实验指导书：非线性规划模型

【实验目的】

1、通过上机采用 MATLAB 优化工具箱求解无约束最优化问题，进一步掌握有关无约束最优化问题的分析、建模与求解方法。

【实验相关知识】

非线性优化包括相当丰富的内容，我们这里就 Matlab 提供的一些函数来介绍相关函数的用法及其所能解决的问题。

（一）非线性一元函数的最小值

线性函数是一次函数的别称，则非线性函数即函数图像不是一条直线的函数。非线性函数包括指数函数、幂函数、对数函数、多项式函数等等基本初等函数以及他们组成的复合函数。主要的解法：可以利用几何图形较为明确的函数，通过几何模型，寻找函数最值。

Matlab 命令为 fminbnd()，其使用格式为：

X=fminbnd(fun,x1,x2)

[X,fval,exitflag,output]= fminbnd(fun,x1,x2)

其中：fun 为目标函数，x1，x2 为变量得边界约束，即 x1≤x≤x2，X 为返回得满足 fun 取得最小值的 x 的值，而 fval 则为此时的目标函数值。exitflag>0 表示计算收敛，exitflag=0 表示超过了最大的迭代次数，exitflag<0 表示计算不收敛，返回值 output 有 3 个分量，其中 iterations 是优化过程中迭代次数，funcCount 是代入函数值的次数，algorithm 是优化所采用的算法。

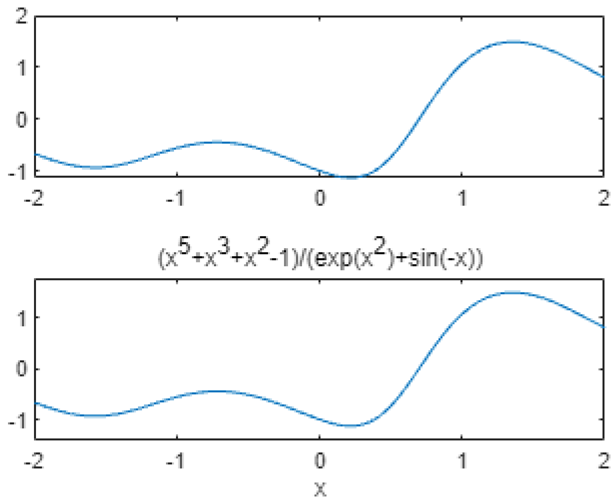
例 1：求函数 $f(x)=\dfrac{x^5+x^3+x^2-1}{e^{x^2}+\sin(-x)}$ 在区间 $[-2,2]$ 的最小值和相应的 x 值。

解决此问题的 Matlab 程序为：

```
clear all
x=-2:0.01:2;
y=(x.^5+x.^3+x.^2-1)./(exp(x.^2)+sin(-1.*x));
subplot (2,1,1)
plot(x,y);
subplot (2,1,2)
fun='(x^5+x^3+x^2-1)/(exp(x^2)+sin(-x))'
ezplot(fun,[-2,2])
```

[X,fval,exitflag,output]= fminbnd(fun,-2,2)

结果为：



X =   0.2176

fval =-1.1312

exitflag = 1

output = iterations: 13

funcCount: 13

algorithm: 'golden section search, parabolic interpolation'

（二）无约束非线性多元变量的优化

这里我们介绍两个命令：fminsearch()和 fminunc()，前者适合处理阶次低，但是间断点多的函数，后者则对于高阶连续的函数比较有效。

命令 fminsearch()的格式为：

X= fminsearch(fun,X0)

[X,fval,exitflag,output]= fminsearch(fun,X0,options)

该命令求解目标函数 fun 的最小值和相应的 x 值，X0 为 x 的初始值，

fval 为返回的函数值，

exitflag=1 表示优化结果收敛，exitflag=0 表示超过了最大迭代次数。

返回值 output 有 3 个分量，其中 iterations 是优化过程中迭代次数，funcCount 是代入函数值的次数，algorithm 是优化所采用的算法。

Options 是一个结构，里面有控制优化过程的各种参数，参考 optimset()命令来设置，一般情况下我们不必改动它，即使用缺省设置就可以了。

例 2：求函数 $f(x,y)=\sin^2 x+\cos y$ 的最小值以及最小值点。

完成该计算的 Matlab 程序如下：

clear

fun1='sin(x)+cos(y)'

fun2='sin(x(1))+cos(x(2))'

ezmesh(fun1)

[X,fval]=fminsearch(fun2,[0,0])

X =   -1.5708   3.1416

fval = -2.0000

其中语句 ezmesh()是为了画出函数的图形，注意这里 fun1 和 fun2 的不同，考虑如果用相同的是否可行。

命令 fminunc()的格式为：

X=fminunc(fun,X0)

[X,fval,exitflag,output,grad,hessian]=fminunc(fun,X0,options)

命令 fminunc()通过计算寻找多变量目标函数 fun 的最小值，

X0 为优化的初始值，

X 为返回的变量的值，

grad 返回解点的梯度，

hessian 返回解点的赫森矩阵。其它参数的意义和命令 fminsearch()相同。

例 3：求函数 $f(x_1,x_2)=e^{x_1}(2x_1+3x_2^2+2x_1x_2+3x_2+1)$ 的最小值。

Matlab 程序为.0

clear

fun='exp(x(1))*(2*x(1)^2+3*x(2)^2+2*x(1)*x(2)+3*x(2)+1)';

x0=[0,0];

options=optimset('largescale','off','display','iter','tolx',1e-8,'tolfun',1e-8);

[x,fval,exitflag,output,grad,hessian]=fminunc(fun,x0,options)

运行结果为：

Iteration	Func-count	f(x)	Step-size	Directional derivative
1	2	1	0.2	-10
2	8	0.369471	0.134277	-0.0203
3	14	0.154419	0.459778	-0.0696
4	20	0.134704	0.746874	-2.28e-005
5	26	0.132961	0.63991	-1.1e-007
6	32	0.132961	0.897232	-7.32e-009

Optimization terminated successfully:

Current search direction is a descent direction, and magnitude of directional derivative in search direction less than 2*options.TolFun x =   0.2695   -0.5898

fval =   0.1330

exitflag =   1

output =   iterations: 6

funcCount: 33

stepsize: 1.0000

firstorderopt: 1.6892e-005

algorithm: 'medium-scale: Quasi-Newton line search'

grad = 1.0e-004 * (-0.1689,   0.0074)

hessian =   5.1110   2.6437

2.6437   8.0539

本例的程序对参数 options 进行了设置，

'largescale','off'，关闭了大规模方式，

'display',用来控制计算过程的显示，

'iter'表示显示优化过程的每次计算结果，

'off'表示不显示所有输出，'final'仅输出最后结果，

'tolx'用来控制输入变量 x 的允许误差精度，本利设置为 1e-8，

'tolfun'是控制目标函数的允许误差精度，缺省值是 1e-4，本例为 1e-8。

（三）有约束非线性多元变量的优化

由线性规划我们看到优化要处理各种约束条件，在非线性规划中问题就更加复杂，除了线性规划中的那些约束外，还要增加非线性约束。Matlab 的命令函数 fmincon()可以处理有约束的非线性多元函数的优化问题。

有约束多变量优化问题的数学模型为：求一组变量 $x_1, x_2, \cdots, x_n$，满足在给定的约束条件下，使目标函数 $f(x_1, x_2, \cdots, x_n)$ 最小。目标函数一般为非线性函数，约束条件分为线性不等式约束、线性等式约束、变量边界约束和非线性约束几部分。除非线性约束外，表示方法与线性规划相同。函数 fmincon()的具体格式为：

X=fmincon(fun,x0,A,b)

X=fmincon(fun,x0,A,b,Aeq,Beq,Lb,Ub)

X=fmincon(fun,x0,A,b,Aeq,Beq,Lb,Ub,nonlcon,options)

[X,fval,exitflag,output]=fmincon(fun,x0,…)

[X,fval,exitflag,output,lambda,grad,Hessian]=fmincon(fun,x0,…)

参数中 fun 为目标函数，x0 为变量的初始值，x 为返回的满足要求的变量的值。

A 和 b 表示线性不等式约束，

Aeq，beq 表示线性等式约束，

Lb 和 Ub 分别为变量的下界和上界约束，

nonlcon 表示非线性约束条件，

options 为控制优化过程的优化参数向量。

返回值 fval 为目标函数。

exitflag>0 表示优化结果收敛于解，exitflag=0 表示优化超过了函数值的计算次数，exitflag<0 表示优化不收敛。

lambda 是拉格朗日乘子，显示那个约束条件有效。

grad 表示梯度，

hessian 表示汉森矩阵。

例 4：求$[x_1,x_2]$，使得目标函数$f(x_1,x_2)=e^{x_1}(4x_1^2+2x_2^2+4x_1x_2+2x_2+1)$在约束条件$1.5+x_1*x_2-x_1-x_2\le 0$，$-x_1*x_2\le 10$下取得最小值。

我们设计的程序如下：

先把目标函数和约束条件分别编写成独立的 m 文件，注意，这样的 m 文件必须用 function 开头，并且文件名一定要和函数名一致。目标函数的文件为：

```
function f=objfun(x)
f=exp(x(1))*(4*x(1)^2+2*x(2)^2+4*x(1)*x(2)+2*x(2)+1);
```

约束条件的文件为：

```
function [c,ceq]=confun(x)
c=[1.5+x(1)*x(2)-x(1)-x(2);-x(1)*x(2)-10];
ceq=[];
```

接着，编写完成优化的程序如下：

```
clear
x0=[-1 1];
options=optimset('largescale','off','display','iter');
[x,fval,exitflag,output]=fmincon(@objfun,x0,[],[],[],[],[],[],@confun,options)
```

运行结果为：

Iter	F-count	f(x)	constraint	max Step-size	Directional derivative	Procedure
1	3	1.8394	0.5	1	0.0486	
2	7	1.85127	-0.09197	1	-0.556	Hessian modified twice
3	11	0.300167	9.33	1	0.17	
4	15	0.529834	0.9209	1	-0.965	
5	20	0.186965	-1.517	0.5	-0.168	

6	24	0.0729085	0.3313	1	-0.0518	
7	28	0.0353322	-0.03303	1	-0.0142	
8	32	0.0235566	0.003184	1	-6.22e-006	
9	36	0.0235504	9.032e-008	1	1.76e-010	Hessian modified

Optimization terminated successfully:

Search direction less than 2*options.TolX and

maximum constraint violation is less than options.TolCon

Active Constraints:

1

2

x = -9.5474 1.0474

fval = 0.0236

exitflag = 1

output = iterations: 9

funcCount: 38

stepsize: 1

algorithm: 'medium-scale: SQP, Quasi-Newton, line-search'

firstorderopt: []

cgiterations: []

例 5：在上例的基础上，再加上边界约束条件，即加上 $x_1 \geq 0$， $x_2 \geq 0$，则我们仅需要修改上面的第三个程序为：

```
clear
x0=[-1 1];
lb=[0,0];
ub=[];
options=optimset('largescale','off','display','iter');
[x,fval,exitflag,output]=fmincon(@objfun,x0,[],[],[],[],lb,ub,@confun,options)
```

现在得到的结果为：

Iter	F-count	f(x)	constraint	max Step-size	Directional derivative	Procedure
1	3	5.0009	0.5	1	3	
2	7	8.5004	1.355e-020	1	-0.0004	
3	11	8.5	3.04e-013	1	2.43e-012	Hessian modified

Optimization terminated successfully:

Search direction less than 2*options.TolX and

maximum constraint violation is less than options.TolCon

Active Constraints:

1

3

x =   0   1.5000

fval = 8.5000

exitflag =   1

output =   iterations: 3

funcCount: 13

stepsize: 1

algorithm: 'medium-scale: SQP, Quasi-Newton, line-search'

firstorderopt: []

cgiterations: []

三、实验练习

1．将例 1 中 x 的范围改为[-5,5]你将得到怎样的结果，你认为正确吗？应该如何解决？

2．求函数 $f(x)=x^2+4x+4$ 的最小值。

3．在区间 $[-10,10]$ 上，求函数 $f(x)=(x-2)^4\sin x-(x-1)^2\cos x$ 的最小值。

4．求有约束的非线性优化问题：

$$\min f(x)=\frac{1}{3}(x_1+1)^3+x_2$$

约束条件为：$\begin{cases} x_1-1\geq 0 \\ x_2\geq 0 \end{cases}$

5．求有约束的非线性优化问题：

$$\min f(x)=2x_1^2+2x_2^2-2x_1x_2-4x_1-6x_2$$

约束条件为：$\begin{cases} x_1+x_2\leq 2 \\ x_1+5x_2\leq 5 \\ x_1\geq 0, x_2\geq 0 \end{cases}$。