

一. 基本概念和基本理论

1. 凸集

2. 凸函数

3. 凸规划

二. 线性规划

1. 线性规划问题的提出

2. 线性规划问题的标准形式

3. 线性规划的图解法

4. 线性规划的单纯形法

5. 线性规划MATLIB求解

三. 非线性规划

1.非线性规划数学模型

2.下降迭代算法

3.一维搜索

3.1 黄金分割法

3.2斐波那契法

3.3牛顿法

3.4函数逼近法(插值法)

4.无约束极值问题的求解方法

4.1 梯度法(最速下降法)

4.2 牛顿法

4.3 拟牛顿法

5.约束极值的最优性条件

5.1 库恩-塔克条件

6.约束极值问题的求解算法

6.1 外点法(罚函数法)

6.2 内点法(障碍函数法)

一. 基本概念和基本理论

1. 凸集

设 $S \subset R^n$, 如果 $x^{(1)}, x^{(2)} \in S, \lambda \in [0, 1]$, 均有 $\lambda x^{(1)} + (1 - \lambda)x^{(2)} \in S$, 则称 S 为凸集

有定义知, 凸集的特征是集合中任两点连成的线段必属于这个集合。

2. 凸函数

设 $S \subset R^n$, 非空, 凸集, 函数 $f: S \rightarrow R$, 如果对于 $\forall x^{(1)}, x^{(2)} \in S, \forall \lambda \in [0, 1]$, 恒有 $f(\lambda x^{(1)} + (1 - \lambda)x^{(2)}) \leq \lambda f(x^{(1)}) + (1 - \lambda)f(x^{(2)})$

凸函数的几何意义是对于S上的任意两点, 这两点连成的线段在函数图像上。

3. 凸规划

设 $S \subset R^n$, 非空, 凸, $f: S \rightarrow R$ 是凸函数。 x^* 为问题 (fS) 的 $l. opt$, 则 x^* 为 $g. opt$; 又如果 f 是严格凸函数, 那么 x^* 是问题 (fS) 的唯一 $g. o$

二. 线性规划

1. 线性规划问题的提出

例：某工厂拥有A,B,C三种类型的设备，生成甲、乙两种产品。每件产品在生成中需要占用的设备机时数、每件产品可以获得的利润以及三种设备可利用的时数见下表。问：工厂应如何安排生产可获得最大的总利润？

	产品甲/H	产品乙/H	设备能力/H
设备A	3	2	65
设备B	2	1	40
设备C	0	3	75
利润(元/件)	1500	2500	

解：设变量 x_i 为第 i 种（甲、乙）产品的生产件数（ $i = 1, 2$ ）

目标函数: $\max z = 1500x_1 + 2500x_2$

约束条件: $s.t. \begin{cases} 3x_1 + 2x_2 \leq 65 \\ 2x_1 + x_2 \leq 40 \\ 3x_2 \leq 75 \end{cases}$

这是一个典型的利润最大化的生产计划问题

2. 线性规划问题的标准形式

一般形式:

max z = c1x1 + c2x2 + c3x3 + ... + cnxn
s.t. a11x1 + a12x2 + ... + a1nxn = b1
a21x1 + a22x2 + ... + a2nxn = b2
...
am1x1 + am2x2 + ... + amnxn = bm
x1, x2, ..., xn ≥ 0

矩阵形式

max z = c^T x
s.t. Ax = b
x ≥ 0

3. 线性规划的图解法

略

4. 线性规划的单纯形法

基本思想: 从线性规划问题的某一个基本可行解(即可行域的某一个顶点)出发, 经过最优性判定, 若当前的基本可行解已经是最优解, 则停止; 否则经过基变换得到另一个目标函数值改善的基本可行解。如此反复, 直到找到最优解(即目标函数值取得最优的基本可行解), 或判定问题为无界解或无最优解。

由此可见, 单纯形法需要解决的三个关键问题是:

- 1. 初始基本可行解的确定;
- 2. 解的最优性判定;
- 3. 基的转换。

迭代步骤:

步骤1 列出初始单纯形表, 要求(B^-1b)_i ≥ 0

步骤2 进行最优性判断。若所有的σ_j = (c^T - c_B^T B^-1 A)_j ≤ 0, 则当前表中的基本可行解即为最优解, 计算结束。否则转到步骤3

步骤3 确定换入变量。若存在σ_j > 0, 那么计算max_j {σ_j | σ_j > 0} = σ_s, 将对应的变量x_s作为换入变量

步骤4 确定换出变量。检查换入变量x_s所在的第s列, 若所有的a_is ≤ 0, 则线性规划问题有无界解; 若存在a_is > 0, 那么计算θ = min_i {b_i/a_is | a_is > 0} = b_r/a_rs, 将对应的列变量x_r作为换出变量。

步骤5 重复步骤2~步骤4, 直到计算结束。

例 用单纯形法求解线性规划问题

max z = 40x1 + 50x2
s.t. x1 + 2x2 ≤ 30
3x1 + 2x2 ≤ 60
2x2 ≤ 24
x1, x2 ≥ 0

标准化

max z = 40x1 + 50x2 + 0x3 + 0x4 + 0x5
x1 + 2x2 + x3 = 30
3x1 + 2x2 + x4 = 60
2x2 + x5 = 24
x1, x2, x3, x4, x5 ≥ 0

1. 基变量为x3,x4,x5

cj			40	50	0	0	0	
CB	XB	b	x1	x2	x3	x4	x5	
0	x3	30	1	2	1	0	0	30/2=15
0	x4	60	3	2	0	1	0	60/2=30
0	x5	24	0	2	0	0	1	24/2=12
sigma_j			40	50	0	0	0	

2. x2入基, x5出基

cj			40	50	0	0	0	
----	--	--	----	----	---	---	---	--

CB	XB	b	x1	x2	x3	x4	x5	
0	x3	6	1	0	1	0	-1	6/1=6
0	x4	36	3	0	0	1	-1	36/3=12
50	x2	12	0	1	0	0	1/2	
sigma_j			40	0	0	0	-25	

3. x1入基, x3出基

cj			40	50	0	0	0	
CB	XB	b	x1	x2	x3	x4	x5	
40	x1	6	1	0	1	0	-1	
0	x4	18	0	0	-3	1	2	18/2=9
50	x2	12	0	1	0	0	1/2	12/(1/2)=24
sigma_j			0	0	-40	0	15	

4. x5入基, x4出基

cj			40	50	0	0	0	
CB	XB	b	x1	x2	x3	x4	x5	
40	x1	15	1	0	-1/2	1	0	
0	x5	9	0	0	-3/2	1	1	
50	x2	15/2	0	1	3/4	-1/4	0	
sigma_j			0	0	-35/2	-15/2	0	

5. 线性规划MATLIB求解

优化形式: 单纯形法标准形式, (最小值、小于等于)

`x=linprog(f,A,b)`

输入: f: 目标函数系数向量,A: 不等式约束系数矩阵, b: 不等式约束常数向量

`x=linprog(f,A,b,Aeq,beq)`

输入: Aeq: 等式约束系数矩阵, beq: 等式约束常量矩阵

`x=linprog(f,A,b,Aeq,beq,lb,ub)`

输入: lb: 可行域下界, ub:可行域上界

```
clear all;
f=[-40 -50];
A=[1 2;3 2;0 2];
b=[30 60 24];
x=linprog(f,A,b);
```

三. 非线性规划

1.非线性规划数学模型

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & \begin{cases} g_i(x) \geq 0, i = 1, 2, \dots, l \\ h_j(x) = 0, j = 1, 2, \dots, m \end{cases} \end{aligned}$$

2.下降迭代算法

从前面的无约束优化问题 极值点存在条件中可以看出,对于可微函数, 为求其极值点, 可首先由一阶必要条件令其梯度等于零, 求得驻点; 然后, 再利用二阶充分条件进行判别。事实上, 很多实际问题往往很难求出或根本无法求出目标函数对各自变量的偏导数, 从而一阶必要条件难以应用。因此, 一般情况下, 对非线性规划问题的求解常采用所谓的下降迭代算法。

下降迭代算法的基本思想: 从一个初始估计点 x_0 出发, 按照一定的规则, 找到一个比 x_0 更好的点 x_1 ,再从点 x_1 出发, 找到一个比 x_1 更好的点 x_2 , 如此继续, 得到一个解点序列 $\{x_k\}$.若该点列有一极限点 x^* ,即 $\lim_{k \rightarrow \infty} ||x_k - x^*|| = 0$,就称该点列收敛于 x^* 。若 x^* 是问题 $\min_{x \in \Omega} f(x)$ 的最优解, 则称该算法是有效的下降迭代算法。

下降迭代算法的基本步骤:

步骤1 选取某一点初始点 x_0 , 令 $k:=0$ 。

步骤2 确定搜索方向。若已得出某一迭代点 x_k , 且 x_k 不是极小点, 那么就要根据一定得规则, 从 x_k 点出发确定一个搜索方向 d_k , 沿该方向应能找到使目标函数值下降得可行点。

步骤3 确定步长。沿方向 d_k 前进一个步长, 得到新点 x_{k+1} , 通过选择合适得步长 α_k , 使得下一个迭代点 $x_{k+1} = x_k + \alpha_k d_k$ 满足 $f(x_{k+1}) = f(x_k + \alpha_k d_k) < f(x_k)$

步骤4 最优性检验。检验新得到的点是否使极小点或达到近似极小点的要求。如满足, 则迭代停止; 否则, 令 $k:=k+1$, 返回上面的**步骤2**继续迭代。

迭代终止条件:

1) 按绝对误差:

$$\|x_{k+1} - x_k\| < \varepsilon \text{ 和 } |f(x_{k+1}) - f(x_k)| < \varepsilon$$

2) 按相对误差:

$$\frac{\|x_{k+1} - x_k\|}{\|x_k\|} < \varepsilon \text{ 和 } \frac{|f(x_{k+1}) - f(x_k)|}{f(x_k)} < \varepsilon$$

3) 按目标函数梯度的模:

$$\|\nabla f(x_k)\| < \varepsilon$$

3.一维搜索

一维搜索就是单变量函数在某个区间上求极值点的问题。

一维搜索的方法, 常用的有:

1) 试探法(黄金分割法、斐波那契法等)

2) 微积分中的求根法(切线法、二分法等)

3) 函数逼近法, 也称插值法(抛物线插值法、三次插值法)

3.1 黄金分割法

基本思想:通过尽可能少地选取试验点获得尽可能大的区间缩短率。

在搜索区间中“对称”取点, 等比例地缩小区间。除第一次需要选取两个试验点外, 之后每次都只需要计算一次函数值, 可使包含极小点的区间缩小相同的比例。

假设区间缩短率为 α , 在初始单峰区间 $[a_0, b_0]$ 中, 对称取两点分别为, $t_1 = a_0 + (1 - \alpha)(b_0 - a_0), t_1' = a_0 + \alpha(b_0 - a_0)$

若 $f(t_1) < f(t_1')$, 则得到新区间 $[a_1, b_1]$, 其中 $a_1 = a_0, b_1 = t_1'$, 再在此区间对称取两点 t_2, t_2' , 得到

$$t_2 = a_1 + (1 - \alpha)(b_1 - a_1), t_2' = a_1 + \alpha(b_1 - a_1)$$

由于在新区间 $[a_1, b_1]$ 中, 已经包含了一个已知函数值的试验点 t_1 , 因此, 如能使 t_1 与 t_2' 重合, 则可以节省计算函数值的次数。要满足该条件, 可推得区间缩短率 α 得取值。

要使

$$t_1 = t_2', \text{ 则有 } a_0 + (1 - \alpha)(b_0 - a_0) = a_1 + \alpha(b_1 - a_1), \text{ 代入 } a_1 = a_0, b_1 - a_1 = \alpha(b_0 - a_0), \text{ 得}$$

$$a_0 + (1 - \alpha)(b_0 - a_0) = a_0 + \alpha^2(b_0 - a_0), \text{ 整理得, } \alpha^2 + \alpha - 1 = 0, \text{ 解得, } \alpha = \frac{\sqrt{5}-1}{2} \approx 0.618$$

例2 用黄金分割法求函数 $f(x) = 4x^2 - 6x - 3$ 在区间 $[0, 1]$ 上的近似极小点, 要求缩短后的区间长度不大于原区间长度的10%。

已知 $a_0 = 0, b_0 = 1$, 用黄金分割法得到第一次的试验点:

$$x_1 = 0.382(1 - 0) = 0.382, x_1' = 0.618(1 - 0) = 0.618, f(x_1) = -4.71, f(x_1') = -5.11,$$

$$\text{因, } f(x_1) < f(x_1'), a_1 = x_1 = 0.382, b_1 = b_0 = 1, x_2 = x_1' = 0.618$$

$$x_2' = 0.382 + 0.618(1 - 0.382) = 0.764, f(x_2) = -5.11, f(x_2') = -5.249$$

$$\text{因, } f(x_2') < f(x_2), a_2 = x_2 = 0.618, b_2 = b_1 = 1, x_3 = x_2' = 0.764$$

$$x_3' = 0.618 + 0.618(1 - 0.618) = 0.854, f(x_3) = -5.249, f(x_3') = -5.21$$

SO ON

3.2斐波那契法

3.3牛顿法

基本思想 在一个迭代点附近用切线近似代替曲线，以切线方程的零点作为新的迭代点，逐步逼近最优点。

设 x_k 使 $f'(x) = 0$ 得一个近似根，在点 $(x_k, f'(x_k))$ 处做曲线 $f'(x)$ 的切线。该切线的方程即 $f'(x)$ 在点 x_k 的一阶 *Taylor* 展开式：

$$f'(x) \approx f'(x_k) + f''(x_k)(x - x_k)$$

于是，方程 $f'(x_k) = 0$ 可近似表示为： $f'(x_k) + f''(x_k)(x - x_k) = 0$ ，这是一个线性方程，记其根为 x_{k+1} ，则 x_{k+1} 的计算公式就是牛顿迭

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

牛顿法需要计算目标函数的二阶导数，如果迭代序列收敛，则其收敛速度是比较快的。但是，牛顿法的收敛性依赖于初始值 x_0 的选取。如果 x_0 偏离所求的真解 x^* 比较远，则牛顿法可能发散。

3.4函数逼近法(插值法)

函数逼近法(又称插值法) 是在极小点附近以插值多项式来逼近目标函数的一种方法。

用抛物线插值，对 $f(x)$ 取三个点 $x_1 < x_2 < x_3$ ，且满足 $f(x_1) > f(x_2), f(x_2) < f(x_3)$ 。

$$\phi(x) = ax^2 + bx + c, \text{ 根据插值条件要求}$$

$$\phi(x_1) = ax_1^2 + bx + c = f(x_1)$$

$$\phi(x_2) = ax_2^2 + bx_2 + c = f(x_2)$$

$$\phi(x_3) = ax_3^2 + bx_3 + c = f(x_3)$$

由上述三个等式可求得系数 a, b, c ，从而得到 $f(x)$ 的拟合抛物线 $\phi(x)$ 。接下来，用抛物线 $\phi(x)$ 的极小点来近似 $f(x)$ 的极小点。利用一阶条件

$$\phi'(x) = 2ax + b = 0 \text{ 得到 } \phi(x) \text{ 的极小点：} \hat{x} = -b/2a$$

我们用该极小点 \hat{x} 来近似 $f(x)$ 的极小点 x_k ，然后从 x_1, x_2, x_3 这四个点中再选出3个点。选择的原理是以目标函数值最小的点作为新的 x_2 ，其左右两个相邻的点作为新的 x_1 和 x_3 。依照上述方法继续用新的抛物线函数来逼近 $f(x)$ ，一直下去，就可以得到一个点列 $\{x_k\}$ 。在一定条件下，这个点列收敛于原问题的最优解。

4.无约束极值问题的求解方法

对于一个复杂的非线性函数，用求驻点即求解方程 $\nabla f(x^*) = 0$ 的方法得到其极值点往往是很困难的，特别是对于多变量问题。因此，在实际计算中通常还是要使用迭代的方法。迭代法一般分为两类：一类需要用到函数的导数值，称为**解析法**；另一类则仅用到各点的函数值，称为**直接法**。

4.1 梯度法(最速下降法)

步骤1 给定初始点 x_0 和误差容限 $\epsilon > 0$ ，令 $k := 0$

步骤2 计算负梯度方向 $d_k = -\nabla f(x_k)$

步骤3

检验是否满足收敛性准则 $\|\nabla f(x_k)\| < \epsilon$ 。若满足则停止迭代，得到 $x^* := x_k$ ；否则，进入

步骤4

步骤4 进行一维线性搜索，即求解单变量极值问题，得到步长 α_k

步骤5 令 $x_{k+1} = x_k + \alpha_k d_k$ ，且 $k := k + 1$ ，转到 **步骤2**

例：用梯度法求解下面的二次函数极值问题：

$$\min_x f(x) = \frac{1}{2} x^T Q x - c^T x$$

$$\text{其中, } Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 25 \end{pmatrix}, c = \begin{pmatrix} -1 \\ -1 \\ -1 \end{pmatrix}$$

解

这里目标函数的负梯度方向是 $d_k = -\nabla f(x_k) = -(Qx_k - c)$

$$\text{做精确一维搜索 } \alpha_k = -\frac{\nabla f(x_k)^T d_k}{d_k^T Q d_k}$$

设初始点为

$$x_0 = (0, 0, 0)^T, \text{ 那么 } f(x_0) = 0, \nabla f(x_0) = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \|\nabla f(x_0)\| = 1.7321$$

$$\text{由此得步长 } \alpha_0 = 0.0968, \text{ 新的迭代点 } x_1 = \begin{pmatrix} -0.0968 \\ -0.0968 \\ -0.0968 \end{pmatrix}$$

$$\text{在点 } x_1 \text{ 处, } f(x_1) = -0.1452, \nabla f(x_1) = \begin{pmatrix} 0.9032 \\ 0.5161 \\ -1.4194 \end{pmatrix}, \|\nabla f(x_1)\| = 1.7598$$

$$\text{由此得步长 } \alpha_1 = 0.0590, \text{ 新得迭代点 } x_2 = \begin{pmatrix} -0.1500 \\ -0.1272 \\ -0.0131 \end{pmatrix}$$

$$\text{在点 } x_2 \text{ 处, } f(x_2) = -0.2365, \nabla f(x_2) = \begin{pmatrix} 0.8500 \\ 0.3639 \\ 0.6732 \end{pmatrix}, \|\nabla f(x_2)\| = 1.1437$$

如此迭代下去, 经过216步, 梯度得范数小于 10^{-8} , $x^* = (-1.000, -0.2000, -0.0400)^T$ 。

4.2 牛顿法

牛顿法在搜索方向上比梯度法有所改进。它不但利用了目标函数在搜索方向上得梯度, 还利用了目标函数得二阶导数。

算法

步骤1 给定初始点 x_0 和误差容限 $\epsilon > 0$, 令 $k := 0$

步骤2

检验是否满足收敛性准则 $\|\nabla f(x_k)\| < \epsilon$, 若满足则停止迭代, 得到 $x^* := x_k$; 否则, 进入

步骤3

步骤3 计算牛顿方向: $d_k = -[\nabla^2 f(x_k)]^{-1} \nabla f(x_k)$ 。

步骤4 令 $x_{k+1} = x_k + d_k$, 且 $k := k + 1$, 转到 **步骤2**

例1 用牛顿法求解下面的无约束极值问题: $\min (x_1 + 1)^4 + (x_2 - 2)^2$

解 目标函数的梯度和Hessian矩阵分别为

$$\nabla f(x) = \begin{pmatrix} 4(x_1 - 1)^3 \\ 2(x_2 - 2) \end{pmatrix}, \nabla^2 f(x) = \begin{pmatrix} 12(x_1 - 1)^2 & 0 \\ 0 & 2 \end{pmatrix}$$

取初始点 $x_0 = (0, 0)^T$, 于是

$$\text{第一次迭代: } \nabla f(x_0) = \begin{pmatrix} -4 \\ -4 \end{pmatrix}, \nabla^2 f(x_0) = \begin{pmatrix} 12 & 0 \\ 0 & 2 \end{pmatrix}$$

$$x_1 = x_0 - [\nabla^2 f(x_0)]^{-1} \nabla f(x_0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 12 & 0 \\ 0 & 2 \end{pmatrix}^{-1} \begin{pmatrix} -4 \\ -4 \end{pmatrix} = \begin{pmatrix} 1/3 \\ 2 \end{pmatrix}$$

$$\text{第二次迭代: } \nabla f(x_1) = \begin{pmatrix} -32/27 \\ 0 \end{pmatrix}, \nabla^2 f(x_1) = \begin{pmatrix} 16/3 & 0 \\ 0 & 2 \end{pmatrix}$$

$$x_2 = x_1 - [\nabla^2 f(x_1)]^{-1} \nabla f(x_1) = \begin{pmatrix} 1/3 \\ 2 \end{pmatrix} - \begin{pmatrix} 16/3 & 0 \\ 0 & 2 \end{pmatrix}^{-1} \begin{pmatrix} -32/27 \\ 0 \end{pmatrix} = \begin{pmatrix} 5/9 \\ 2 \end{pmatrix}$$

继续迭代下去.....

4.3 拟牛顿法

牛顿法的优点是收敛速度快, 缺点是计算复杂, 每步迭代都需要计算目标函数的二阶偏导数(Hessian矩阵)和矩阵的逆。这显然会带来一些问题, 例如计算量大、Hessian矩阵可能是非正定的, 从而导致牛顿方向不是一个下降方向等。为了克服牛顿法的缺点, 人们提出了**拟牛顿法**。其

基本思想是用目标函数 f 以及一阶导数 ∇f 构造Hessian矩阵的近似矩阵，由此获得一个搜索方向，生成新的迭代点。

5. 约束极值的最优性条件

5.1 库恩-塔克条件

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g_i(x) \geq 0, i = 1, 2, \dots, l \end{array}$$

定理

设 x^* 是约束优化问题的局部极小值点， $f(x)$ 和 $g_i(x)$ 在点 x^* 处有一阶连续偏导数，并且 x^* 是约束条件的一个正则点，则存在向量 $\mu = (\mu_1, \mu_2, \dots, \mu_l)$

$$\begin{cases} \nabla f(x^*) - \sum_{i=1}^l \mu_i^* \nabla g_i(x^*) = 0 \\ \mu_i^* g_i(x^*) = 0, & i = 1, 2, \dots, l \\ \mu_i^* \geq 0, & i = 1, 2, \dots, l \end{cases}$$

这就是 **K-T条件**, 满足K-T条件的点称为K-T点。

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & \begin{cases} g_i(x) \geq 0, i = 1, 2, \dots, l \\ h_j(x) = 0, j = 1, 2, \dots, m \end{cases} \end{array}$$

定理

设 x^* 是约束优化问题的局部极小点， $f(x)$ 、 $g_i(x)$ 和 $h_j(x)$ 在点 x^* 处有一阶连续偏导数，并且 x^* 是约束条件的一个正则点，则存在向量 $\mu = (\mu_1, \mu_2, \dots, \mu_l, \lambda_1, \lambda_2, \dots, \lambda_m)$

$$\begin{cases} \nabla f(x^*) - \sum_{i=1}^l \mu_i^* \nabla g_i(x^*) - \sum_{j=1}^m \lambda_j^* \nabla h_j(x^*) = 0 \\ \mu_i^* g_i(x^*) = 0, & i = 1, 2, \dots, l \\ \mu_i^* \geq 0, & i = 1, 2, \dots, l \end{cases}$$

K-T条件是确定某点为最优点的必要条件，只要是最优点，且此处起作用约束得梯度线性无关，就会满足这个条件。但一般来说，它不是充分条件，即满足**K-T条件**的点不一定是最优点。特别地，对于凸规划问题，**K-T条件**既是最优点的必要条件，同时也是充分条件。

6. 约束极值问题的求解算法

借助罚函数把约束问题转化为无约束问题来进行求解的约束极值问题。

外点法和内点法都是通过构造某种罚函数，将有约束的优化问题转换为一系列无约束的优化问题来进行求解，因此称为**序列无约束极小化技术**。极限意义下，无约束优化问题的解将是最终收敛到有约束优化问题的解。

6.1 外点法(罚函数法)

考虑下面的非线性规划问题:

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in S \end{array}$$

其中,

f 是 R^n 上的连续函数， S 是约束集。罚函数法的基本思路是用如下的约束优化问题：

$$\min \pi(x, \rho) = f(x) + \rho \phi(x)$$

这里， ρ 是一个正常数， $\phi(x)$ 是 R^n 上的一个连续函数，构造如下：

$$\phi(x) = \begin{cases} 0 & , \text{如果 } x \in S \\ > 0 & , \text{其他} \end{cases}$$

ρ 称为罚因子， $\phi(x)$ 称为罚函数

最常见的罚函数是:

$$\phi(x) = \frac{1}{2} \sum_{i=1}^m g_i(x)^2 = \frac{1}{2} g(x)^T g(x)$$

$$\phi(x) = \frac{1}{\gamma} \sum_{i=1}^m |g_i(x)|^\gamma (\gamma \geq 1)$$

例2 请用外点罚函数法求解下面的约束优化问题:

$$\begin{array}{ll} \min & f(x) = -x_1 x_2 \\ \text{s.t.} & g(x) = x_1 + 2x_2 - 4 = 0 \end{array}$$

解 以二次损失函数为罚函数，得下面得无约束优化问题:

$$\min \quad \pi(x, \rho) = -x_1 x_2 + \frac{1}{2} \rho (x_1 + 2x_2 - 4)^2$$

对给定得 ρ ，由无约束优化问题的一阶最优性条件，得

$$\begin{cases} -x_2 + \rho(x_1 + 2x_2 - 4) = 0 \\ -x_1 + 2\rho(x_1 + 2x_2 - 4) = 0 \end{cases}, \text{求解这个方程组, 得}$$

$$x_1(\rho) = \frac{8\rho}{4\rho-1}, x_2(\rho) = \frac{4\rho}{4\rho-1}$$

$$\text{将 } x(\rho) \text{ 代入约束函数 } g(x), \text{ 得: } g(x(\rho)) = x_1 + 2x_2 - 4 = \frac{16\rho}{4\rho-1} - 4 = \frac{4}{4\rho-1}$$

$$\text{等式约束得拉个朗日乘子 } \lambda(\rho) \text{ 为: } \lambda(\rho) = -\rho g(x(\rho)) = \frac{-4\rho}{4\rho-1}$$

可以看出，当 $\rho \rightarrow \infty$ 时， $g(x(\rho)) \rightarrow 0$, 并且

$$\lim_{\rho \rightarrow \infty} x_1(\rho) = \lim_{\rho \rightarrow \infty} \frac{2}{1-1/4\rho} = 2$$

$$\lim_{\rho \rightarrow \infty} x_2(\rho) = \lim_{\rho \rightarrow \infty} \frac{1}{1-1/4\rho} = 1$$

$$\lim_{\rho \rightarrow \infty} \lambda(\rho) = \lim_{\rho \rightarrow \infty} \frac{-1}{1-1/4\rho} = -1$$

因此，该问题的最优解为 $x^* = (2, 1)^T$ ，最优解的拉格朗日乘子为 $\lambda^* = -1$ 。

6.2 内点法(障碍函数法)

主要思想：在可行域的边界上筑起一道很高的“围墙”，当迭代点从可行域内部靠近边界时，目标函数突然增大以示惩罚，阻止迭代点穿越边界，因此搜索过程始终在可行域内，每一个迭代点都是严格可行的。显然，内点法要求问题的可行域内部非空，因而其不适用于等式约束的优化问题。

考虑不等式约束优化问题:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & x \in S \end{aligned}$$

其中， $S = \{x | g_i(x) \geq 0, i = 1, 2, \dots, m\}$, 并且至少存在某个 x_0 , 使得 $g_i(x_0) > 0, i = 1, \dots, m$ 。内点法的障碍函数时定义于 S 内部的一

通过障碍函数可将不等式优化问题转换为如下的无约束优化问题:

$$\min \beta(x, \mu) = f(x) + \mu \phi(x) \text{ 其中, } \mu \text{ 称为障碍参数。}$$

常见的障碍函数:

$$\text{对数障碍函数: } \phi(x) = -\sum_{i=1}^m \ln(g_i(x))$$

$$\text{倒数障碍函数: } \phi(x) = \sum_{i=1}^m \frac{1}{g_i(x)}$$

例：请用内点法求解下面的约束优化问题:

$$\begin{aligned} \min f(x) &= x_1 - 2x_2 \\ \text{s.t.} \quad &\begin{cases} g_1(x) = 1 + x_1 - x_2^2 \geq 0 \\ g_2(x) = x_2 \geq 0 \end{cases} \end{aligned}$$

解：以对数函数为障碍函数，得到下面的无约束优化问题:

$$\min \beta(x, \mu) = x_1 - 2x_2 - \mu \ln(1 + x_1 - x_2^2) - \mu \ln(x_2), \text{对给定的 } \mu, \text{由无约束优化问题的一阶最优性条件，得:}$$

$$\begin{cases} 1 - \frac{\mu}{1+x_1-x_2^2} = 0 \\ -2 + \frac{2\mu x_2}{1+x_1-x_2^2} - \frac{\mu}{x_2} = 0 \end{cases}, \text{求解这个方程组得}$$

$$x_1(\mu) = \frac{\sqrt{1+2\mu}+3\mu-1}{2}, x_2(\mu) = \frac{1+\sqrt{1+2\mu}}{2}$$

$$\text{令障碍参数 } \mu \rightarrow 0 \text{ 取极限, 得: } \lim_{\mu \rightarrow 0} x_1(\mu) = 0, \lim_{\mu \rightarrow 0} x_2(\mu) = 1$$

因此，该问题的最优化解为 $x^* = (0, 1)^T$