

### 3.6 共轭梯度法 (算法)

共轭方向法首先要求确定初始点  $\mathbf{x}_0$  和一组共轭方向  $\mathbf{d}_0$ ,  $\mathbf{d}_1$ , …,  $\mathbf{d}_{n-1}$ , 沿着方向  $\mathbf{d}_0$  使得 q(x) 达到最小,从而获得 $x_1$ 。以此类推,沿着方向d,使得q(x)最小,获得 $x_2$ 。最后,沿 着方向  $\mathbf{d}_{n-1}$  可获得  $\mathbf{x}_n$ , 这就是函数的极小点  $q(\mathbf{x})$ 。这意味着能够通过 n 次搜索获得二次型 函数的极小点。该方法的关键就是如何获得共轭方向,在接下来介绍的算法中,利用  $q(\mathbf{x})$ 的梯度来产生共轭方向。

基本的共轭方向算法。给定初始点  $x^{(0)}$  和一组关于 Q 共轭的方向  $d^{(0)}$ ,  $d^{(1)}$ , …,  $d^{(n-1)}$ , 迭代公式为( $k \ge 0$  表示迭代次数)

$$\begin{split} \boldsymbol{g}^{(k)} &= \nabla f(\boldsymbol{x}^{(k)}) = \boldsymbol{Q} \boldsymbol{x}^{(k)} - \boldsymbol{b} \\ \boldsymbol{\alpha}_k &= -\frac{\boldsymbol{g}^{(k)\top} \boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top} \boldsymbol{Q} \boldsymbol{d}^{(k)}} \\ \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)} \end{split}$$

### 明理,精工,笃学,致远

# 3.6 共轭梯度法(证明)

证明: 由于方向 $d^{(i)}$ ,  $i=0,1,\dots,n-1$ 线性无关,因此, $x^*-x^{(0)} \in \mathbb{R}^n$  可由它们线性 表出,即

$$m{x}^*-m{x}^{(0)}=eta_0m{d}^{(0)}+\cdots+eta_{n-1}m{d}^{(n-1)}$$
 其中, $eta_i$ , $i=0$ , $1$ , $\cdots$ , $n-1$  为常数。

$$x^* - x^{(0)} = \beta_0 d^{(0)} + \dots + \beta_{n-1} d^{(n-1)}$$

$$x^{(n)} - x^{(0)} = \alpha_0 d^{(0)} + \dots + \alpha_{n-1} d^{(n-1)}$$

明理,精工,笃学,致远

## 3.6 共轭梯度法(证明)

$$f(\boldsymbol{x}) = \frac{1}{2} \boldsymbol{x}^{\top} \boldsymbol{Q} \boldsymbol{x} - \boldsymbol{x}^{\top} \boldsymbol{b}$$

其中,  $Q = Q^{\top} > 0$ ,  $x \in \mathbb{R}^n$ 。注意, 由于 Q > 0, 因此函数 f 有一个全局极小点, 可通过求解 Qx = b 得到。

定理 10.1 对于任意初始点  $x^{(0)}$  , 基本的共轭方向算法都能在 n 次迭代之内收敛到 唯一的全局极小点 $x^*$ , 即 $x^{(n)} = x^*$ 。

证明: 由于方向 $\mathbf{d}^{(i)}$ ,  $i=0,1,\cdots,n-1$ 线性无关,因此, $\mathbf{x}^*-\mathbf{x}^{(0)} \in \mathbb{R}^n$  可由它们线性

$$\boldsymbol{x}^* - \boldsymbol{x}^{(0)} = \beta_0 \boldsymbol{d}^{(0)} + \dots + \beta_{n-1} \boldsymbol{d}^{(n-1)}$$

其中,  $\beta_i$ ,  $i=0,1,\dots,n-1$  为常数。

### 明理,精工,笃学,致远

### 3.6 共轭梯度法 (证明)

证明: 由于方向 $d^{(i)}$ ,  $i=0,1,\dots,n-1$ 线性无关,因此, $x^*-x^{(0)} \in \mathbb{R}^n$  可由它们线性 表出。即

$$x^* - x^{(0)} = \beta_0 d^{(0)} + \cdots + \beta_{n-1} d^{(n-1)}$$

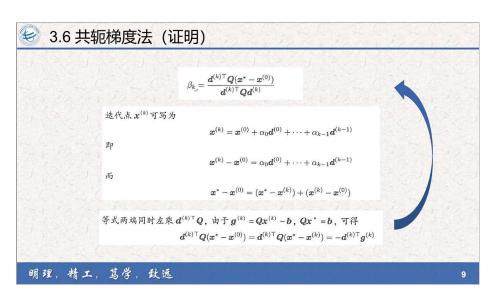
其中,  $\beta_i$ ,  $i=0,1,\dots,n-1$  为常数。

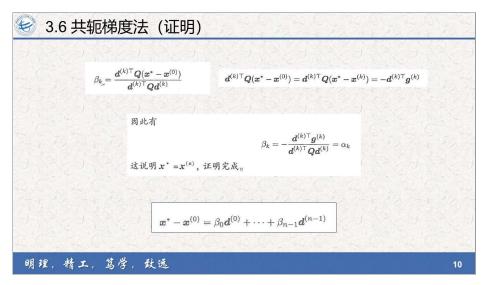
在等号两端同时左乘  $\mathbf{d}^{(k)\top}\mathbf{Q}$ ,  $0 \le k < n$ , 由  $\mathbf{Q}$  共轭的性质可知.  $\mathbf{d}^{(k)\top}\mathbf{Q}\mathbf{d}^{(i)} = 0$ ,  $k \ne i$ . 因此有

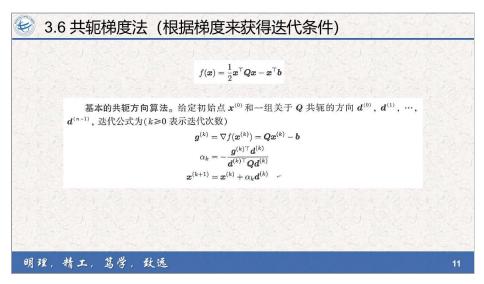
$$\boldsymbol{d}^{(k)\top}\boldsymbol{Q}(\boldsymbol{x}^* - \boldsymbol{x}^{(0)}) = \beta_k \boldsymbol{d}^{(k)\top}\boldsymbol{Q}\boldsymbol{d}^{(k)}$$

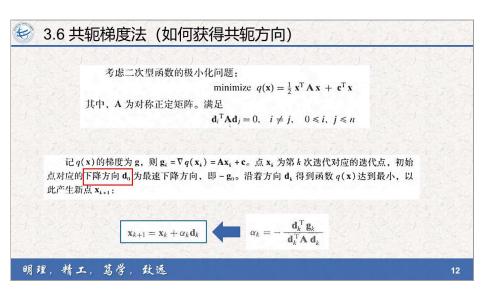
整理后,可得

$$\beta_{k} = \frac{d^{(k)\top}Q(x^* - x^{(0)})}{d^{(k)\top}Qd^{(k)}}$$











## 3.6 共轭梯度法(如何获得共轭方向)



按照微分法则,由  $dq(\alpha)/d\alpha=0$  可推导出

$$\frac{d\mathbf{q}(\alpha)}{d\mathbf{k}^T\mathbf{g}_{k+1}} = 0 \qquad \frac{d\mathbf{q}(\alpha)}{d\alpha} = \nabla f(\mathbf{x}_k + \alpha \, \mathbf{d}_k) \cdot \mathbf{d}_k = \mathbf{g}_{k+1} \cdot \mathbf{d}_k = 0$$

接下来就是最关键的步骤了,按照下式确定下降方向 d,:::

$$\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_k \mathbf{d}_k$$

上式的含义为对最速下降方向 - gkal 进行适当的"纠偏",如图 3.6 所示。由于要求  $\mathbf{d}_{i+1}$ 和  $\mathbf{d}_{i}$  关于 A 共轭,即  $\mathbf{d}_{i+1}^{\mathsf{T}}\mathbf{A}\mathbf{d}_{i}=0$  由此可得

$$-\mathbf{g}_{k+1}^{\mathrm{T}} \mathbf{A} \mathbf{d}_{k} + \beta_{k} \mathbf{d}_{k}^{\mathrm{T}} \mathbf{A} \mathbf{d}_{k} = 0$$

### 明理,精工,笃学,致远

## 3.6 共轭梯度法 (如何获得共轭方向)

上式的含义为对最速下降方向  $-g_{k+1}$ 进行适当的"纠偏",如图 3.6 所示。由于要求  $\mathbf{d}_{k+1}$ 和  $\mathbf{d}_k$  关于 A 共轭,即  $\mathbf{d}_{k+1}^{\mathsf{T}}$  A  $\mathbf{d}_k$  = 0 由此可得

$$-\mathbf{g}_{k+1}^{\mathrm{T}} \mathbf{A} \mathbf{d}_k + \beta_k \mathbf{d}_k^{\mathrm{T}} \mathbf{A} \mathbf{d}_k = 0$$

$$\beta_k = \frac{\boldsymbol{g}_{k+1}^T \boldsymbol{A} \boldsymbol{d}_k}{\boldsymbol{d}_k^T \boldsymbol{A} \boldsymbol{d}_k}$$

$$\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_k \mathbf{d}_k$$

$$\begin{aligned} \boldsymbol{g}^{(k)} &= \nabla f(\boldsymbol{x}^{(k)}) = \boldsymbol{Q} \boldsymbol{x}^{(k)} - \boldsymbol{b} \\ \alpha_k &= -\frac{\boldsymbol{g}^{(k)\top} \boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)\top} \boldsymbol{Q} \boldsymbol{d}^{(k)}} \end{aligned}$$

明理,精工,笃学,致远

### 3.6 共轭梯度法 (一般性)

对于一般形式函数的极小点求解也非常有效

$$\beta_k = \frac{\boldsymbol{g}_{k+1}^T \boldsymbol{A} \boldsymbol{d}_k}{\boldsymbol{d}_k^T \boldsymbol{A} \boldsymbol{d}_k}$$

由式 (3.20) 可得,  $\mathbf{d}_k = (\mathbf{x}_{k+1} - \mathbf{x}_k)/\alpha_k$ , 因此,  $\mathbf{A}\mathbf{d}_k = (\mathbf{g}_{k+1} - \mathbf{g}_k)/\alpha_k$ , 代入上式并整 理后可得

$$\beta_k = \frac{\boldsymbol{g}_{k+1}^T (\boldsymbol{g}_{k+1} - \boldsymbol{g}_k)}{\alpha_k \boldsymbol{d}_k^T A \boldsymbol{d}_k}$$

明理,精工,笃学,致远

15

# 3.6 共轭梯度法 (一般性)

$$\beta_k = \frac{\boldsymbol{g}_{k+1}^T (\boldsymbol{g}_{k+1} - \boldsymbol{g}_k)}{\alpha_k \boldsymbol{d}_k^T A \boldsymbol{d}_k} \qquad \alpha_k = \frac{-\boldsymbol{d}_k^T \boldsymbol{g}_k}{\boldsymbol{d}_k^T A \boldsymbol{d}_k}$$

$$d_{k}^{T} g_{k+1} = 0$$

$$d_{k+1}^{T} = -g_{k+1} + \beta_{k} d_{k}$$

$$d_{k+1}^{T} g_{k+1} = -g_{k+1}^{T} g_{k+1} + \beta_{k} d_{k}^{T} g_{k+1}$$

### 😂 3.6 共轭梯度法 (一般性)

$$d_{k}^{T} \boldsymbol{g}_{k+1} = 0$$

$$d_{k+1}^{T} = -\boldsymbol{g}_{k+1}^{T} + \beta_{k} d_{k}$$

$$d_{k+1}^{T} \boldsymbol{g}_{k+1} = -\boldsymbol{g}_{k+1}^{T} \boldsymbol{g}_{k+1} + \beta_{k} d_{k}^{T} \boldsymbol{g}_{k+1}$$

$$d_{k+1}^T g_{k+1} = -g_{k+1}^T g_{k+1}$$
  $d_k^T g_k = -g_k^T g_k$ 

$$\alpha_k = \frac{-\boldsymbol{d}_k^T \boldsymbol{g}_k}{\boldsymbol{d}_k^T A \boldsymbol{d}_k} \qquad \qquad \boldsymbol{\alpha}_k = \frac{\boldsymbol{g}_k^T \boldsymbol{g}_k}{\boldsymbol{d}_k^T A \boldsymbol{d}_k}$$

明理,精工,笃学,致远

3.6 共轭梯度法 (一般性)

$$\beta_k = \frac{\boldsymbol{g}_{k+1}^T (\boldsymbol{g}_{k+1} - \boldsymbol{g}_k)}{\boldsymbol{g}_k^T \boldsymbol{g}_k} \qquad \qquad \beta_k = \frac{\boldsymbol{g}_{k+1}^T \boldsymbol{g}_{k+1}}{\boldsymbol{g}_k^T \boldsymbol{g}_k}$$

$$\mathbf{g}_{k+1}^{T} \mathbf{g}_{k} = \mathbf{g}_{k+1}^{T} (-\mathbf{d}_{k} + \beta_{k-1} \mathbf{d}_{k-1}) = \beta_{k-1} \mathbf{g}_{k+1}^{T} \mathbf{d}_{k-1} \left[ \mathbf{A} \mathbf{d}_{k} = (\mathbf{g}_{k+1} - \mathbf{g}_{k}) / \alpha_{k} \right] \\
= \beta_{k-1} (\mathbf{g}_{k}^{T} + \alpha_{k} \mathbf{d}_{k}^{T} \mathbf{A}) \mathbf{d}_{k-1} \\
= 0$$

这称为 Fletcher-Reeves 算法 [Fletcher and Reeves, 1964]

明理,精工,笃学,致远

### 3.6 共轭梯度法 (一般性)

$$\alpha_k = \frac{\boldsymbol{g}_k^T \boldsymbol{g}_k}{\boldsymbol{d}_k^T \boldsymbol{A} \boldsymbol{d}_k} \qquad \qquad \beta_k = \frac{\boldsymbol{g}_{k+1}^T (\boldsymbol{g}_{k+1} - \boldsymbol{g}_k)}{\alpha_k \boldsymbol{d}_k^T \boldsymbol{A} \boldsymbol{d}_k}$$

$$\beta_k = \frac{\boldsymbol{g}_{k+1}^T (\boldsymbol{g}_{k+1} - \boldsymbol{g}_k)}{\boldsymbol{g}_k^T \boldsymbol{g}_k}$$

由式 (3.32) 可以看出、 $\beta$ 。与矩阵 A 和向量 c 无关、这意味着该算法也适用于非二次 型函数的极小点求解问题。但是,对于非二次型函数,式(3.31)所示的步长公式就不再 适用了,必须采用数值线性搜索方法来求解步长  $\alpha_k$ 。当然,通过n步迭代就可以得到最优 解这一特性只适用于二次型函数。对于非二次型函数,应该每 n 次迭代完成之后,都需要将 下降方向重置为当前梯度的负方向,即最速下降方向。采用式 (3.32) 来计算  $\beta_{\epsilon}$  的共轭梯 度法称为 Polak - Rebiere 算法。

明理,精工,笃学,致远

## 3.6 共轭梯度法 (例题)

例 3.9 函数为  $f=x_1^2+4x_2^2$ , 初始点为  $\mathbf{x}_0=[1,1]^T$ 。利用共轭梯度法开展两次迭代,即 可得到极小点。第一次迭代对应的下降方向为最速下降方向、即

$$\mathbf{d}_0 = -\nabla f(\mathbf{x}_0) = -[2, 8]^{\mathrm{T}}$$

此处未将方向向量进行归一化处理、不影响后续计算。求函数

$$f(\alpha) = f(\mathbf{x}_0 + \alpha \mathbf{d}_0) = (1 - 2\alpha)^2 + 4(1 - 8\alpha)^2$$

的极小点,可得  $\alpha_0 = 0.1307692$ ,构造新点为  $\mathbf{x}_1 = \mathbf{x}_0 + \alpha_0 \mathbf{d}_0 = [0.7384615, -0.0461538]^{\mathsf{T}}$ 。

$$\beta_0 = \frac{\|\nabla f(\mathbf{x}^1)\|^2}{\|\nabla f(\mathbf{x}^0)\|^2} = 2.3176/68 = 0.034\ 082\ 8$$

$$\mathbf{d}_1 = -\nabla f^{\mathrm{T}}(\mathbf{x}_1) + \beta_0 \mathbf{d}_0 = \begin{pmatrix} -1.476 & 923 \\ 0.369 & 231 \end{pmatrix} + 0.034 & 082 & 8 & \begin{pmatrix} -2 \\ -8 \end{pmatrix} = \begin{pmatrix} -1.545 & 08 \\ 0.096 & 56 \end{pmatrix}$$

