数学实验

Experiments in Mathematics

实验7　无约束优化

Dept. of Mathematical Sciences
Tsinghua University
Beijing 100084, China

优化模型和算法的重要意义

最优化是工程技术、经济管理、科学研究中经常遇到的问题

结构设计　资源分配　生产计划　运输方案

解决优化问题的手段

- 经验积累，主观判断
- 作试验，比优劣
- 建立数学模型，求解最优策略

优化问题的数学模型

$$\min_x z = f(x),\ x \in \Omega \in R^n$$

x ~决策变量，f ~目标函数，Ω ~可行域

$$\min_x \quad z = f(x) \qquad (1)$$

$$s.t.\ \ g_i(x) \le 0,\ i = 1,2,\cdots m \qquad (2)$$

- 可行解（只满足**(2)**）与最优解（满足**(1),(2)**）
- 无约束优化（只有**(1)**）与约束优化（**(1),(2)**）
- 实际问题一般总有约束，何时可用无约束优化处理？

无约束优化的主要内容

1. 优化问题的最优解条件；算法模式

2. 无约束优化的基本方法：梯度法，牛顿法，拟牛顿法

3. 非线性最小二乘法

4. 优化工具箱的使用

5. 实际问题中的无约束优化模型

实例1　产销量安排

某厂生产两个牌号的同一种产品，如何确定产量使利润最大

假设A　产销平衡

牌号	产量	成本	价格
甲	x1	q1	p1
乙	x2	q2	p2

假设B　***p***随***x***（两种牌号）增加而减小，呈线性关系

$$p_1 = b_1 - a_{11}x_1 - a_{12}x_2,\ b_1, a_{11}, a_{12} > 0,\ a_{11} > a_{12}$$

$$p_2 = b_2 - a_{21}x_1 - a_{22}x_2,\ b_2, a_{21}, a_{22} > 0,\ a_{22} > a_{21}$$

实例1　产销量安排

假设C　***q***随***x***（本牌号）增加而减小，呈负指数关系

$$q_1 = r_1 e^{-\lambda_1 x_1} + c_1,\ \ r_1, \lambda_1, c_1 > 0$$

$$q_2 = r_2 e^{-\lambda_2 x_2} + c_2,\ \ r_2, \lambda_2, c_2 > 0$$

目标　利润最大

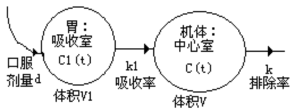
$$\max_{x_1, x_2} z(x_1, x_2) = (p_1 - q_1)x_1 + (p_2 - q_2)x_2$$

实例2 药物的吸收与排除

问题 一室(中心室)模型口服给药方式下的血药浓度c(t)

假设 药物经吸收室进入中心室。药物从吸收室向中心室的转移率与吸收室血药浓度成正比；中心室的排除率与中心室血药浓度成正比。

建模 $\dot{c}_1(t) = -k_1 c_1$



$V\dot{c}(t) = -kVc + k_1 V_1 c_1$

$$\Rightarrow \begin{cases} \dot{c}_1(t) = -k_1 c_1 \\ \dot{c}(t) = -kc + k_1 c_1 V_1 / V \end{cases}$$

$c_1(0) = d/V_1,\ c(0) = 0$ (t=0, 口服剂量d)

模型求解

$\dot{c}_1(t) = -k_1 c_1$

$\dot{c}(t) = -kc + k_1 c_1 V_1 / V$

$c_1(0) = d/V_1,\ c(0) = 0$

$$c_1(t) = \frac{d}{V_1} e^{-k_1 t}$$

$$c(t) = \frac{d}{V} \frac{k_1}{k_1 - k}(e^{-kt} - e^{-k_1 t})$$

参数估计 用测试分析确定$c(t)$中的参数k, k_1, V

今测得一组数据 (t_i, c_i), i=1,…n

t_i	0.083	0.167	0.25	0.50	0.75	1.0	1.5
c_i	10.9	21.1	27.3	36.4	35.5	38.4	34.8
t_i	2.25	3.0	4.0	6.0	8.0	10.0	12.0
c_i	24.2	23.6	15.7	8.2	8.3	2.2	1.8

参数估计

$$c(t) = \frac{d}{V} \frac{k_1}{k_1 - k}(e^{-kt} - e^{-k_1 t})$$

用最小二乘法求参数k, k_1, V（已知d），使理论值$c(t_i)$与实测值c_i的误差平方和最小.

$$R(k, k_1, V) = \sum_{i=1}^{n} [c(t_i) - c_i]^2 = \sum_{i=1}^{n} \left[\frac{d}{V} \frac{k_1}{k_1 - k}(e^{-kt_i} - e^{-k_1 t_i}) - c_i\right]^2$$

非线性最小二乘问题, 无约束优化.

约束：k, k_1, V>0?

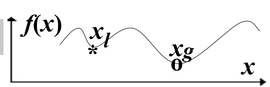
无约束优化的提法

给定一个函数 $f(x)$, 寻找 x^* 使得 $f(x^*)$最小，即

$\underset{x}{Min}\ f(x)$ 其中 $x = (x_1, x_2, \cdots, x_n)^T \in \Re^n$

最优解的分类和条件

局部最优解 全局最优解

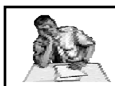


必要条件 $\nabla f(x^*) = (f_{x_1}, \cdots, f_{x_n})^T = 0$

充分条件 $\nabla f(x^*) = 0,\ \nabla^2 f(x^*) > 0$

$$\nabla^2 f = \left[\frac{\partial^2 f}{\partial x_i \partial x_j}\right]_{n \times n}$$

Hessian阵



求解无约束优化的基本思路

基本思想 在$\Re^n$中某一点，确定一个搜索方向及沿该方向的移动步长，得到使目标函数下降的新的点

迭代步骤

Step 1 初始化：初始点x^0，终止准则等

Step 2 迭代改进：方向d^k，步长α^k

$x^{k+1} = x^k + \alpha^k d^k$ $f(x^{k+1}) < f(x^k)$

Step 3 终止检验：得到近优解或k+1⇒k转2

选择d^k,α^k 使f下降更快 ⇒ 不同算法

无约束优化的基本方法(搜索方向的选择)

1 最速下降法（梯度法）

暂不考虑搜索步长，可设α^k=1

将 $f(x^{k+1})$ 在 x^k 点作泰勒展开，只保留一阶项，有

$$f(x^{k+1}) = f(x^k + d^k) = f(x^k) + \nabla f^T(x^k) d^k$$

下降方向 $\nabla f^T(x^k) d^k < 0$

最速下降方向 $d^k = -\nabla f(x^k)$ （负梯度方向）

迭代改进格式 $x^{k+1} = x^k - \nabla f(x^k)$

算法特点 初始阶段改进较快，最优解附近改进较慢

2 Newton方法

将$f(x^{k+1})$在x^k点作泰勒展开至二阶项，用d替代d^k

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

求d使$f(x^{k+1})$极小⇒右端对d导数为0 $\Rightarrow \nabla f(x^k)+\nabla^2 f(x^k)d=0$

牛顿方程 $\nabla^2 f(x^k)d = -\nabla f(x^k)$

牛顿方向 $d^k = -(\nabla^2 f(x^k))^{-1}\nabla f(x^k)$

迭代格式 $x^{k+1} = x^k - (\nabla^2 f(x^k))^{-1}\nabla f(x^k)$

特点 局部2阶收敛;需计算Hessian阵,它可能病态或不正定

比较 解$F(x)=0$的牛顿法 $x^{k+1} = x^k - [F'(x^k)]^{-1}F(x^k)$

$F(x) \Leftrightarrow \nabla f(x)$

2000-11-11 13

3 拟Newton方法

目的 不计算Hessian阵，克服病态、不正定、计算复杂等缺陷，同时保持收敛较快的优点

思路 回顾解方程组 F(x)=0的拟牛顿法

$x^{k+1} = x^k - [F'(x^k)]^{-1}F(x^k)$ ⇨ $x^{k+1} = x^k - (A^k)^{-1}F(x^k)$

使A^k满足 $A^k(x^k - x^{k-1}) = F(x^k) - F(x^{k-1})$

用迭代方法$A^k = A^{k-1} + \Delta A^{k-1}$ 计算A^k

优化问题 $\underset{x}{Min}\ f(x)$ $\nabla f(x)$相当$F(x)$

$\nabla^2 f(x)$相当$F'(x)$, $\nabla^2 f$不一定正定，构造正定阵G代替$\nabla^2 f$

2000-11-11 14

无约束优化(搜索方向的选择) 3 拟Newton方法

设在第k步, G^k已得到, $H^k=(G^k)^{-1}$,可计算

$$x^{k+1} = x^k - H^k\nabla f(x^k)$$

记 $\Delta x^k = x^{k+1} - x^k$, $\Delta f^k = \nabla f(x^{k+1}) - \nabla f(x^k)$

按照拟牛顿条件:

$$G^{k+1}\Delta x^k = \Delta f^k \text{ 或 } \Delta x^k = H^{k+1}\Delta f^k$$

构造 迭代公式$G^{k+1} = G^k + \Delta G^k$ 或$H^{k+1} = H^k + \Delta H^k$

于是有 $x^{k+2} = x^{k+1} - H^{k+1}\nabla f(x^{k+1})$

2000-11-11 15

无约束优化(搜索方向的选择) 3 拟Newton方法

3.1 Davidon-Fletcher-Powell(DFP)公式

$$\Delta H^k = \frac{\Delta x^k(\Delta x^k)^T}{(\Delta x^k)^T\Delta f^k} - \frac{H^k\Delta f^k(\Delta f^k)^T H^k}{(\Delta f^k)^T H^k \Delta f^k}$$

$$\Delta G^k = (1 + \frac{(\Delta x^k)^T G^k \Delta x^k}{(\Delta x^k)^T\Delta f^k})\frac{\Delta f^k(\Delta f^k)^T}{(\Delta x^k)^T\Delta f^k} - \frac{\Delta f^k(\Delta x^k)^T G^k + G^k\Delta x^k(\Delta f^k)^T}{(\Delta x^k)^T\Delta f^k}$$

特点 若迭代点处梯度非零，则矩阵H^k对称正定，并且具有二次终止性。

2000-11-11 16

无约束优化(搜索方向的选择) 3 拟Newton方法

3.2 Broyden-Fletcher-Goldfarb-Shanno(BFGS)公式

$$\Delta G^k = \frac{\Delta f^k(\Delta f^k)^T}{(\Delta f^k)^T\Delta x^k} - \frac{G^k\Delta x^k(\Delta x^k)^T G^k}{(\Delta x^k)^T G^k \Delta x^k}$$

$$\Delta H^k = (1 + \frac{(\Delta f^k)^T H^k \Delta f^k}{(\Delta x^k)^T\Delta f^k})\frac{\Delta x^k(\Delta x^k)^T}{(\Delta x^k)^T\Delta f^k} - \frac{\Delta x^k(\Delta f^k)^T H^k + H^k\Delta f^k(\Delta x^k)^T}{(\Delta x^k)^T\Delta f^k}$$

特点 若$(\Delta f^k)^T\Delta x_k > 0$, 则$H^k$对称正定，且迭代2阶收敛

- **DFP与BFGS 互为对偶：** $\Delta x^k, \Delta f^k$互换, $\Delta G^k, \Delta H^k$互换

2000-11-11 17

无约束优化的基本方法(搜索步长的确定)

4 线性搜索确定步长方法

问题 给定x^k和方向d^k, 确定步长α^k, 使得

$\min_{\alpha} f(x^k + \alpha d^k)$ ~一维优化问题

$\left.\frac{df}{d\alpha}\right|_{\alpha^k} = 0$ ⇨ $(d^k)^T\nabla f(x^k + \alpha^k d^k) = 0$

优化算法 黄金分割(0.618)法、Fibonacci法、Newton切线法、割线法、2次或3次插值法等

2000-11-11 18

无约束优化的基本方法

5 非线性最小二乘拟合方法

问题　给定(t_i, y_i), i=1,…n, 拟合一个函数$y=f(t, x)$, 其中x为待定的参数向量, f对x非线性。

记误差　$r_i(x)=y_i-f(t_i,x)$　$r(x)=(r_1(x),\cdots r_n(x))^T$

优化模型　$\min\limits_{x} R(x)=r^T(x)r(x)$

根据目标函数是$r(x)$的二次函数的特点
构造简单算法

无约束优化(非线性最小二乘拟合)　$R(x)=r^T(x)r(x)$

记$r(x)$的雅各比阵为　$J(x)=(\partial r_i/\partial x_j)_{n\times m}$

$\nabla R=2J(x)^T r(x)$　$\nabla^2 R=2J(x)^T J(x)+2S$

$S=\sum\limits_{i=1}^{n} r_i(x)\nabla^2 r_i(x)$　$\nabla^2 r_i(x)=(\partial^2 r_i/\partial x_k \partial x_l)_{m\times m}$

讨论
- 牛顿法要计算Hessian矩阵，其中S计算量大。
- 若f对x线性, 则化为线性最小二乘拟合,此时S=0

特定算法考虑如何忽略或近似矩阵S。

无约束优化(非线性最小二乘拟合)

5.1 Gauss-Newton算法：忽略矩阵S

$\nabla R=2J(x)^T r(x)$　$\nabla^2 R=2J(x)^T J(x)$

牛顿方程　$\nabla^2 f(x^k)d=-\nabla f(x^k)$

f用R代替，下降方向d^k满足

$J(x^k)^T J(x^k)d^k=-J(x^k)r(x^k)$

G-N算法　收敛性依赖f对x的线性程度, 及偏差r的大小

无约束优化(非线性最小二乘拟合)

5.2 Levenbery-Marquardt算法：G-N算法修正

$J(x^k)^T J(x^k)d^k=-J(x^k)r(x^k)$

⇓　防止J^TJ出现病态

$(J(x^k)^T J(x^k)+\alpha^k I)d^k=-J(x^k)r(x^k)$

其中$\alpha^k>0$为修正参数.

L-M算法　d^k位于牛顿方向(α^k很小)和负梯度方向(α^k很大)之间

优化工具箱 (matlab\toolbox\optim)

无约束优化　模型: $\mathop{Min}\limits_{x} f(x), x\in R^n$

1) 基本程序： `x=fminu('fun',x0)`
输入：`fun.m` ~f(x)的m文件名，　x0 ~初始点，
输出：　x ~最优解

例 1　求解 $\min \dfrac{x^2}{a}+\dfrac{y^2}{b}$,　$a=b=2$　　examp071.m

```
function y=fun1(x)
a=2;b=2;
y=x(1)^2/a+x(2)^2/b;
```

```
x0=[1,1];
x=fminu('fun1',x0)
```

⇨ `X=1.0e-009*(-0.3649,-0.3649)`

无约束优化　模型: $\mathop{Min}\limits_{x} f(x), x\in R^n$

2) 提高精度，观察中间结果，给出迭代次数和最优值

`[x,fopt] =fminu('fun',x0,fopt)`

输入：　`fopt(1)=0`（缺省值）～无中间结果输出
`fopt(1)=1` ～中间结果输出

fOpt~控制参数　`fopt(2)` ~设置最优解x的精度
`fopt(3)` ~设置最优值f的精度

输出：　`fopt(8)`　~最优值f
`fopt(10)`~迭代次数

例 2　$\min \dfrac{x_1^2}{a}+\dfrac{x_2^2}{b}$,　$a=10, b=1$　　examp072.m

无约束优化　　模型: $\underset{x}{Min}\ f(x), x\in R^n$

3) 算法选择: `x=fminu('fun',x0,fopt)`

输入: `fopt(6)=0`（缺省值） ~BFGS公式

`fopt(6)=1` ~DFP公式

`fopt(6)=2` ~最速下降法

`fopt(7)=0`（缺省值） ~混合2, 3次多项式插值

`fopt(7)=1` ~ 3次多项式插值

`fopt(6)`~搜索方向　　`fopt(7)` ~搜索步长

`help foptions`

2000-11-11　　25

无约束优化　　模型: $\underset{x}{Min}\ f(x), x\in R^n$

例3. $\min f(x,y)=100(y-x^2)^2+(1-x)^2$

精确解: $x=y=1,\ f(x,y)=0$

计算结果　　**examp073.m**

方向	步长	最优解x	最优值f	n
BFGS	2,3次	(9.9997e-001 9.9994e-001)	1.0944e-009	14
DFP	2,3次	(9.9997e-001 9.9994e-001)	2.1173e-009	1000
GRAD	2,3次	(-6.4113e-001 4.2250e-001)	2.7064e+000	1000
BFGS	3次	(1.0000e+000 9.9999e-001)	5.9567e-010	226
DFP	3次	(5.9547e-001 3.4591e-001)	1.7117e-001	220

2000-11-11　　26

无约束优化　　模型: $\underset{x}{Min}\ f(x), x\in R^n$

4) 采用分析梯度:

`x=fminu('fun',x0,fopt,'grad')`

`grad.m` ~ $\nabla f(x)$ 的m文件名

例4. $\min f(x,y)=100(y-x^2)^2+(1-x)^2$

$$\text{算出}\ \nabla f=\begin{bmatrix}-400x(y-x^2)-2(1-x)\\200(y-x^2)\end{bmatrix}$$

examp074.m

5）其他: 用`fopt(14)`输入最大迭代次数，（缺省值0表示100n,n为变量个数 ）。

2000-11-11　　27

计算结果

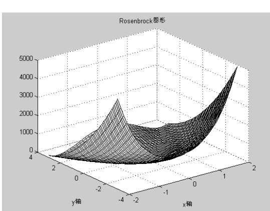
方向	步长	最优解x	最优值f	n
BFGS	2,3次	(1.0000e+000 1.0000e+000)	6.3209e-011	54
DFP	2,3次	(3.7903e-001 1.3278e-001)	3.9744e-001	72
BFGS	3次	(1.0000e+000 1.0000e+000)	8.0378e-015	26
DFP	3次	(1.0001e+000 1.0003e+000)	2.2499e-008	25

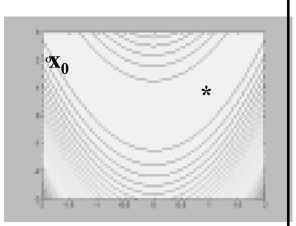
与不用分析梯度的结果比较

方向	步长	最优解x	最优值f	n
BFGS	2,3次	(9.9997e-001 9.9994e-001)	1.0944e-009	14
DFP	2,3次	(9.9997e-001 9.9994e-001)	2.1173e-009	1000
GRAD	2,3次	(-6.4113e-001 4.2250e-001)	2.7064e+000	1000
BFGS	3次	(1.0000e+000 9.9999e-001)	5.9567e-010	226
DFP	3次	(5.9547e-001 3.4591e-001)	1.7117e-001	220

2000-11-11　　28

$f(x,y)=100(y-x^2)^2+(1-x)^2$ 的图形





2000-11-11　　29

无约束优化　　几个值得注意的问题

梯度函数: 利用分析梯度可能改进算法的性能

算法选择: **foptions(6)=0**（缺省值）~ **BFGS**公式，**foptions(7)=0**（缺省值)~混合**2**，**3**次插值，一般较好。

高度非线性、不连续时可用程序 **fmins**

精度控制: 用**foptions(2: 3)**控制，对迭代次数有重大影响，应适当选择。

改变初始值 由一个初值出发通常得到局部最优解，如果函数存在多个局部最优，只有改变初值，对局部最优进行比较，才有可能得到全局最优解。

注: fminu将被fminunc取代; fmins将被fminsearch取代; foptions将被optimset，optimget取代（用法有变化）

优化工具箱 (matlab\toolbox\optim)

非线性最小二乘方法

$$r_i(x)=y_i-f(t_i,x)$$

$$\min_x R(x)=r^T(x)r(x)$$

$$r(x)=(r_1(x),\cdots r_n(x))^T$$

```
[x,fopt,F]=leastsq('fun',x0,fopt,
'grad',p1,…)
```

输入的用法与fminu相同， 但注意：fun.m ~r(x) 的m文件名， grad.m ~ $\nabla r(x)$ 的m文件名

输出 F=r(x)（误差向量），R(x)=F^T*F

注：leastsq 将被 lsqnonlin取代（用法有变化）；

数据拟合也可用curvefit（将被 lsqcurvefit取代）

优化实例 实例1 产销量安排

原问题

$$p_i=b_i-a_{i1}x_1-a_{i2}x_2,\ b_i,a_{i1},a_{i2}>0,\ i=1,2,$$

$$q_i=r_ie^{-\lambda_i x_i}+c_i,\ r_i,\lambda_i,c_i>0,\ i=1,2,$$

$$\max_{x_1,x_2} z(x_1,x_2)=(p_1-q_1)x_1+(p_2-q_2)x_2$$

已知数据

$$\begin{pmatrix}a_{11}&a_{12}\\a_{21}&a_{22}\end{pmatrix}=\begin{pmatrix}1&0.1\\0.2&2\end{pmatrix}\quad b=\begin{pmatrix}100\\280\end{pmatrix}$$

$$r^T=(30\quad 100),\quad \lambda^T=(0.015\quad 0.02),\quad c^T=(20\quad 30)$$

$$\min\ f(x)=-(b_1-a_{11}x_1-a_{12}x_2-r_1e^{-\lambda_1x_1}-c_1)x_1$$
$$-(b_2-a_{21}x_1-a_{22}x_2-r_2e^{-\lambda_2x_2}-c_2)x_2$$

2000-11-11 32

优化实例 实例1 产销量安排

初始点的选择

忽略成本及价格中的 a_{12}, a_{21}

$$\begin{pmatrix}a_{11}&a_{12}\\a_{21}&a_{22}\end{pmatrix}=\begin{pmatrix}1&0.1\\0.2&2\end{pmatrix}$$

问题简化为 $\min\ f(x)=-(b_1-a_{11}x_1)x_1-(b_2-a_{22}x_2)x_2$

其解为 $x_1=b_1/2a_{11}=50,\quad x_2=b_2/2a_{22}=70$

作为原问题的初始点

命令和最优解 **x=fminu('f(x)', x0)** **shili071.m**

x = 23.9025 62.4977 y = 6.4135e+003

即甲产量为23.9025，乙产量为62.4977，最大利润为6413.5

2000-11-11 33

优化实例 实例2 药物的吸收与排除

原问题 $c(t)=\dfrac{d}{V}\dfrac{k_1}{k_1-k}(e^{-kt}-e^{-k_1t})$ 测得一组数据 (t_i, c_i), i=1,…n

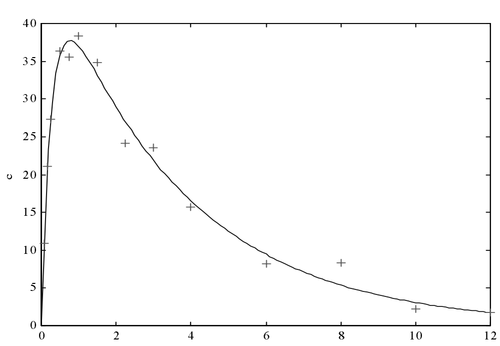
用非线性最小二乘拟合求参数k,k_1,V, 使误差R最小

$$R(k,k_1,V)=\sum_{i=1}^{n}[c(t_i)-c_i]^2=\sum_{i=1}^{n}\left[\frac{d}{V}\frac{k_1}{k_1-k}(e^{-kt_i}-e^{-k_1t_i})-c_i\right]^2$$

拟合结果为 $(k_1,k,d/V)=(3.6212,0.2803,46.8275)$

根据d/V和d可计算V **shili072.m**

2000-11-11 34



血药浓度实验数据拟合图

200 35

无 约 束 优 化 实 验 内 容

实验目的

1．掌握 Matlab 优化工具包的基本用法，对不同算法进行初步分析、比较。

2．练习实际问题的非线性最小二乘拟合。

内容 2； 6（参看182页及198页内容）

2000-11-11 36