西安交通大學
XI'AN JIAOTONG UNIVERSITY
Systems Engineering Institute
Ministry of Education Key Lab for Intelligent Networks and Network Security

约束最优化方法
Constrained optimization method

电信学部·自动化科学与工程学院
系统工程研究所
吴江

Outline

- 简约梯度法
- 罚函数法
  - 内点罚函数法
  - 外点罚函数法
- 总结

简约梯度法-简介

对于目标函数及约束函数可微的NLP问题，大部分算法的目标在于获得一个可行的KT点，为此需要：

- 将KT条件融入算法设计
- 充分利用问题结构特征进行算法设计

- 简约梯度法(Reduced Gradient Method)是将KT条件与问题结构特征充分结合的一个典型例子
- 简约梯度法是可行方向法的一种

简约梯度法的问题形式

- 简约梯度法可看作是单纯形法的推广
- 核心目标为寻找可行的KT点

$$\begin{aligned} \min \quad & z = f(x) \\ s.t. \quad & g_i(x) \le 0 \\ & h_j(x) = 0 \end{aligned} \quad \xrightarrow{\text{线性约束}} \quad \begin{aligned} \min \quad & z = f(x) \\ s.t. \quad & Ax = b \\ & x \ge 0 \end{aligned}$$

- $x \in R^n$, $f: R^n \to R^1$, $A \in R^{m\times n}$, $\mathrm{Rank}(A)=m$
- 非退化假设
  - 每一个可行点至少有m个大于0的分量
  - A的任意m列线性无关

基变量

非基变量

简约梯度法-基本概念

简约梯度法针对的特定形式NLP:

$$\begin{cases} \min\limits_{x\in R^n} f(x) \\ s.t. \quad Ax=b \\ \qquad\quad x\geq 0 \end{cases}$$

核心思想：消元法＋梯度法 $\rightarrow$ 可行下降方向

$$Ax=b \Leftrightarrow \begin{bmatrix} B & N \end{bmatrix}\begin{pmatrix} x_B \\ x_N \end{pmatrix}=b \Leftrightarrow Bx_B+Nx_N=b \Leftrightarrow x_B=B^{-1}b-B^{-1}Nx_N$$

x_B：基变量；　x_N：非基变量

$$f(x)=f(x_B,x_N)=f(B^{-1}b-B^{-1}Nx_N,x_N)=F(x_N)$$ 简约后的目标函数

$$\frac{\partial F(x_N)}{\partial x_{N_j}}=\frac{\partial f(x)}{\partial x_{N_j}}-\sum_{i=1}^{m}\frac{\partial f(x)}{\partial x_{B_i}}\left(B^{-1}N\right)_{(i,j)}$$ 链导法则

$$r_N=\nabla F(x_N)=\left[\nabla f(x)\right]_N-(B^{-1}N)^T\left[\nabla f(x)\right]_B$$ 简约梯度

简约梯度法-原理

简约梯度法针对的特定形式NLP:

$$\begin{cases} \min\limits_{x\in R^n} f(x) \\ s.t.\quad Ax=b \\ \qquad\quad x\geq 0 \end{cases}$$

$$f(x)=f(x_B,x_N)=f(B^{-1}b-B^{-1}Nx_N,\ x_N)=F(x_N)$$

$$r_N=\nabla F(x_N)=\left[\nabla f(x)\right]_N-(B^{-1}N)^T\left[\nabla f(x)\right]_B$$

- **怎样判断当前点是否为KT点?**
- **当前点不是KT点时怎样获得可行下降方向?**

$\boxed{KT\text{点当且仅当}r_N\geq 0, r_N^T x_N=0}$

x为KT点，当且仅当存在

$\lambda\in R^n,\lambda\geq 0,\mu\in R^m$使得

$$\begin{cases} \nabla f(x)+A^T\mu-\lambda=0 \\ \lambda^T x=0 \end{cases}$$

$$\Leftrightarrow\begin{cases} \begin{pmatrix}(\nabla f(x))_B \\ (\nabla f(x))_N\end{pmatrix}+\begin{pmatrix}B^T\mu \\ N^T\mu\end{pmatrix}=\begin{pmatrix}\lambda_B \\ \lambda_N\end{pmatrix} \\ \lambda_B^T x_B=0,\lambda_N^T x_N=0,\ \lambda\geq 0 \end{cases}$$

非退化假设 $\Rightarrow\lambda_B=0$

$$\begin{cases} (\nabla f(x))_B+B^T\mu=0 \\ (\nabla f(x))_N+N^T\mu=\lambda_N \\ \lambda_N^T x_N=0,\ \lambda_N\geq 0 \end{cases}$$

$$\begin{cases} \mu=-B^{-T}(\nabla f(x))_B \\ \lambda_N=(\nabla f(x))_N-(B^{-1}N)^T(\nabla f(x))_B \\ \lambda_N^T x_N=0,\ \lambda_N\geq 0 \end{cases}$$

简约梯度法-原理

简约梯度法针对的特定形式NLP:

$$\begin{cases} \min\limits_{x\in R^n} f(x) \\ s.t. \quad Ax=b \\ \qquad x\geq 0 \end{cases}$$

$$f(x)=f(x_B,x_N)=f(B^{-1}b-B^{-1}Nx_N,\ x_N)=F(x_N)$$

$$r_N=\nabla F(x_N)=[\nabla f(x)]_N-(B^{-1}N)^T[\nabla f(x)]_B$$

- 怎样判断当前点是否为KT点?
- 当前点不是KT点时怎样获得可行下降方向?

$\boxed{KT点当且仅当 r_N\geq 0, r_N^T x_N=0}$

$$p_{N_j}=\begin{cases} -r_{N_j}, & if \quad r_{N_j}\leq 0 \\ -x_{N_j}r_{N_j}, if & r_{N_j}>0 \end{cases} \Rightarrow \begin{cases} p_N=0\Leftrightarrow r_N\geq 0, r_N^T x_N=0 \\ p_N\neq 0\Rightarrow p_N^T r_N<0 \end{cases}$$

$\boxed{KT点当且仅当 p_N=0}$

结论：$p_N\neq 0$时，是$F(x_N)$的下降方向，是否为可行方向?

$$\begin{cases} x_N: \quad x_N+tp_N; \\ x_B: \quad B^{-1}b-B^{-1}N(x_N+tp_N)=x_B-tB^{-1}Np_N \end{cases}$$

注意$p_{N_j}<0$时$x_{N_j}>0$

结论：非退化假设下，步长较小时，可保证可行性。

$$\boxed{搜索方向 p=\begin{pmatrix} p_B \\ p_N \end{pmatrix}=\begin{pmatrix} -B^{-1}Np_N \\ p_N \end{pmatrix}}$$

$$\boxed{0\leq t\leq t_{\max}=\begin{cases} +\infty, & if\ all\ p_i\geq 0 \\ \min\limits_{1\leq i\leq n}\{-x_i/p_i \mid p_i<0\} \end{cases}}$$

简约梯度法计算步骤

- Step 1 给定初始可行解x^0
- Step 2 由x^k 最大的m个分量, 确定B, N
- Step 3 计算简约梯度

$$r_n = -(B^{-1}N)^T \nabla_B f(x) + \nabla_N f(x)$$

- Step 4 确定搜索方向

$$p_{N_j} = \begin{cases} -r_{N_j}, if & r_{N_j} \le 0 \\ -x_{N_j} r_{N_j}, if & r_{N_j} > 0 \end{cases}$$

- Step 5 判定$\| p \| \le \varepsilon$, 否则一维搜索
- Step 6 k++, 转Step 2

简约梯度法针对的特定形式NLP:

$$\begin{cases} \min\limits_{x \in R^n} f(x) \\ s.t. \quad Ax = b \\ \qquad\ x \ge 0 \end{cases}$$

对中小规模问题效果很好

非退化假设很强，一般不能满足，算法实现时需要其他处理措施

Wolfe(1962)
McCormic(1968)

§5. 约束最优化方法：简约梯度法-例子

例4: (P141例4.5.2)对下述问题，判断给定的两个解是否为KT点.

$$\begin{cases} \min & f(x_1,x_2)=x_1^2+x_2^2+2x_1x_2+2x_1+6x_2 \\ s.t. & x_1+x_2\le 4 \\ & -x_1+x_2\le 2 \\ & x_1,x_2\ge 0 \end{cases}$$

，考虑解 $x=\begin{pmatrix}1\\1\end{pmatrix}$ 和 $x=\begin{pmatrix}0\\0\end{pmatrix}$

$\boxed{KT\text{点当且仅当}p_N=0}$

$$r_N=[\nabla f(x)]_N-(B^{-1}N)^T[\nabla f(x)]_B$$

$$p_{N_j}=\begin{cases}-r_{N_j}, & if \quad r_{N_j}\le 0\\ -x_{N_j}r_{N_j}, & if \quad r_{N_j}>0\end{cases}$$

解: 先化为标准形式，引入松弛变量x_3，x_4.

$$\begin{cases} \min & f(x_1,x_2,x_3,x_4)=x_1^2+x_2^2+2x_1x_2+2x_1+6x_2 \\ s.t. & x_1+x_2+x_3=4 \\ & -x_1+x_2+x_4=2 \\ & x_1,x_2,x_3,x_4\ge 0 \end{cases}$$

A满足非退化假设

$$A=\begin{bmatrix}1&1&1&0\\-1&1&0&1\end{bmatrix},\ b=\begin{pmatrix}4\\2\end{pmatrix}$$

$$r_N=\begin{pmatrix}6\\10\end{pmatrix},\ p_N=\begin{pmatrix}-6\\-10\end{pmatrix}$$

由 $\underbrace{\begin{pmatrix}x_1\\x_2\end{pmatrix}}_{x_N}=\begin{pmatrix}1\\1\end{pmatrix}\Rightarrow\underbrace{\begin{pmatrix}x_3\\x_4\end{pmatrix}}_{x_B}=\begin{pmatrix}2\\2\end{pmatrix}\Rightarrow B=\begin{bmatrix}1&0\\0&1\end{bmatrix},\ N=\begin{bmatrix}1&1\\-1&1\end{bmatrix},\ \begin{cases}(\nabla f(x))_B=\begin{pmatrix}0\\0\end{pmatrix}\\(\nabla f(x))_N=\begin{pmatrix}6\\10\end{pmatrix}\end{cases}$

p_N为负梯度方向

例4: (P141例4.5.2)对下述问题，判断给定的两个解是否为KT点.

$$\begin{cases}\min & f(x_1,x_2)=x_1^2+x_2^2+2x_1x_2+2x_1+6x_2\\ s.t. & x_1+x_2\le 4\\ & -x_1+x_2\le 2\\ & x_1,x_2\ge 0\end{cases}$$

，考虑解 $x=\begin{pmatrix}1\\1\end{pmatrix}$ 和 $x=\begin{pmatrix}0\\0\end{pmatrix}$

KT点当且仅当$p_N=0$

$$r_N=[\nabla f(x)]_N-(B^{-1}N)^T[\nabla f(x)]_B$$

$$p_{N_j}=\begin{cases}-r_{N_j}, & if \quad r_{N_j}\le 0\\ -x_{N_j}r_{N_j}, & if \quad r_{N_j}>0\end{cases}$$

解: 先化为标准形式，引入松弛变量x_3，x_4.

$$\begin{cases}\min & f(x_1,x_2,x_3,x_4)=x_1^2+x_2^2+2x_1x_2+2x_1+6x_2\\ s.t. & x_1+x_2+x_3=4\\ & -x_1+x_2+x_4=2\\ & x_1,x_2,x_3,x_4\ge 0\end{cases}$$

A满足非退化假设

$$A=\begin{bmatrix}1&1&1&0\\-1&1&0&1\end{bmatrix},\ b=\begin{pmatrix}4\\2\end{pmatrix}$$

$$r_N=\begin{pmatrix}2\\6\end{pmatrix},\ p_N=\begin{pmatrix}0\\0\end{pmatrix}$$

由 $\underbrace{\begin{pmatrix}x_1\\x_2\end{pmatrix}}_{x_N}=\begin{pmatrix}0\\0\end{pmatrix}\Rightarrow\underbrace{\begin{pmatrix}x_3\\x_4\end{pmatrix}}_{x_B}=\begin{pmatrix}4\\2\end{pmatrix}\Rightarrow B=\begin{bmatrix}1&0\\0&1\end{bmatrix},N=\begin{bmatrix}1&1\\-1&1\end{bmatrix},\begin{cases}(\nabla f(x))_B=\begin{pmatrix}0\\0\end{pmatrix}\\ (\nabla f(x))_N=\begin{pmatrix}2\\6\end{pmatrix}\end{cases}$

是KT点

惩罚函数法简介

- 基本原理：
  - 把约束优化问题转化成无约束优化问题来求解。
- 两个前提条件：
  - 一是不破坏原约束的约束条件
  - 二是最优解必须归结到原约束问题的最优解上去
- 按照惩罚函数的构成方式，惩罚函数法分为三种：
  - 外点法、内点法、混合法

外点罚函数法-方法

$$\begin{cases} \min \ f(x) \\ s.t. \quad g_i(x) \le 0\,; \quad i = 1,2,\cdots,p \\ \qquad\quad h_j(x) = 0; \quad j = 1,2,\cdots,q \\ \quad x \in R^n \end{cases}$$

$F(x)$的特点：

- $c > 0$, $\alpha > 1$, 一般取$\alpha = 2$
- x可行时$F(x) = f(x)$
- 仅对违反的约束进行惩罚

$$\begin{cases} \min \ F(x) = f(x) + c \cdot \sum_{j=1}^{q} \left| h_j(x) \right|^{\alpha} + c \cdot \sum_{i=1}^{p} \left(\max \left\{ g_i(x), 0 \right\} \right)^{\alpha} \\ s.t. \quad x \in R^n \end{cases}$$

具体实现方法：

选c_k单增$\to +\infty$ $(k = 1,2,\cdots)$, $\quad x^{(k)} = \arg\min_{x \in R^n} F_k(x)$, $\quad$ 则 $\lim_{k \to +\infty} x^{(k)} = x^*$

主要缺点：

- 需要求解一系列无约束优化
- $x^{(k)}$一般不是可行解
- 收敛缓慢

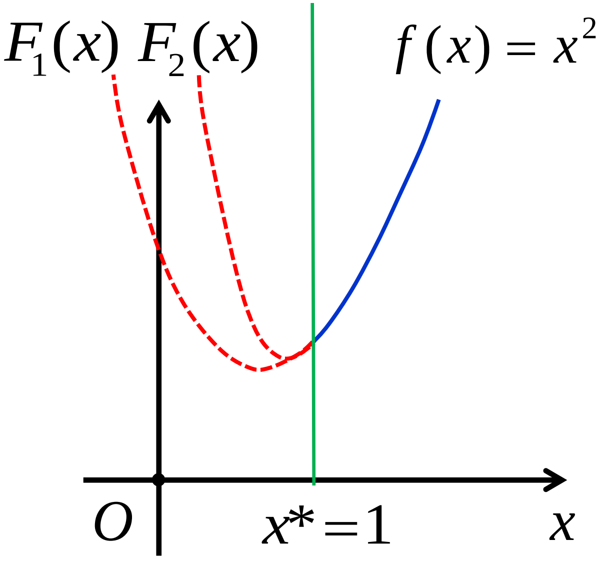
外点罚函数法-原理

例： $\begin{cases} \min \; f(x) = x^2 \\ s.t. \quad 1 - x \le 0 \end{cases}$

解： $\min\limits_{x \in R} F_k(x) = x^2 + k \cdot \left(\max\{1 - x, 0\} \right)^2$

$$\min_{x \in R} F_k(x) = \begin{cases} x^2 \quad , & if \;\; x \ge 1 \\ x^2 + k \cdot (1 - x)^2, & if \;\; x < 1 \end{cases}$$

$$x^{(k)} = \frac{k}{1+k}, \quad x^{(k)} \to 1 = x^*$$



罚因子必须无限增大才能保证解收敛！

具体实现方法：

选c_k单增 $\to +\infty \; (k = 1, 2, \cdots)$, $\quad x^{(k)} = \arg\min\limits_{x \in R^n} F_k(x)$, $\quad$ 则 $\lim\limits_{k \to +\infty} x^{(k)} = x^*$

主要缺点：

- 需要求解一系列无约束优化
- $x^{(k)}$一般不是可行解
- 收敛缓慢

内点法(障碍函数法)

- 思想: 为保证可行性, 从**可行域内部**出发, 在充分接近边界时,所构造的增广目标函数值就会突然增大.

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & g_i(x) \le 0 \end{aligned} \xrightarrow{\text{罚函数}B} \begin{aligned} \min \quad & F(x) = f(x) + B(x) \\ s.t. \quad & x \in R^n \end{aligned}$$

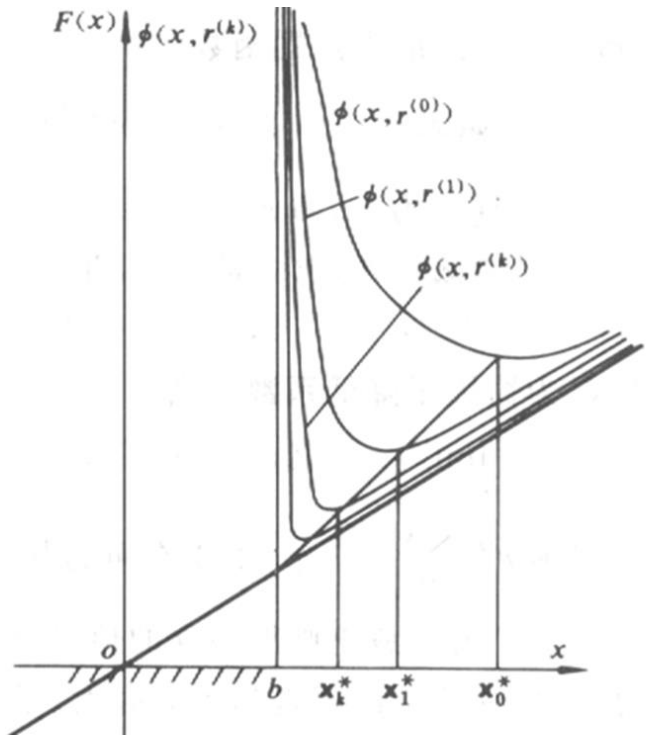
$$B(x) = -\beta \sum_{i=1}^{p} \frac{1}{g_i(x)} \qquad B(x) = -\beta \sum_{i=1}^{p} \ln[-g_i(x)]$$

例:

$$\begin{cases} \min & x^2 \\ s.t. & 1-x \le 0 \end{cases}$$

- 取$\beta_k=1/k$, $k=1, 2, \ldots\ldots$

收敛速度慢
要求初始可行解



约束最优化方法：其他罚函数法

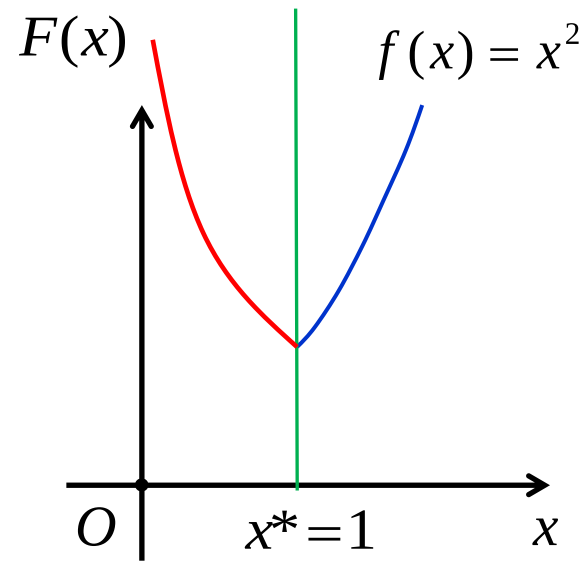
混合罚函数：

外点罚函数 + 内点罚函数

乘子罚函数：

$$\min_{x\in R^n} F(x) = f(x) + \sum_{j=1}^{q} \mu_j h_j(x) + \sum_{i=1}^{p} \lambda_i g_i(x)$$

$$+ c\cdot\sum_{j=1}^{q} \left|h_j(x)\right|^{\alpha} + c\cdot\sum_{i=1}^{p} \left(\max\left\{g_i(x), 0\right\}\right)^{\alpha}$$



精确罚函数：罚因子不必趋于无穷

例：$\begin{cases} \min \ f(x) = x^2 \\ s.t. \quad 1 - x \le 0 \end{cases}$

解：$\min_{x\in R} F(x) = x^2 + c\cdot\max\{1-x, 0\}$

$$\min_{x\in R} F(x) = \begin{cases} x^2 \quad , & if \ x \ge 1 \\ x^2 + c\cdot(1-x), & if \ x < 1 \end{cases}$$

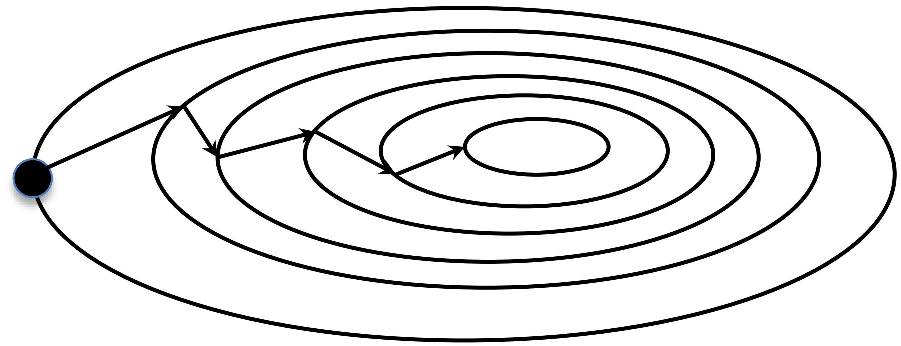
$c \ge 2$时$F(x)$的无约束极小就是x^*

$F(x)$存在不可导点

非线性规划：总结

1. 非线性规划的例子与基本概念

   迭代下降算法框架



2. 凸函数与凸规划

3. 一维搜索方法

4. 无约束最优化方法

   最优性条件、最速下降法、共轭梯度法、……

5. 约束最优化方法

   最优性条件(KT条件)、简约梯度法、罚函数法、…