《最优化方法》(修订版)解习题解答(学堂室团队自编答案).

第1章 最优化问题概述.

1.1：解：设第 j 种食品购进 x_j (kg), $j=1,2\cdots n$.

则要求总费用：

$\sum_{j=1}^{n} c_j x_j$ 达到最小，其中要满足的约束条件为：

$x_1 \le d_1$, $x_2 \le d_2$; $x_3 \ge d_3$; $x_j \ge 0$, $j=1,2\cdots n$.

$\sum_{j=1}^{n} a_{ij} x_j \ge 1000 b_i$, $i=1,2\cdots m$.

综上，可把所求得的线性规划问题化为以下数学模型：

$$\begin{cases} \min \sum_{j=1}^{n} c_j x_j \\ s.t.\ \sum_{j=1}^{n} a_{ij} x_j \ge 1000 b_i,\ i=1,2\cdots m. \\ \quad x_1 \le d_1; \\ \quad x_2 \le d_2; \\ \quad x_3 \ge d_3; \\ \quad x_j \ge 0,\ j=1,2\cdots n. \end{cases}$$

1.2：解：设第 j 号货物装载 x_j 件.

则要求最大价值：$50x_1+100x_2+150x_3+100x_4+250x_5+250x_6$.

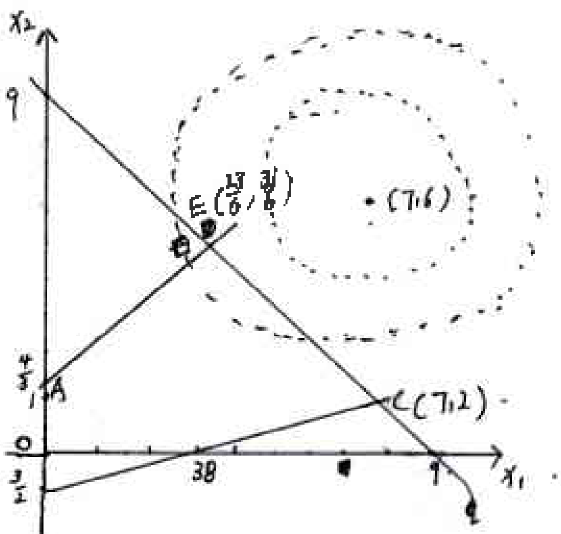
的最大值等价于：$\min z = -(50x_1+100x_2+150x_3+100x_4+250x_5+250x_6)$

则问题的数学模型为：

$$\begin{cases} \min z = -(50x_1+100x_2+150x_3+100x_4+250x_5+250x_6) \\ s.t.\ 20x_1+5x_2+10x_3+12x_4+25x_5+50x_6 \le 400000. \\ \quad 1x_1+2x_2+4x_3+3x_4+2x_5+5x_6 \le 50000 \\ \quad 1x_1+3x_4 \le 10000 \\ \quad 0.1x_1+0.2x_2+0.4x_3+0.1x_4+0.3x_5+0.9x_6 \le 750 \\ \quad x_j \in Z^+,\ j=1,2,\cdots 6. \\ \quad x_j \ge 0,\ j=1,2,\cdots 6. \end{cases}$$

1.3．解：由题可得可行域D为，D为多边形 $AOBCE$ 围成区域包括边界.

目标函数值 $f(x)=4$, $f(x)=9$ 的等值线是以 $(7,6)^T$ 为圆心的两个同心圆，如图所示：



1.4. 证明：当$n=1$时，$F_1=\frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}-\frac{1-\sqrt{5}}{2}\right)=1$，命题成立；

假设$n<k$时命题成立，则：

$$F_k=F_{k-1}+F_{k-2}=\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k-1}-\left(\frac{1-\sqrt{5}}{2}\right)^{k-1}\right]+\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k-2}-\left(\frac{1-\sqrt{5}}{2}\right)^{k-2}\right]$$

$$=\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k-1}+\left(\frac{1+\sqrt{5}}{2}\right)^{k-2}-\left(\frac{1-\sqrt{5}}{2}\right)^{k-1}-\left(\frac{1-\sqrt{5}}{2}\right)^{k-2}\right]$$

$$=\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k-1}\cdot\left(1+\frac{\sqrt{5}-1}{2}\right)-\left(\frac{1-\sqrt{5}}{2}\right)^{k-1}\cdot\left(1+\frac{-1-\sqrt{5}}{2}\right)\right]$$

$$=\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k-1}\cdot\left(\frac{\sqrt{5}+1}{2}\right)-\left(\frac{1-\sqrt{5}}{2}\right)^{k-1}\cdot\left(\frac{1-\sqrt{5}}{2}\right)\right]$$

$$=\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k}-\left(\frac{1-\sqrt{5}}{2}\right)^{k}\right]$$

即$n=k$时命题成立。

从而命题对一切自然数成立。

1.5. 黄金分割法：

```
%huangjinfenge.m
function [xm,fm]=huangjinfenge(f,a,b,eplison)
% [a,b]为要求f(x)最小值的区间，eplison为精度要求，函数huangjinfenge的作用
是利用黄金分割法求f(x)在给定区间的最小值.
x2=a+0.618*(b-a); f2=feval('f',x2);          %取试探点x2
x1=a+0.382*(b-a); f1=feval('f',x1);          %取试探点x1
while abs(b-a)>eplison
  if f1<f2                                   %当f1<f2时，将原区间缩为[a,x2]
     b=x2; x2=x1; f2=f1;
  else if f1==f2                             %当f1=f2时，原区间缩为[x1,x2]
     a=x1; b=x2;
     x2=a+0.618*(b-a); f2=feval('f',x2);
     x1=a+0.382*(b-a); f1=feval('f',x1);
     else                                    %当f1>f2时，将原区间缩为[x1,b].
       a=x1; x1=x2; f1=f2;
       x2=a+0.618*(b-a); f2=feval('f',x2);
     end
  end
end
xm=(a+b)/2;
fm=feval('f',xm);
```

平分法：

```
%pingfen.m
function [xm,fm]=pingfen(f,a,b,eplison)
c=(a+b)/2;
While b-a>eplison
    df=diff(f);            %求f(x)的导数
    t=feval('df',c);       %f(x)的导数为零，即满足取极值条件
  if t==0
    xm=c;                  %f(x)的导数小于零，取[c,b]为新的区间.
  else if t<0
        a=c;               %f(x)的导数大于零，取[a,c]为新的区间
      else b=c;
      end.
  end
  c=(a+b)/2;
end
xm=c;fm=feval('f',xm);
```

1.6:解：计算法可参照P19.

~~函数f(x)在[-1,1]上为下单峰函数，ε=0.1.~~

~~取$x_1=a+0.382(b-a)=-0.236$,~~

~~$x_2=a+0.618(b-a)=0.236$.~~

编程法：(调用1.5题的hangjinfenge.m).

```
%f.m
function y=f(x)
y=exp(-x)+x^2;
>> [xm,fm]=hangjinfenge('f',-1,1,0.1)
xm =
      0.3393.
fm = ~~0.8274~~
      0.8274
```

1.7. 解:先给出Fibonacci法和二次插值法的源程序:

Fibonacci法:

```
%fib.m
function y=fib(x)
y=(power((1+sqrt(5))/2,x+1)-power((1-sqrt(5))/2,x+1))/sqrt(5);
%fibonacci.m
function [xm,fm]=fibonacci(f,a,b,eplison)

n=0;
while feval('fib',n)<(b-a)/eplison                %求试探点的个数n
   n=n+1;
end
x1=a+feval('fib',n)<(b-a)/eplison
x1=a+feval('fib',n-2)/feval('fib',n)*(b-a);       %取试探点x1
f1=feval('f',x1);
x2=a+feval('fib',n-1)/feval('fib',n)*(b-a);       %取试探点x2
f2=feval('f',x2);

for k=1:n-3
   if f1<f2                                       %f1<f2,原区间缩短为[a,x2]
      b=x2; x2=x1; f2=f1;
      x1=a+feval('fib',n-k-2)/feval('fib',n-k)*(b-a);
      f1=feval('f',x1);
   else a=x1; x1=x2; f1=f2;                       %f1>=f2,原区间缩为[x1,b]
      x2=a+feval('fib',n-k-1)/feval('fib',n-k)*(b-a);
      f2=feval('f',x2);
   end
end
if f1<f2
   b=x2; x2=x1; f2=f1;
else a=x1;
end
x1=x2-0.1*(b-a); f1=feval('f',x1);                %写出最后一次迭代
if f1<f2
   xm=(a+x2)/2;
else if f1==f2
      xm=(x1+x2)/2;
   else xm=(x1+b)/2;
   end
end
fm=feval('f',xm);
```

抛物线法（二次插值法）

```
%paowuxian.m
function [xm,fm]=paowuxian(f,x0,x1,x2,eplison)
while abs(x1-x2)>eplison
    f1=feval('f',x1);f0=feval('f',x0);f2=feval('f',x2);
    xb=0.5*((x2^2-x0^2)*f1+(x1^2-x2^2)*f0+(x0^2-x1^2)*f2)/((x2-x0)*f1+(x1-x2)*f0+(x0-x1)*f2+eps);
    fb=feval('f',xb);
      if f0-fb<0
        if x0<xb                                   %f0<fb,x0<xb,取x1,x0,xb为新的二次插值点.
          x2=xb;f2=fb;
        else x1=xb;f1=fb;                          %f0<fb,xb<x0,取xb,x0,x2为新的二次插值点.
        end
      else if f0-fb>0
          if x0>xb                                 %f0>fb,x0<xb,取x0,xb,x2为新的二次插值点.
             x2=x0;x0=xb;f2=f0;f0=fb;
          else x1=x0;x0=xb;
               f1=f0;f0=fb;                        %f0>fb,x0>xb,取x1,xb,x0为新的二次插值点.
          end
          else if x0<xb                            %f0=fb,x0<xb,取x0,xb,(x0+xb)/2为新二次插值点.
               x1=x0;x2=xb;x0=(x1+x2)/2;
               f1=f0;f2=fb;f0=feval('f',x0);
             else if x0>xb                         %f0=fb,x0>xb,取x0,xb,(x0+xb)/2为新的二次插值点.
                  x1=xb;x2=x0;x0=(x1+x2)/2;
                  f1=fb;f2=f0;f0=feval('f',x0);
                  else xj=(x1+x0)/2;fj=feval('f',xj);   %f0=fb,x0=xb.
                     if fj<f0
                       x2=x0;x0=xj;f2=f0;f0=fj;
                     else if fj>f0
                          x1=xj;f1=fj;
                        else x1=xj;x2=x0;x0=(x1+x2)/2;
                           f1=fj;f2=f0;f0=feval('f',x0);
                        end
                     end
                  end
               end
             end
          end
      end
xm=x0;fm=f0;
```

① 用Fibonacci法求：

```
%f.m
function y=f(x)
y=x^4 +2*4 + 4;
>> [xm,fm]=fibonacci('f',-1,0,0.01)
xm =
    -0.7951

fm =
    2.8095.
```

② 二次插值法求：

```
%f.m.
function y=f(x)
y=x^4+2*x +4;
>> [xm,fm]=paowuxian('f',-0.7,-1.0,0.01)
xm =
    -0.7937
fm =
    2.8094.
```

1.8. 解：构造可行下降方向 $P_0=\frac{\pi}{2}^T$.

$\nabla\varphi(t)=\cos(t+\pi)=-\cos t$

$\varphi_0=\varphi(t_0)=1$，$g_0=0^T$，$g_0^TP_0=0$.

step1：给定 $u=0.1$. $\delta=0.7$，令 $a=0$. $b=+\infty$，$\alpha=1$，$j=0$

step2：$t_1=t_0+\alpha P_0=2\pi$，$\varphi_1=\varphi(2\pi)=0$，$g_1=-1$.

$\varphi_0-\varphi_1=1\geq-u\alpha g_0^TP_0=0$，满足(1.6)式.

然而 $\nabla\varphi_1^TP_0=-\frac{\pi}{2}<0$ 不满足(1.7)式，$\underset{令j=j+1}{转}$ step3.

step3：令 $a=\alpha=1$，$\alpha=\min\{2\alpha,\frac{\alpha+b}{2}\}=2$，转step2. 重新计算 φ_1.

计算过程如下表：

j	[illegible]	φ_0	α	t_1	φ_1	条件(1.6)	条件(1.7)
0	$\frac{3}{2}\pi^T$	1	1	$2\pi^T$	0	成立.	不成立
1	$\frac{3}{2}\pi^T$	1	2	$\frac{5}{2}\pi^T$	-1	成立.	成立

则迭代两次可得出满足Wolfe条件的步长 $\alpha_0=2$.

则 $\varphi_1=\varphi_0+\alpha_0P_0=\frac{5}{2}\pi^T$.

1.9. 程序:

```
%wolfe0.m
function [alpha,xm,fm]=wolfe0(x0,p0,mu,sigma)
% x0为搜索的初始点,p0为搜索方向,mu,sigma为控制因子,alpha为返回的搜索步长,
% (xm,fm)为返回的最小值,函数wolfe0.m的作用是利用wolfe原则进行一维不确定搜索函数最小值.
syms x1 x2;
y=100*(x2-x1^2)^2+(1-x1)^2;                         %定义Rosenbrock函数.
y1=diff(y,'x1');
y2=diff(y,'x2');
a=0;b=inf;alpha=1;
f0=subs(y,{x1,x2},{x0(1),x0(2)});
g0=[subs(y1,{x1,x2},{x0(1),x0(2)}),subs(y2,{x1,x2},{x0(1),x0(2)})]';
xk=subs(y,{x1,x2},{xk(1),xk(2)});
g1=[subs(y1,{x1,x2},{xk(1),xk(2)}),subs(y2,{x1,x2},{xk(1),xk(2)})]';
t=g0'*p0;
while (f0-fk)<(-mu*alpha*t)|(g1'*p0<sigma*t)     %利用wolfe原则计算.
  if (f0-fk)<(-mu*alpha*t)
     b=alpha; alpha=(a+alpha)/2;
  else if g1'*p0<sigma*t
        a=alpha;alpha=min(2*alpha,(alpha+b)/2);
        end
  end
  xk=x0+alpha.*p0;
  fk=subs(y,{x1,x2},{xk(1),xk(2)});
  g1=[subs(y1,{x1,x2},{xk(1),xk(2)}),subs(y2,{x1,x2},{xk(1),xk(2)})]';
  end
  xm=xk;fm=fk;
```

对于本题:

```
>> [alpha,xm,fm]=wolfe0([-1,1]',[1,1]',0.1,0.5)
alpha =
   0.0039
 xm =
   -0.9961
    1.0039
 fm =
   3.9981
```