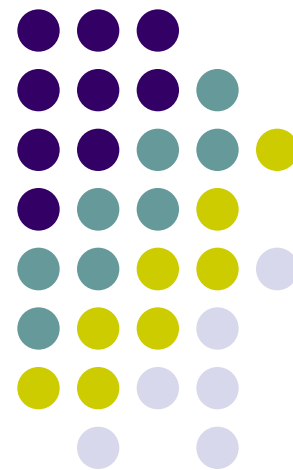


# 最优化方法 (IV)

张宏鑫，华炜

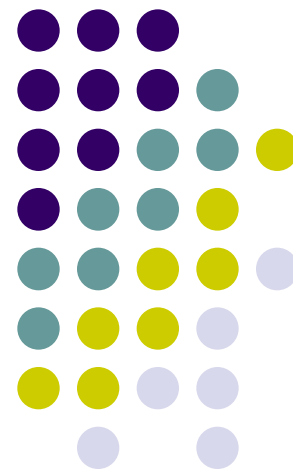
2011-06-16

浙江大学计算机学院



# 1. 约束非线性最优化

---





# 约束优化最优性条件

约束最优化问题通常写为

$$\min f(\mathbf{x})$$

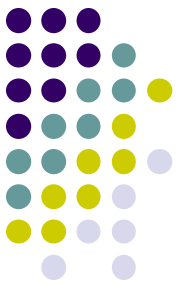
$$\text{s.t. } c_i(\mathbf{x})=0, i \in E=\{1, \dots, m_e\},$$

$$c_i(\mathbf{x}) \geq 0, i \in I=\{m_e+1, \dots, m\}$$

在 $\mathbf{x}^*$ 处的**非积极约束**

设 $\mathbf{x}^*$ 为一个局部极小点，若不等式约束 $i_0$ 有,  $c_{i_0}(\mathbf{x}^*) > 0$ , 则可将第 $i_0$ 个约束去掉，且 $\mathbf{x}^*$ 仍然是去掉第 $i_0$ 个约束条件的问题的局部极小点。称约束 $c_{i_0}$ 在 $\mathbf{x}^*$ 处是非积极的。

定义:  $I(\mathbf{x}) = \{i \mid c_i(\mathbf{x}) \leq 0, i \in I\}$ ;  $A(\mathbf{x}) = E \cup I(\mathbf{x})$  为 $\mathbf{x}$ 点处的**积极集合**。



# 一阶最优性条件

Kuhn - Tucker必要条件:

若 $x^*$ 是问题 $P$ 的一个局部极小点, 如果 $\nabla c_i(x^*) (i \in E \cup I(x^*))$ 线性无关, 则必存在 $\lambda_i^* (i = 1, \dots, m)$ , 使得

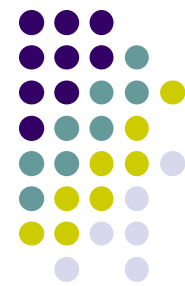
$$\nabla f(x^*) = \sum_{i=1}^m \lambda_i^* \nabla c_i(x^*), \quad (*)$$

$$\lambda_i^* \geq 0, \quad \lambda_i^* c_i(x^*) = 0, i \in I. \quad (**)$$

满足上述两式的点称为K-T点。与该定理联系密切的是Lagrange函数:

$$L(x, \lambda) = f(x) - \lambda^T c(x).$$

则(\*)条件等价于 $\nabla_x L = 0$ 。 $\lambda$ 称为Lagrange乘子。



# 二阶必要条件

定义：设 $x^*$ 是 $K-T$ 点， $\lambda^*$ 称为相应的Lagrange乘子，若存在序列 $\{d_k\}$ 和 $\{\delta_k > 0\}$ 使得

$$x^* + \delta_k d_k \in X$$

$$\sum_{i=1}^m \lambda_i^* c_i(x^* + \delta_k d_k) = 0, i \in I.$$

且有 $d_k \rightarrow d$ ， $\delta_k \rightarrow 0$ ，则称 $d$ 为 $x^*$ 处的序列零约束方向。在 $x^*$ 处的所有序列零约束方向的集合记为 $S(x^*, \lambda^*)$ 。

二阶必要性条件：

设 $x^*$ 为局部极小点， $\lambda^*$ 称为相应的Lagrange乘子，则必有

$$d^T \nabla_{xx}^2 L(x^*, \lambda^*) d \geq 0, \forall d \in S(x^*, \lambda^*). \quad \text{其中 } L(x, \lambda) \text{ 为Lagrange函数。}$$

稍加强可得充分性条件：

设 $x^*$ 为 $K-T$ 点， $\lambda^*$ 称为相应的Lagrange乘子，若

$$d^T \nabla_{xx}^2 L(x^*, \lambda^*) d > 0, \forall 0 \neq d \in S(x^*, \lambda^*), \quad \text{则 } x^* \text{ 为局部严格极小点。}$$



# 可行方向法

- 可行方向法即要去每次迭代产生的点 $\mathbf{x}_k$ 都是约束优化问题的可行点。
- 关键在于每一步寻找可行下降方向：

$$\mathbf{d}^T \nabla f(\mathbf{x}_k) < 0, \mathbf{d} \in FD(\mathbf{x}_k, X).$$

可行方向：设 $\bar{x} \in X, 0 \neq d \in R^n$ , 如存在 $\delta > 0$ 使得 $\bar{x} + td \in X$ , 则称 $d$ 为 $\bar{x}$ 处的可行方向。 $X$ 在 $\bar{x}$ 处的所有可行方向集合记为 $FD(\bar{x}, X)$ 。



# 变量消去法

考虑等式约束问题

$$\begin{aligned} \min f(\mathbf{x}), \\ \text{s.t. } c(\mathbf{x}) = 0 \end{aligned}$$

设有变量分解  $\mathbf{x} = \begin{bmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{bmatrix}$ , 其中  $x_B \in R^m$ ,  $x_N \in R^{n-m}$ .

则  $c(x) = 0$  可改写为

$$c(x_B, x_N) = 0. \quad (1)$$

假定可以从(1)解出  $x_B = \phi(x_N)$ , 则原问题等价于

$$\min_{x_N \in R^{n-m}} f(x_B, x_N) = f(\phi(x_N), x_N) = \tilde{f}(x_N).$$

称  $\tilde{g}(x_N) = \nabla_{x_N} \tilde{f}(x_N)$  为既约梯度。



# 变量消去法

不难验证

$$\tilde{g}(x_N) = \frac{\partial f(x_B, x_N)}{\partial x_N} + \frac{\partial x_B^T}{\partial x_N} \frac{\partial f(x_B, x_N)}{\partial x_B},$$

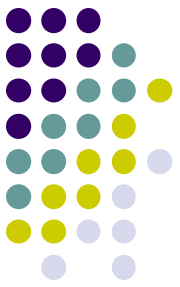
从(1)可得

$$\frac{\partial x_B^T}{\partial x_N} \frac{\partial c(x_B, x_N)^T}{\partial x_B} + \frac{\partial c(x_B, x_N)^T}{\partial x_N} = 0.$$

假设  $\frac{\partial c(x_B, x_N)^T}{\partial x_B}$  非奇异, 可得到

$$\tilde{g}(x_N) = \frac{\partial f(x_B, x_N)}{\partial x_N} - \frac{\partial c(x_B, x_N)^T}{\partial x_N} \left( \frac{\partial c(x_B, x_N)^T}{\partial x_B} \right)^{-1} \frac{\partial f(x_B, x_N)}{\partial x_B}$$





# 变量消去法

令  $\lambda = \left( \frac{\partial c(x_B, x_N)^T}{\partial x_B} \right)^{-1} \frac{\partial f(x_B, x_N)}{\partial x_B}$ , 则发现可以将

既约梯度写成  $L \arg range$  函数在既约空间上的梯度

$$\tilde{g}(x_N) = \frac{\partial}{\partial x_N} [f(x) - \lambda^T c(x)]$$

或者说, 有

$$\nabla_x L(x, \lambda) = \begin{bmatrix} 0 \\ \tilde{g}(x_N) \end{bmatrix}$$



# 变量消去法

故既约梯度可看作Lagrange函数之梯度的非零部分。

利用既约梯度，可以构造无约束优化问题的线搜索方向。

例如可取最速下降方向

$$\tilde{d}_k = -\tilde{g}\left(\left(x_N\right)_k\right),$$

或拟牛顿方向

$$\tilde{d}_k = -H_k \tilde{g}\left(\left(x_N\right)_k\right)。$$

在无约束问题上作线性搜索，等价于对原目标函数

$f(x)$ 在曲线

$$c\left(x_B, \left(x_N\right)_k + \alpha \tilde{d}_k\right) = 0 \quad (2)$$

上作曲线搜索。因为 $\phi(x)$ 的解析表达式并不知道，故

作一维搜索时，每个试探步长 $\alpha > 0$ 都要用(2)来求解 $x_B$ 。



# 变量消去法

1. 给出可行点 $x_1, \varepsilon \geq 0, k = 1$

2. 计算 $\frac{\partial c(x_k)^T}{\partial x} = \begin{bmatrix} A_B \\ A_N \end{bmatrix}$ , 其中划分使得 $A_B$ 非奇异;

计算 $\lambda = \left( \frac{\partial c(x_B, x_N)^T}{\partial x_B} \right)^{-1} \frac{\partial f(x_B, x_N)}{\partial x_B}$  及 $\tilde{g}((x_N)_k) = \frac{\partial}{\partial x_N} [f(x) - \lambda^T c(x)]$ 。

3. 如果 $\|\tilde{g}_k\| \leq \varepsilon$ , 则停止; 否则利用某种方式产生下降方向 $\tilde{d}_k$ , 即使得

$$\tilde{d}_k \tilde{g}_k < 0;$$

4.  $\min_{\alpha \geq 0} f(\phi((x_N)_k + \alpha \tilde{d}_k), (x_N)_k + \alpha \tilde{d}_k)$  进行线性搜索给出 $\alpha_k > 0$ , 令

$$x_{k+1} = (\phi((x_N)_k + \alpha_k \tilde{d}_k), (x_N)_k + \alpha_k \tilde{d}_k), k = k + 1,$$

转2步。



# 广义消去法

考虑更一般的形式。任意非奇异矩阵  $S \in R^{n \times n}$ ，以及变量替换  $x = Sw$

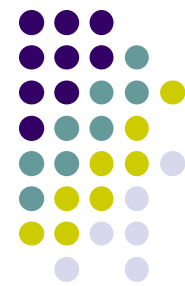
对  $w$  进行变量分离  $w = \begin{bmatrix} w_B \\ w_N \end{bmatrix}$ 。利用约束条件  $c(x) = 0$  进行变量消去

得到  $w_B = \phi(w_N)$ 。于是原问题等价于

$$\min_{w_N \in R^{n-m}} f(S_B w_B + S_N w_N) = \tilde{f}(w_N)$$

可直接计算得到  $\nabla_{w_n} \tilde{f}(w_N) = \tilde{g}((x_N)_k) = (S_k)_N \left[ \nabla f(x) - \nabla c(x)^T \lambda \right]$ ,

其中  $\lambda$  满足  $(S_k)_B \left[ \nabla f(x) - \nabla c(x)^T \lambda \right] = 0$ 。



# 广义消去法

1. 给出可行点 $x_1, \varepsilon \geq 0, k = 1$

2. 以某种方式构造一非奇异矩阵 $S_k$ , 且有划分 $S_k = \begin{bmatrix} (S_k)_B & (S_k)_N \end{bmatrix}$ , 使得 $(S_k)_B^T \frac{\partial c(x_k)^T}{\partial x}$  非奇异;

根据 $(S_k)_B \left[ \nabla f(x) - \nabla c(x)^T \lambda \right] = 0$  计算 $\lambda$ , 以及计算 $\tilde{g}((x_N)_k) = (S_k)_N \left[ \nabla f(x) - \nabla c(x)^T \lambda \right]$ 。

3. 如果 $\|\tilde{g}_k\| \leq \varepsilon$ , 则停止; 否则利用某种方式产生下降方向 $\tilde{d}_k$ , 即使得

$$\tilde{d}_k \tilde{g}_k < 0;$$

4.  $\min_{\alpha \geq 0} f\left((S_k)_B \phi\left((w_k)_N + \alpha \tilde{d}_k\right), (S_k)_B \left((x_k)_N + \alpha \tilde{d}_k\right)\right)$  进行线性搜索给出 $\alpha_k > 0$ , 令

$$x_{k+1} = \left( (S_k)_B \phi\left((w_k)_N + \alpha_k \tilde{d}_k\right), (S_k)_B \left((x_k)_N + \alpha_k \tilde{d}_k\right) \right), k = k + 1,$$

转2步。

若 $S_k = I$ , 广义消去法就是变量消去法。



# 投影梯度法

广义消去法每次迭代的变量增量 $x_{k+1} - x_k$ 由两部分组成,

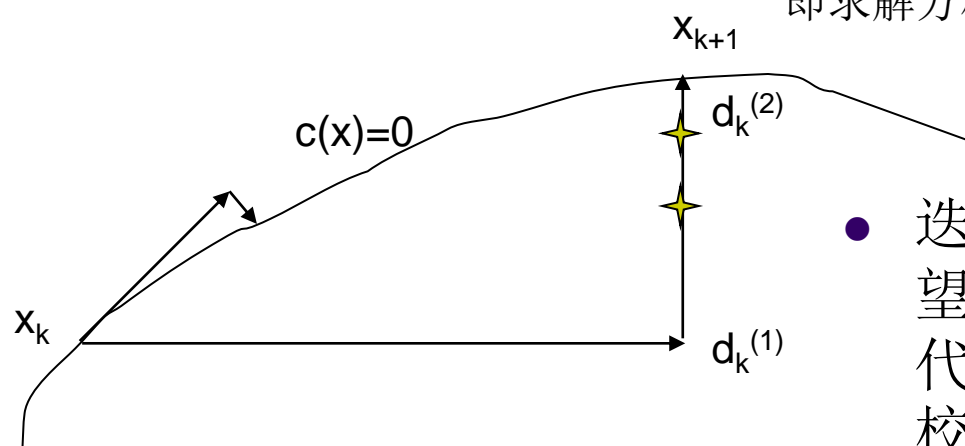
$x_{k+1} = x_k + d_k^{(1)} + d_k^{(2)}$ , 其中

$$d_k^{(1)} = \alpha_k (S_k)_N \tilde{d}_k, d_k^{(2)} = (S_k)_B \left[ \varphi \left( (w_k)_N + \alpha_k \tilde{d}_k \right) - (w_k)_B \right]$$

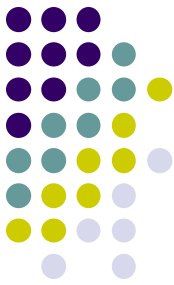
迭代过程是先得到 $d_k^{(1)}$ , 再在 $d_k^{(2)}$ 方向上迭代得到正确的步长。

在 $d_k^{(2)}$ 方向上迭代的过程就是利用 $c(x) = 0$ 求解 $w_B$ 的过程。

即求解方程 $c \left( (S_k)_B w_B + (S_k)_N \left[ (w_k)_N + \alpha \tilde{d}_k \right] \right) = 0$



- 迭代方式存在不合理之处。本来希望迭代点都在可行域上, 但具体迭代过程却是先远离可行域, 然后再校正回可行域中。
- 希望离开程度尽可能小? 沿线性化方向。迭代过程可以更快的收敛。



线性化可行方向:

设 $x^* \in$ 可行域 $X, d \in R^n$

$$d^T \nabla c_i(x^*) = 0, i \in E; d^T \nabla c_i(x^*) \geq 0, i \in I(x^*)$$

则 $d$ 为 $X$ 的 $x^*$ 处的线性化可行方向。

为使 $d_k^{(1)}$ 沿线性化可行方向, 应选取 $S_k$ 使得 $(S_k)_N^T \nabla c(x_k)^T = 0$ .

设 $A_k = \nabla c(x_k)^T$ , 则由 $(S_k)_N$ 的列向量所张成的子空间就是 $A_k^T$ 的零空间。

而另一方面,  $P = I - A_k(A_k^T A_k)^{-1} A_k^T$ 为到该零空间的投影算子。

当在广义消去法中用最速下降法时, 可得 $d_k^{(1)} = -\alpha_k (S_k)_N (S_k)_N^T \nabla f(x_k)$ ,

因为 $d_k^{(1)}$ 为线性可行方向, 故 $d_k^{(1)}$ 在 $A_k^T$ 的零空间中, 进而可知 $(S_k)_N (S_k)_N^T$ 为投影算子 $P$ 。

在计算中, 可利用 $A_k$ 的 $QR$ 分解,

$$A_k = QR = \begin{bmatrix} Y_k & Z_k \end{bmatrix} \begin{bmatrix} R_k \\ 0 \end{bmatrix}, A_k \in R^{n \times m}, Q \text{ 为正交阵}, Y_k \in R^{n \times m}, Z_k \in R^{n \times (n-m)},$$

$R_k \in R^{m \times m}$ 可逆上三角矩阵。可取 $(S_k)_N = Z_k, (S_k)_B = I$ 。

若将搜索方向换成 $d_k = -Z_k z_k, z_k \in R^{n-m}$ , 且满足 $z_k^T \tilde{g}_k < 0$ , 则算法是一个一般形式的线性化可行方向法, 简称可行方向法。



# 罚函数法

- 利用目标函数和约束函数构造具有“罚性质”的函数

$$P(\mathbf{x}) = P(f(\mathbf{x}), c(\mathbf{x}))$$

所谓的罚性质，即要求对于可行点 $P(\mathbf{x})=f(\mathbf{x})$ ，当约束条件破坏很大时， $P(\mathbf{x}) \gg f(\mathbf{x})$

定义约束违反度函数：

$$C^{(-)}(\mathbf{x}) = (c_1^{(-)}(\mathbf{x}), \dots, c_m^{(-)}(\mathbf{x})),$$

其中：

$$c_i^{(-)}(\mathbf{x}) = c_i(\mathbf{x}), \quad i=1, \dots, m_e;$$

$$c_i^{(-)}(\mathbf{x}) = \min\{c_i(\mathbf{x}), 0\}, \quad i=m_e+1, \dots, m$$





# 罚函数法

罚函数一般可表示为目标函数与一项与 $c(x)$ 有关的“罚项”之和，即

$$P(x) = f(x) + h(c^{(-)}(x))$$

罚项是定义在 $R^m$ 上的函数，满足 $h(0) = 0, \lim_{\|c\| \rightarrow \infty} h(c) = +\infty$

最早的罚函数是*Courant*罚函数，定义如下

$$P(x) = f(x) + \sigma \|c^{(-)}(x)\|_2^2, \sigma > 0 \text{ 是一正常数, 罚因子。}$$

事实上，更一般的可定义为

$$P(x) = f(x) + \sigma \|c^{(-)}(x)\|^\alpha, \alpha > 0.$$



# 罚函数法

一般定义为

$$P(x) = f(x) + \sigma \|c^{(-)}(x)\|^\alpha, \quad \alpha > 0.$$

若罚函数在可行域边界上取值为无穷，则称为内点罚函数。

内点罚函数仅适合不等式约束问题。

常见有倒数罚函数和对数罚函数：

$$P(x) = f(x) + \sigma^{-1} \sum_{i=1}^m \frac{1}{c_i(x)}, \quad P(x) = f(x) + \sigma^{-1} \sum_{i=1}^m \log c_i(x)$$

内点罚函数在可行域的边界上形成一堵无穷高的“障碍墙”，所以也称为障碍罚函数。

罚函数法的基本点是：

每次迭代（求解一个无约束优化问题）增加罚函数因子，直到使 $\|c^{(-)}(x)\|$ 缩小到给定误差。



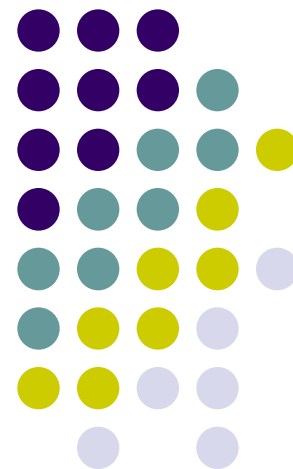
# 罚函数法

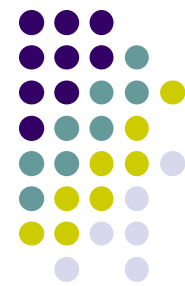
设 $x^*$ 为原问题的 $K-T$ 点, 而 $x^*$ 一般不是Courant函数的稳定点。为了克服这一缺点, 引入参数 $\theta=(\theta_1, \dots, \theta_m)$ , 其中 $\theta_i \geq 0 (i = m_e + 1, \dots, m)$ 。则

$$P(x) = f(x) + \sum_{i=1}^{m_e} \left[ -\lambda_i c_i(x) + \frac{1}{2} \sigma_i (c_i(x))^2 \right] +$$
$$\sum_{i=m_e+1}^m \begin{cases} -\lambda_i c_i(x) + \frac{1}{2} \sigma_i (c_i(x))^2, & \text{如 } c_i(x) < \lambda_i / \sigma_i \\ -\frac{1}{2} \lambda_i^2 / \sigma_i, & \text{否则} \end{cases}$$

其中 $\lambda_i = \sigma_i \theta_i$

## 2. 二次规划





# 非线性最优化

- 最优化的一般形式为

$$\text{Min } f(\mathbf{x}) \text{ s.t. } \mathbf{x} \in X$$

$f(\mathbf{x})$ 为目标函数， $X \subset E^n$ 为可行域。

如 $X = E^n$ ，则以上最优化问题为**无约束最优化**问题。

**约束最优化**问题通常写为

$$\text{Min } f(\mathbf{x})$$

$$\text{s.t. } c_i(\mathbf{x}) = 0, i \in E,$$

$$c_i(\mathbf{x}) \geq 0, i \in I,$$

其中 $E, I$ 分别为等式约束的指标集和不等式约束的指标集， $c_i(\mathbf{x})$ 是约束函数。



# 非线性优化中的概念

- 可行点，可行域
- 极小，全局极小（总体极小点），全局严格极小，局部极小，局部严格极小
- 积极与非积极，积极约束，非积极约束
- 可行方向集，线性可行方向集，序列可行方向集
  
- Farkas引理与K-T条件
  
- 以上参见《最优化理论与方法》第八章



# 无约束二次最优化

$$\min f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{c}^T \mathbf{x}, \mathbf{x} \in R^n$$

H是对称阵

基本解法：求导然后找局部极值。



# 二次规划的一般形式

$$\min f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{c}^T \mathbf{x}, \mathbf{x} \in R^n$$

$$\text{s.t. } A\mathbf{x} \leq \mathbf{b}$$

(1)

- 当 $H$ 为对称矩阵时，被称为二次规划(Quadratic Programming，记作QP)。
- 特别，当 $H$ 正定时，目标函数为凸函数，线性约束下可行域又是凸集。上式被称为凸二次规划。





# 二次规划的一般形式

$$\begin{aligned} \min f(\mathbf{x}) &= \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{c}^T \mathbf{x}, \mathbf{x} \in R^n \\ \text{s.t. } \mathbf{a}_i^T \mathbf{x} &\geq b_i, i \in I, \\ \mathbf{a}_i^T \mathbf{x} &= b_i, i \in E. \end{aligned} \tag{1}$$

- 当 $H$ 为对称矩阵时，被称为二次规划(Quadratic Programming，记作QP)。
- 特别，当 $H$ 正定时，目标函数为凸函数，线性约束下可行域又是凸集。上式被称为凸二次规划。



# 二次规划的性质

- QP是一种最简单的非线性规划。QP有如下良好的性质，当 $H$ 是半正定时：
  - K-T条件不仅是最优解的必要条件，而且是充分条件；
  - 局部最优解就是全局最优解。



# 等式约束下的二次规划

$$\begin{aligned} \min f(\mathbf{x}) &= \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{c}^T \mathbf{x}, \mathbf{x} \in R^n \\ \text{s.t. } A \mathbf{x} &= \mathbf{b} \end{aligned} \quad (2)$$

求解方法：Lagrange乘子法，求解以下无约束二次最优化问题。

$$L(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{c}^T \mathbf{x} + \boldsymbol{\lambda}^T (A \mathbf{x} - \mathbf{b})$$

令 $L(\mathbf{x}, \boldsymbol{\lambda})$ 对 $\mathbf{x}$ 和 $\boldsymbol{\lambda}$ 的导数为零，得线性方程组

$$H \mathbf{x} + \mathbf{c}^T + A^T \boldsymbol{\lambda} = \mathbf{0}$$

$$A \mathbf{x} - \mathbf{b} = \mathbf{0}$$

可解得 $\mathbf{x}$ ，即为上式的解。



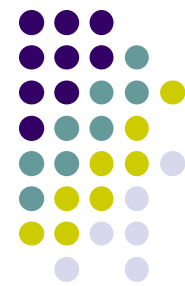
# 凸二次规划的有效集方法

- 直观解释：将不起作用约束去掉，将起作用约束作为等式约束，通过解一系列等式约束的二次规划来实现不等式约束的优化。

- 基本原理：若 $\mathbf{x}$ 为问题（1）的最优解，则它也是问题

$$\begin{aligned} \min & \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{c}^T \mathbf{x} \\ \text{s.t.} & \mathbf{a}_i^T \mathbf{x} = b_i, i \in I \end{aligned} \quad (3)$$

- 的最优解，其中 $\mathbf{a}_i$ 是 $A$ 的第 $i$ 行， $I$ 为起作用约束指标集（有效集）。
- 反之，若 $\mathbf{x}$ 为（1）的可行解，又是（3）的K-T点，且相应的乘子 $\lambda_i \geq 0$ ，则 $\mathbf{x}$ 为（1）的最优解。



# 凸二次规划的有效集方法

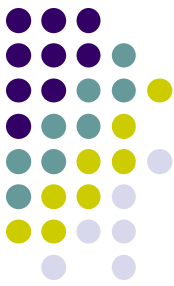
- 算法步骤（迭代法）：

- 设当前迭代点为 $\mathbf{x}_k$ ，它也是（1）的可行解。该点的有效集记作 $I_k$ ，为寻求 $\mathbf{x}_k$ 点的迭代方向 $\mathbf{d}$ ，用乘子法求解

$$\min \frac{1}{2} (\mathbf{x}_k + \mathbf{d})^T H (\mathbf{x}_k + \mathbf{d}) + \mathbf{c}^T (\mathbf{x}_k + \mathbf{d})$$

$$\text{s.t. } \mathbf{a}_i^T \mathbf{d} = 0, i \in I_k$$

- 若所得最优值 $\mathbf{d}_k=0$ ，则 $\mathbf{x}_k$ 是（3）的最优解。
  - 为判断它是否（1）的最优解，考察对应于有效约束的乘子 $\lambda_i \geq 0$ 是否成立。若成立，则 $\mathbf{x}_k$ 是K-T点，由二次规划性质 $\mathbf{x}_k$ 是（1）的最优解。



# 凸二次规划的有效集方法

- 算法步骤（迭代法）：

$$\min \frac{1}{2} (\mathbf{x}_k + \mathbf{d})^T H (\mathbf{x}_k + \mathbf{d}) + \mathbf{c}^T (\mathbf{x}_k + \mathbf{d})$$

$$\text{s.t. } \mathbf{a}_i^T \mathbf{d} = 0, i \in I_k$$

- 若最优值  $\mathbf{d}_k \neq 0$ ，则取  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{d}_k$ ，在  $\mathbf{x}_{k+1}$  为可行点的条件下确定  $\mathbf{d}_k$  方向的步长  $\alpha_k$ 
  - 如果存在  $p$  不在  $I_k$  中，使得  $\mathbf{a}_p \mathbf{x}_{k+1} = \mathbf{b}_p$ ，则将  $p$  加入有效集
- 如果存在  $I_k$  中的指标  $q$ ，使得  $\lambda_i < 0$ ，则  $\mathbf{x}^{(k)}$  不是最优解，从有效集中去掉  $q$



# 可行步长的选取：阻塞约束

$$\alpha_k = \min\left\{1, \min_{i \notin I_k, \mathbf{a}_i^T \mathbf{d}_k < 0} \frac{b_i - \mathbf{a}_i^T \mathbf{x}_k}{\mathbf{a}_i^T \mathbf{d}_k}\right\}$$

- $\alpha_k = 1$ 时，对应约束集不影响，保持不变
- $\alpha_k < 1$ 时，对应约束称为阻塞约束，此时沿着  $\mathbf{d}_k$  运动，会被不在指标集中的约束给阻塞了，约束集因此改变。



# 凸二次规划实例

- libSVM代码
  - <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
  - 有效集方法



# 更多凸二次规划



- 线性互补问题(LCP), Lemke算法
- 可行方向法:
  - Zoutendijk可行方向法
  - Rosen梯度投影法
  - Wolfe既约梯度法
- 罚函数法
- 逐次二次规划法
- 信赖域法