 - b$, $p_0 \leftarrow -r_0$, $k \leftarrow 0$;

while $r_k \neq 0$

$$\alpha_k \leftarrow \frac{r_k^T r_k}{p_k^T A p_k};$$

$$x_{k+1} \leftarrow x_k + \alpha_k p_k;$$

$$r_{k+1} \leftarrow r_k + \alpha_k A p_k;$$

$$\beta_{k+1} \leftarrow \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k};$$

$$p_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} p_k;$$

$$k \leftarrow k + 1;$$

end (while)

收敛性结果

- 如果矩阵A只有r个不同的特征值，那么CG迭代最多r步迭代到解。

- 如果A有特征值 $\lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$， 那么

$$\|x_{k+1} - x^*\|_A^2 \le \left(\frac{\lambda_{n-k} - \lambda_1}{\lambda_{n-k} + \lambda_1} \right)^2 \|x_0 - x^*\|_A^2.$$

$$\kappa(A) = \|A\|_2 \|A^{-1}\|_2 = \lambda_n / \lambda_1.$$

$$\|x_k - x^*\|_A \le 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k \|x_0 - x^*\|_A.$$

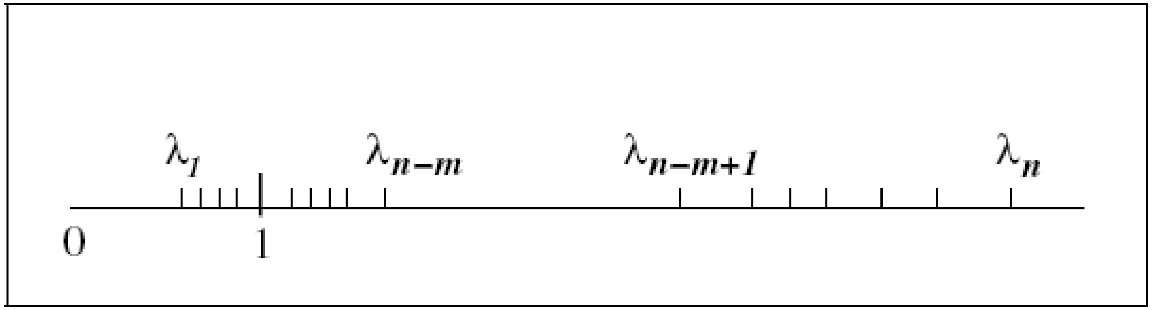
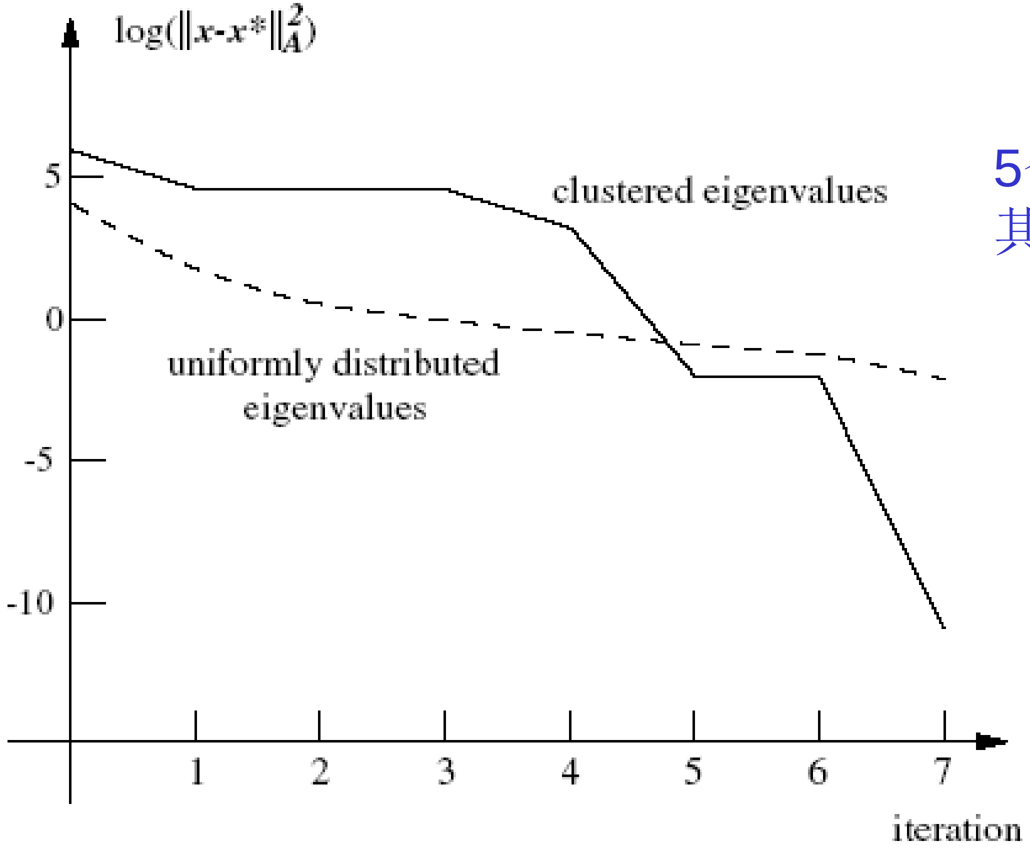


Figure 5.3 Two clusters of eigenvalues.



5个大的特征值
其余聚点为1

预处理

$$\hat{x} = Cx.$$

$$\hat{\phi}(\hat{x}) = \frac{1}{2}\hat{x}^T(C^{-T}AC^{-1})\hat{x} - (C^{-T}b)^T\hat{x}.$$

$$(C^{-T}AC^{-1})\hat{x} = C^{-T}b,$$

应用Algorithm 5.2解上面问题，再转化成x变量，得到预处理共轭梯度方法

预处理的目的，是使矩阵的条件数变小

Algorithm 5.3 (Preconditioned CG).

Given x_0, preconditioner M;

Set $r_0 \leftarrow Ax_0 - b$;

Solve $My_0 = r_0$ for y_0;

Set $p_0 = -y_0, k \leftarrow 0$;

while $r_k \neq 0$

$$\alpha_k \leftarrow \frac{r_k^T y_k}{p_k^T A p_k}; \tag{5.39a}$$

$$x_{k+1} \leftarrow x_k + \alpha_k p_k; \tag{5.39b}$$

$$r_{k+1} \leftarrow r_k + \alpha_k A p_k; \tag{5.39c}$$

$$\text{Solve } My_{k+1} = r_{k+1}; \tag{5.39d}$$

$$\beta_{k+1} \leftarrow \frac{r_{k+1}^T y_{k+1}}{r_k^T y_k}; \tag{5.39e}$$

$$p_{k+1} \leftarrow -y_{k+1} + \beta_{k+1} p_k; \tag{5.39f}$$

$$k \leftarrow k + 1; \tag{5.39g}$$

end (while)

总　　结：

1. 共轭梯度法的优点是不需要存储矩阵，只需存储矩阵乘向量，存储小, 计算简单, 适合求解大规模线性方程组和二次函数极小化问题.

2. 至多n步迭代收敛到最优解。预处理可以加速

3. 采用精确搜索，　初始方向必须选择最速下降方向.

作 业

- 习题5（李董辉书）

  2, 4

- 推导《Numerical Optimization的Algorithm 5.3
- 编程练习：Exercise 5.1. 实现Algorithm5.2