# 第八章 数据处理和回归分析

## 用线性方程 Ax+B 来描述一组数据

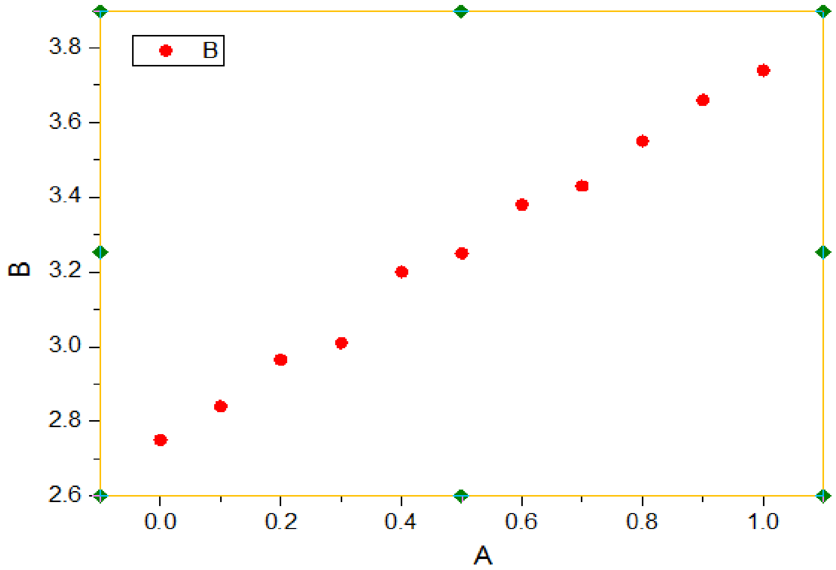
### 六、例

给定 11 个观测值如下：

| $x$ | 0.0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $y$ | 2.75 | 2.84 | 2.965 | 3.01 | 3.20 | 3.25 | 3.38 | 3.43 | 3.55 | 3.66 | 3.74 |

求回归系数 A 与 B、偏差平方和 Q、平均标准偏差 S、回归平方和 P、偏差最大值 UMAX，偏差最小值 UMIN、偏差平均值 U。

主程序（文件名：ISQT10.FOR）为



# 一元线性回归分析

## 一、功能

对于给定的 $n$ 个数据点$(x_i, y_i)(i=1,2,\cdots,n)$用直线 $y=ax+b$ 作回归分析。

## 二、方法说明

设随机变量 $y$ 随自变量 $x$ 变化。已知 $n$ 组观测数据$(x_i, y_i)$ $(i=1,2,\cdots,n)$,用直线

$$y=ax+b$$

作回归分析。其中 $a$ 与 $b$ 为回归系数。

为确定回归系数 $a$ 与 $b$,一般采用最小二乘法,即使得

$$q=\sum_{i=1}^{n}[y_i-(ax_i+b)]^2$$

达到最小。根据极值原理,$a$ 与 $b$ 满足下列方程:

$$\frac{\partial q}{\partial a}=2\sum_{i=1}^{n}[y_i-(ax_i+b)](-x_i)=0$$

$$\frac{\partial q}{\partial b}=2\sum_{i=1}^{n}[y_i-(ax_i+b)](-1)=0$$

从而解得

$$a=\frac{dxy}{dx},\ b=\overline{y}-a\overline{x}$$

其中

$$\overline{x}=\sum_{i=1}^{n}\frac{x_i}{n},\ \overline{y}=\sum_{i=1}^{n}\frac{y_i}{n}$$

$$dx=\sum_{i=1}^{n}(x_i-\overline{x})^2,\ dxy=\sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y})$$

最后计算出以下几个量：

（1）偏差平方和

$$q=\sum_{i=1}^{n}[y_i-(ax_i+b)]^2$$

（2）平均标准偏差

$$s=\sqrt{q/n}$$

（3）回归平方和

$$p=\sum_{i=1}^{n}[(ax_i+b)-\overline{y}]^2$$

（4）最大偏差

$$umax=\max_{1\leqslant i\leqslant n}|y_i-(ax_i+b)|$$

（5）最小偏差

$$umin=\min_{1\leqslant i\leqslant n}|y_i-(ax_i+b)|$$

(6) 偏差的平均值

$$u = \sum_{i=1}^{n} |y_i - (ax_i + b)|/n$$

三、子程序语句

SUBROUTINE ISQT1(X,Y,N,A,B,Q,S,P,UMAX,UMIN,U)

四、形参说明

X,Y——均匀双精度实型一维数组,长度为N,输入参数。存放N个观测值$(x_i, y_i)$ $(i=1,2,\cdots,N)$。

N——整型变量,输入参数。观测点数。

A,B——均为双精度实型变量,输出参数。回归系数,即线性函数为

$$y = Ax + B$$

Q,S,P——均为双精度实型变量,输出参数。分别为偏差平方和、平均标准偏差与回归平方和。

UMAX,UMIN——均为双精度实型变量,输出参数。分别为最大偏差与最小偏差。

U——双精度实型变量,输出参数。偏差的平均值。

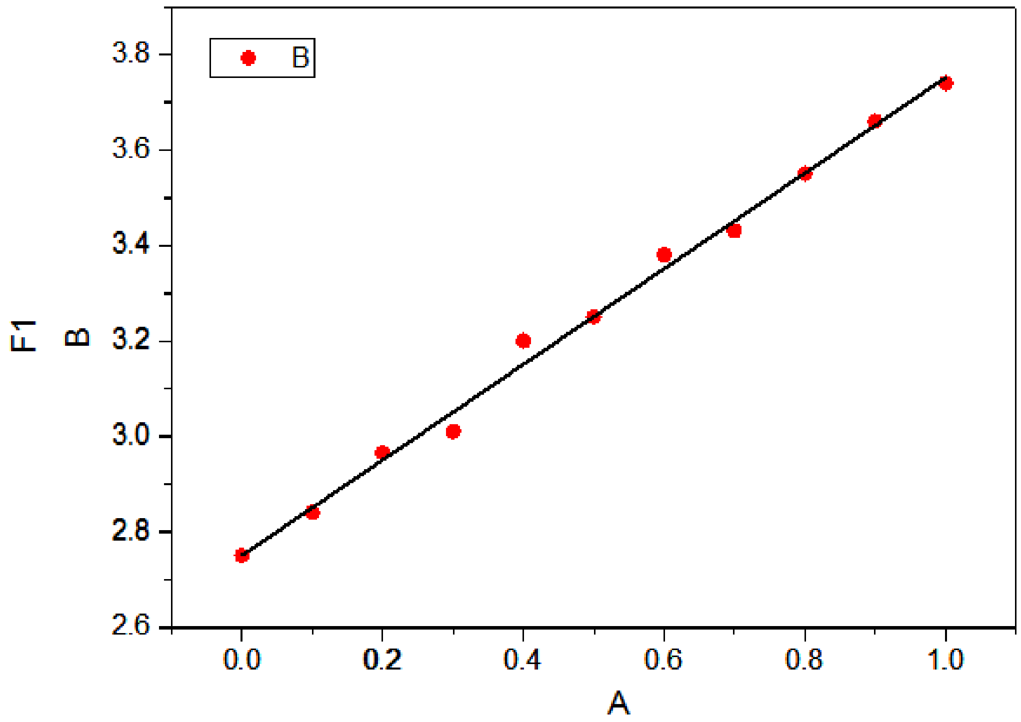
```
PROGRAM EXAMPLES09
      DIMENSION X(11),Y(11)
      DOUBLE PRECISION X,Y,A,B,Q,S,P,UMAX,UMIN,U
      DATA X/0.0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0/
      DATA Y/2.75,2.84,2.965,3.01,3.20,3.25,3.38,3.43,3.55,    &
                   3.66,3.74/
      N=11
      CALL ISQT1(X,Y,N,A,B,Q,S,P,UMAX,UMIN,U)
      WRITE(*,*)
      WRITE(*,10) A,B
10    FORMAT(1X,'A=',D13.6,3X,'B=',D13.6)
      WRITE(*,*)
      WRITE(*,20) Q,S,P
      WRITE(*,*)
      WRITE(*,30) UMAX,UMIN,U
      WRITE(*,*)
20    FORMAT(1X,'Q=',D13.6,3X,'S=',D13.6,3X,'P=',D13.6)
30    FORMAT(1X,'UMAX=',D13.6,3X,'UMIN=',D13.6,3X,'U=',D13.6)
      END
```

运行结果：

A=.100045D+01　　B=.275205D+1

Q=.586796D-02　　S=.230966D-01　　P=.110100D+01

UMAX=.477728D-01　　UMIN=.204541D-02　　U=.174298D-01



```
SUBROUTINE ISQT1(X,Y,N,A,B,Q,S,P,UMAX,UMIN,U)
      DIMENSION X(N),Y(N)
      DOUBLE PRECISION X,Y,A,B,Q,S,P,UMAX,UMIN,U
      DOUBLE PRECISION XX,YY,DX,DXY
      XX=0.0
      YY=0.0
      DO 10 I=1,N
         XX=XX+X(I)/N
         YY=YY+Y(I)/N
10    CONTINUE
      DX=0.0
      DXY=0.0
      DO 20 I=1,N
         Q=X(I)-XX
         DX=DX+Q*Q
```

```
      DXY=DXY+Q*(Y(I)-YY)
20  CONTINUE
    A=DXY/DX
    B=YY-A*XX
    Q=0.0
    U=0.0
    P=0.0
    UMAX=0.0
    UMIN=1.0D+30
    DO 30 I=1,N
      S=A*X(I)+B
      Q=Q+(Y(I)-S)*(Y(I)-S)
      P=P+(S-YY)*(S-YY)
      DX=ABS(Y(I)-S)
      IF (DX.GT.UMAX) UMAX=DX
```

```
      IF (DX.LT.UMIN) UMIN=DX
      U=U+DX/N
30    CONTINUE
      S=SQRT(Q/N)
      RETURN
      END
```

# 第九章 极值问题

1. 定义：

单形调优法(simplex evolutionary method 一种多维直接搜索法。

指先给定多维空间中一个初始单纯形，求出单纯形各个顶点的目标函数值，并加以比较，丢掉其中最坏的点，代之以新点，从而构成一新的单纯形. 如此迭代下去，逐步逼近最优点。

此法于 1962 年由斯本得莱，赫斯蒂，海姆斯瓦尔斯等人提出，1965 年涅尔得和梅得两人加以改进。

**2. 问 题:**

$$\min_{x \in R^n} f(x)$$

$f(x)$是$R^n$上连续函数

即$f(x) \in C(R^n)$

# 3. 算法思想

(1) 集合迭代的思想。

$$S_0 \to S_1 \to \cdots \to S_k \to S_{k+1} \to \cdots$$

这里 $S_i(i=1,2,\cdots)$为单纯形

(2) 下降迭代的思想。

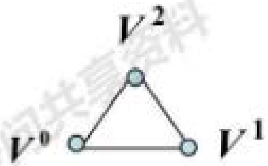
使$S_i$中顶点的目标函数值下降。

## 4. 单纯形概念

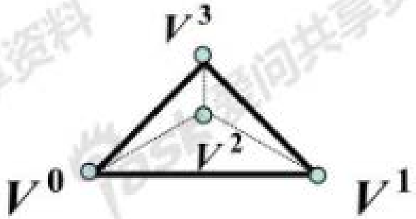
### (1) 例：

$R^1$： 线段


$R^2$： 三角形




$R^3$： 四面体



## （2）单纯形的定义

设$V^0, V^1, \cdots, V^n \in R^n$，如果 $V^j - V^0 \ (j=1,2,\cdots,n)$ 线性无关，

则$V^0, V^1, \cdots, V^n$的凸组合称为由$V^0, V^1, \cdots, V^n$构成的单纯形，记为$S=[V^0, V^1, \cdots, V^n]$，即

$$S=[V^0, V^1, \cdots, V^n]$$
$$=\{x \mid \sum_{j=0}^{n}\alpha_j V^j, \text{其中} \sum_{j=0}^{n}\alpha_j = 1, \alpha_j > 0\}$$

称 $V^0, V^1, \cdots, V^n$为该单纯形的顶点。

## 5. 如何构造单纯形?

对于给定的点$x^0$和正数$\delta$，有两种方式构造单纯形：

(1) $V^0 = x^0$

$$V^1 = V^0 + \delta e_1 \qquad V^1 - V^0 = \delta e_1$$

$$V^2 = V^1 + \delta e_2 \qquad V^2 - V^0 = \delta(e_1 + e_2)$$

$$\cdots\cdots \qquad \cdots\cdots$$

$$V^n = V^{n-1} + \delta e_n \qquad V^n - V^0 = \delta(e_1 + e_2 + \cdots e_n)$$

则 $S = [V^0, V^1, \cdots, V^n]$构成一个单纯形 。

下面以三维空间为例，寻找相应函数的最小值

例 设$V^0=(1,0,1)^T,\delta=\dfrac{1}{2}$。

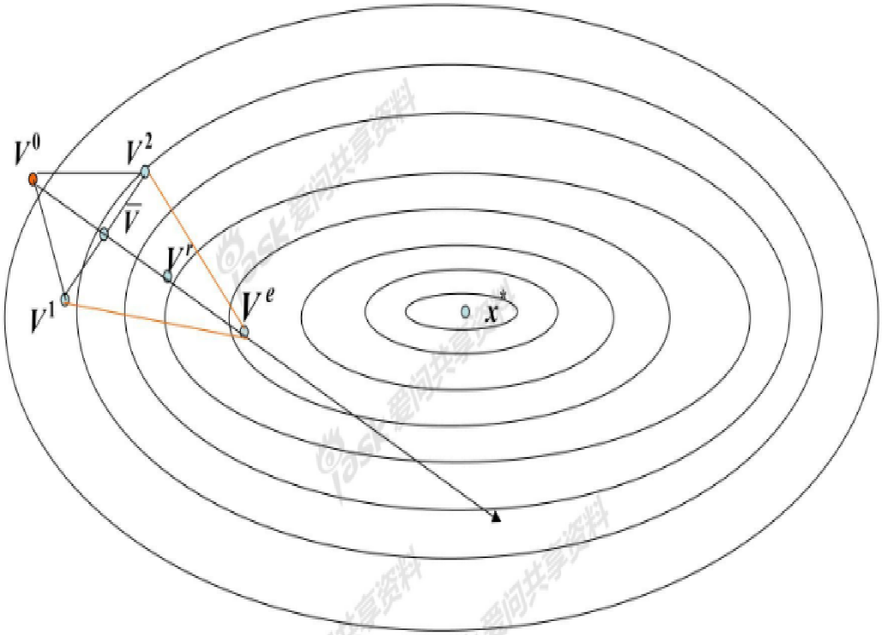
则 $$V^1=V^0+\delta e_1=(1,0,1)^T+\frac{1}{2}(1,0,0)^T=(\frac{3}{2},0,1)^T,$$

$$V^2=V^1+\delta e_2=(\frac{3}{2},0,1)^T+\frac{1}{2}(0,1,0)^T=(\frac{3}{2},\frac{1}{2},1)^T,$$

$$V^3=V^2+\delta e_3=(\frac{3}{2},\frac{1}{2},1)^T+\frac{1}{2}(0,0,1)^T=(\frac{3}{2},\frac{1}{2},\frac{3}{2})^T。$$

则 $S=[V^0,\ V^1,\ V^2,\ V^3]$构成一个单纯形 。

## 6. 单纯形调优法的几何解释




给定单纯形 $S=[V^0, V^1,\cdots,V^n]$。

设 $f(V^h)=\max\{f(V^0),f(V^1),\cdots,f(V^n)\}$， $f(V^s)=\max\limits_{0\le i\le n,i\ne h} f(V^i)$ 。

$f(V^l)=\min\{f(V^0),f(V^1),\cdots,f(V^n)\}$，

令 $\bar{V}=\frac{1}{n}\sum_{i\ne h}V^i$。

反射步： $V^r=\bar{V}+\alpha(\bar{V}-V^h)$

$V^r$：反射点，$\alpha$：反射系数，一般取 $\alpha=1$。

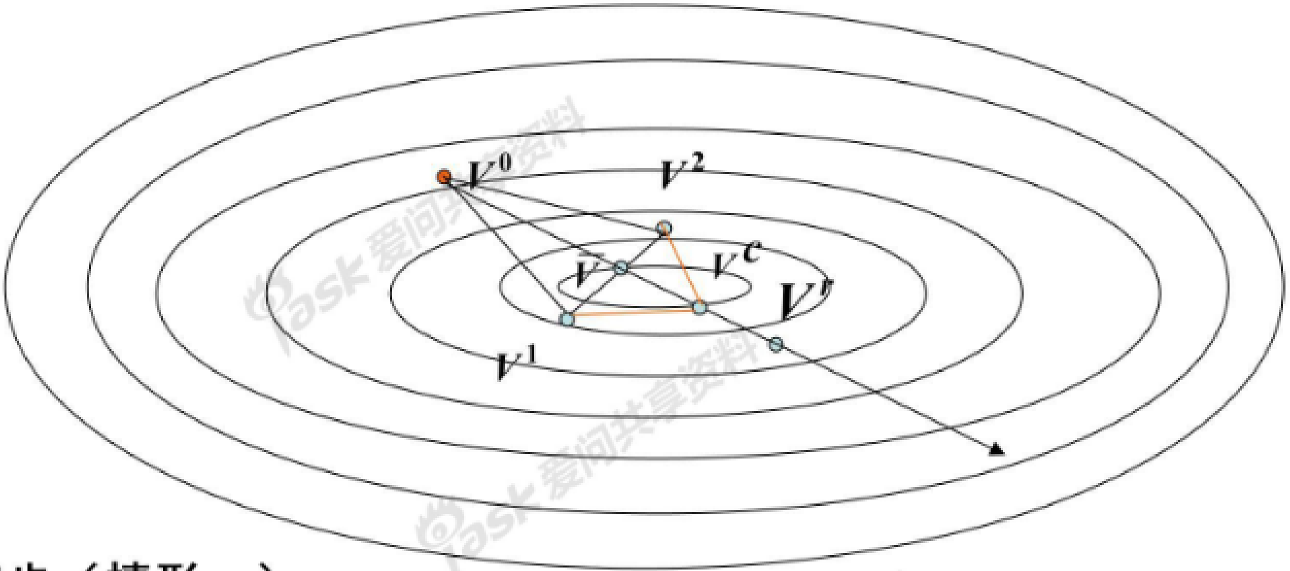
如果 $f(V^r)<f(V^l)$：

延伸步：则令 $V^e=\bar{V}+\beta(V^r-\bar{V})$ ，

$V^e$：延伸点□β：延伸系数□一般取 $\beta=2$。

若 $f(V^e)<f(V^r)$，则用 $V^e$ 代替 $V^h$，构成新的单纯形，

否则用 $V^r$ 代替 $V^h$，构成新的单纯形。

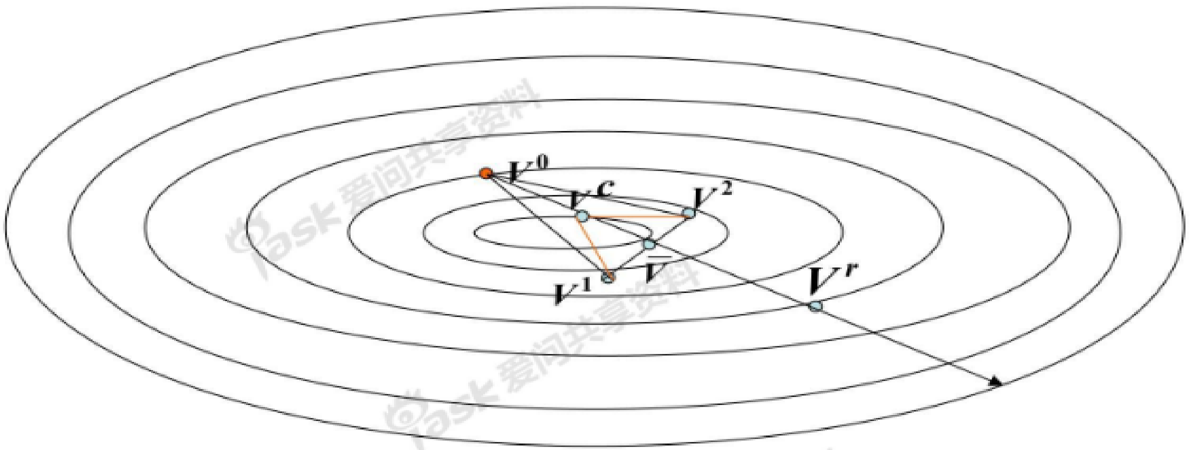



**收缩步（情形一）：**

若 $f(V^s) \le f(V^r) \le f(V^h)$，且用 $V^r$ 替代 $V^h$，会出现循环； 则收缩，令 $V^c = \overline{V} + \gamma(V^r - \overline{V})$，

$V^c$：收缩点， $\gamma$：收缩系数，一般取 $\gamma = \dfrac{1}{2}$。

若 $f(V^c) < f(V^s)$，则用 $V^c$ 代替 $V^h$，构成新的单纯形。




收缩步（情形二）：

若 $f(V^r) > f(V^h)$， 则令 $V^c = \overline{V} + \gamma(V^h - \overline{V})$，

$V^c$：收缩点， $\gamma$：收缩系数，一般取 $\gamma = \frac{1}{2}$。

若 $f(V^c) < f(V^s)$，则用 $V^c$ 代替 $V^h$，构成新的单纯形；
否则棱长减半。

如果 $f(V^c) > f(V^s)$

棱长减半步： $V^i = \frac{V^i + V^l}{2}$ $(\mathrm{i} = 0,1,\cdots,\mathrm{n})$

$S_{k+1} = [V^o, V^1, \cdots, V^n]$。

## 7. 单纯形替换法的步骤

*Step* 1.(初始步)给定初始点$x^0$,构造初始单纯形

$$S^0=[V^o,V^1,\cdots,V^n],\text{精度}\varepsilon>0,\ k:=0$$

*Step* 2.(准备步)计算 $f(V^h)=\max\limits_{0\le i\le n} f(V^i)$, $f(V^l)=\min\limits_{0\le i\le n} f(V^i)$,

$f(V^s)=\max\limits_{0\le i\le n,i\ne h} f(V^i),\overline{V}=\sum\limits_{i\ne h} V^i/n$ 。

*step* 3.(反射步)$V^r=\overline{V}+\alpha(\overline{V}-V^h)$

($V^r$:反射点,$\alpha$:反射系数,一般取 $\alpha=1$)

如果 $f(V^r)<f(V^l)$,则转 *step* 4.

如果 $f(V^r)\ge f(V^s)$,则转 *step* 5.

$V^h:=V^r$,转7.

*step* 4.(延伸步) $V^e = \overline{V} + \beta(\overline{V} - V^h) = \overline{V} + \beta(V^r - \overline{V})$

($V^e$ :□□□ ,$\beta$:□□□□ □□□□ $\beta = 2$)

如果 $f(V^e) < f(V^r)), V^h := V^e$;否则, $V^h := V^r$.

转 *step* 7（判断步）.

*step* 5.(收缩步)

计算$V^{h'} = \arg\min\{f(V^h), f(V^r)\}$,

$V^c = \overline{V} + \gamma(V^{h'} - \overline{V})$($V^c$:收缩步，$\gamma$收缩系数，一般为$\frac{1}{2}$).

若$f(V^c) \le f(V^s), V^h := V^c$,转7

*step* 6.(棱长减半步) $V^i = \dfrac{V^i + V^l}{2}$ $(i = 0,1,\cdots,n)$

$S_{k+1} = [V^o, V^1, \cdots, V^n]$, 转 Step 7(判断步)。

*step* 7.(判断步)计算 $V=\frac{\sum_{i=0}^{n}V^{i}}{n+1}$,

如果 $\sum_{i=0}^{n}\left\|V^{i}-V\right\|\le\varepsilon$（或者 $\max_{1\le i\le n}\left\|V^{i}-V^{j}\right\|\le\varepsilon$),则算法结束,

得到 $x^{*}=V=\frac{\sum_{i=0}^{n}V^{i}}{n+1}$。

**如果** $\sum_{i=0}^{n}\left\|V^{i}-V\right\|>\varepsilon$（$\max_{1\le i\le n}\left\|V^{i}-V^{j}\right\|>\varepsilon$),**那么**

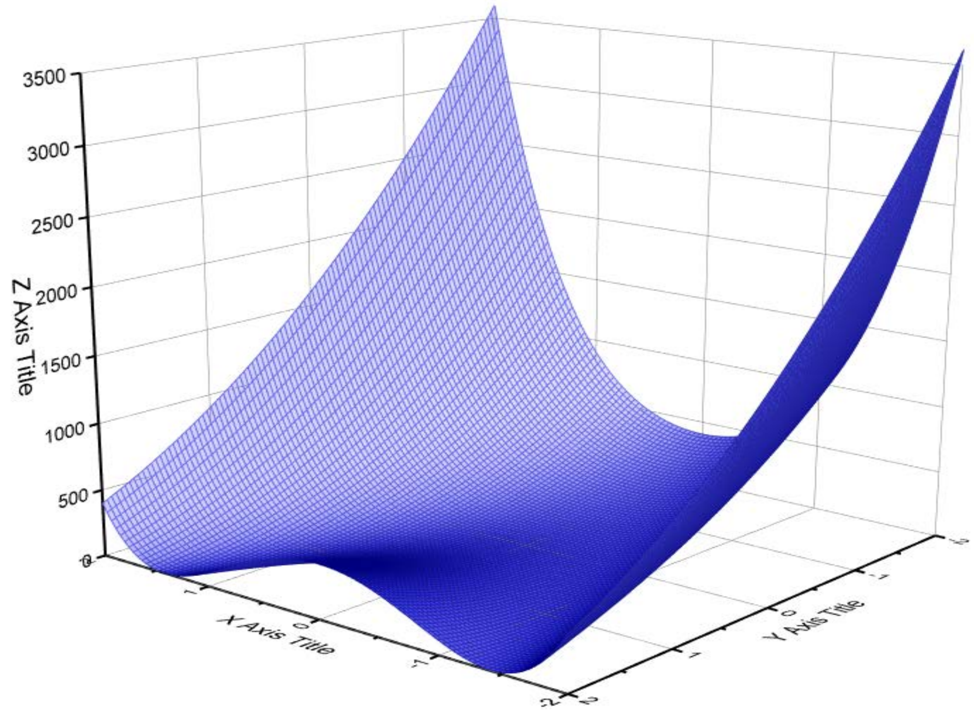
$S_{k+1}=[V^{o},V^{1},\cdots,V^{n}]$, $k=:k+1$, **转 Step 2 (准备步)。**

单纯形法并没有很好的理论性质，即使收敛，收敛也是线性的。但它具有简单实用的优点，计算表明单纯形方法十分可靠，特别地，它能处理函数值变化剧烈的函数。

## 例子：

求目标函数 $J = 100(x_2 - x_1^{\ 2})^2 + (1 - x_1)^2$

的极限值点和极限值，取精度控制在 $10^{-30}$， 初始单形中任意两顶点的距离为 1.0， 扩张系数=1.6，收缩系数为 0.4，变量个数 N=2， 单形顶点数 M=3



# 求 $n$ 维极值的单形调优法

## 一、功能

用单形调优法求解无约束条件下的 $n$ 维极值问题。

## 二、方法说明

设具有 $n$ 个变量的目标函数为

$$J=f(x_1,x_2,\cdots,x_n)$$

单形调优法求目标函数 $J$ 的极小值点的迭代过程如下。

（1）在 $n$ 维变量空间中确定一个由 $n+1$ 个顶点构成的初始单形

$$X_{(i)}=(x_{1i},x_{2i},\cdots,x_{ni}),\quad i=1,2,\cdots,n+1$$

并计算函数值

$$f_{(i)}=f(X_{(i)}),\quad i=1,2,\cdots,n+1$$

（2）确定

$$f_{(R)}=f(X_{(R)})=\max_{1\leqslant i\leqslant n+1}\{f_{(i)}\}$$

$$f_{(G)}=f(X_{(G)})=\max_{\substack{1\leqslant i\leqslant n+1\\ i\neq R}}\{f_{(i)}\}$$

$$f_{(L)}=f(X_{(L)})=\min_{1\leqslant i\leqslant n+1}\{f_{(i)}\}$$

其中 $X_{(R)}$ 称为最坏点。

其中 $X_{(R)}$ 称为最坏点。

(3) 求出最坏点 $X_{(R)}$ 的对称点

$$X_T = 2X_F - X_{(R)}$$

其中

$$X_F = \frac{1}{n} \sum_{\substack{i=1 \\ i \neq R}}^{n+1} X_{(i)}$$

(4) 确定新的顶点代替原顶点，构成一个新的单形。依次按照如下原则进行替代

若 $f(X_T) < f_{(L)}$，则需由下式将 $X_T$ 扩大为 $X_E$

$$X_E = (1+\mu) X_T - \mu X_F$$

其中 $\mu$ 称为扩张系数，一般取 $1.2 < \mu < 2.0$。在这种情况下，如果 $f(X_E) < f_{(L)}$，则

$$X_E \Rightarrow X_{(R)}, \ f(X_E) \Rightarrow f_{(R)}$$

否则

$$X_T \Rightarrow X_{(R)},\ f(X_T) \Rightarrow f_{(R)}$$

若 $f(X_T) \leqslant f_{(G)}$，则 $X_T \Rightarrow X_{(R)},\ f(X_T) \Rightarrow f_{(R)}$。

若 $f(X_T) > f_{(G)}$，如果 $f(X_T) \leqslant f_{(R)}$，则

$$X_T \Rightarrow X_{(R)},\ f(X_T) \Rightarrow f_{(R)}$$

然后由下式将 $X_T$ 缩小为 $X_E$：

$$X_E = \lambda X_{(R)} + (1-\lambda) X_F$$

其中 $\lambda$ 称为收缩系数，一般取 $0.0 < \lambda < 1.0$。在这种情况下，如果 $f(X_E) > f_{(R)}$，则新的单形的 $n+1$ 个顶点为

$$X_{(i)} = \frac{1}{2}(X_{(i)} + X_{(L)}),\ i=1,2,\cdots,n+1$$

且

$$f_{(i)} = f(X_{(i)}),\ i=1,2,\cdots,n+1$$

否则 $X_E \Rightarrow X_{(R)},\ f(X_E) = f_{(R)}$。

重复(2)～(4)，直到单形中各顶点距离小于预先给定的精度要求为止。

如果实际问题中需要求极大值，则只要令目标函数为

$$\tilde{J} = -J = -f(x_1, x_2, \cdots, x_n)$$

即可，此时 $\tilde{J}$ 的极小值的绝对值即为 $J$ 的极大值。

**三、子程序语句**

SUBROUTINE JJSIM (N,M,D,U,V,X,Z,EPS,K,FS,XX,F,XT,XF)

**四、形参说明**

N——整型变量,输入参数。自变量个数。

M——整型变量,输入参数。单形顶点数,M=N+1

D——实型变量,输入参数。初始单形中任意两顶点间的距离。

U——实型变量,输入参数。扩张系数 $\mu$,一般取 $1.2<\mu<2.0$。

V——实型变量,输入参数。收缩系数 $\lambda$,一般取 $0.0<\lambda<1.0$。

X——双精度实型一维数组,长度为 N,输出参数。存放极小值点各坐标值。

Z——双精度实型变量,输出参数。存放极小值。

EPS——实型变量,输入参数。控制精度要求。

K——整型变量,输出参数。迭代次数。如果 K=201,则说明迭代了 200 次还没有满足精度要求,输出的极值点 X 与极值 Z 只作为参考;如果 K≤200,表示正常返回。

FS——双精度实型函数名,输入参数。该函数子程序用于计算目标函数值。在主程序中必须用外部语句对相应的实参进行说明。该函数子程序由用户自编,其语句形式为

FUNCTION FS(N,X)

其中:N 为整型变量,自变量个数;X 为双精度实型一维数组,长度为 N,存放 N 个自变量值。

XX——双精度实型二维数组,体积为 N×M,输出参数。存放最后单形的 M(即 N+1)个顶点坐标

$$(x_{1i},x_{2i},\cdots,x_{ni}),\ i=1,2,\cdots,M$$

F——双精度实型一维数组，长度为 M，输出参数。存放最后单形的 M 个顶点上的目标函数值。

XT，XF——均为双精度实型一维数组，长度为 M。本子程序的工作数组。

```
PROGRAM EXAMPLES10
EXTERNAL FS
 DIMENSION X(2),XX(2,3),F(3),XT(2),XF(2)
 DOUBLE PRECISION X,Z,XX,F,FS,XT,XF
 N=2
 M=3
 D=1.0
 U=1.6
 V=0.4
 EPS=1.0E-30
 CALL JJSIM(N,M,D,U,V,X,Z,EPS,K,FS,XX,F,XT,XF)
 WRITE(*,*)
 WRITE(*,10) K
10  FORMAT(1X,'K=',I4)
 WRITE(*,*)
 WRITE(*,20)
20  FORMAT(7X,'X(1)',11X,'X(2)',11X,'F')
 DO 30 I=1,M
```

```fortran
30  WRITE(*,40) XX(1,I),XX(2,I),F(I)
40  FORMAT(1X,3D15.6)
    WRITE(*,*)
    WRITE(*,50) (I,X(I),I=1,N)
50  FORMAT(1X,'X(',I2,' )=',D15.6)
    WRITE(*,60) Z
60  FORMAT(1X,'Z=',D15.6)
    WRITE(*,*)
    END


    FUNCTION FS(N,X)
    DIMENSION X(N)
    DOUBLE PRECISION X,S,FS
    S=X(2)-X(1)*X(1)
    S=100.0*S*S
    FS=S+(1.0-X(1))*(1.0-X(1))
    RETURN
    END
```

运行结果为

K=　84

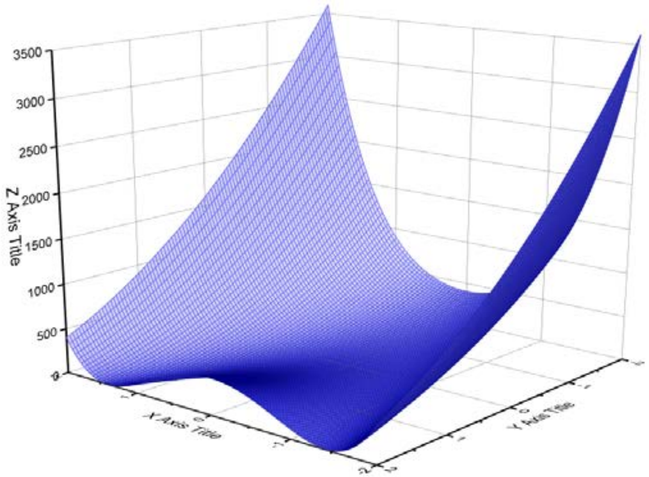
| X(1) | X(2) | F |
|---|---|---|
| .100000D+01 | .100000D+01 | .537804D−15 |
| .100000D+01 | .100000D+01 | .576446D−15 |
| .100000D+01 | .100000D+01 | .257919D−15 |

X( 1 )=　.100000D+01

X( 2 )=　.100000D+01

Z=0.168D-15

本问题的理论极小值点为 x1=1.0　x2=1.0 ，极小值是 0.0



```
SUBROUTINE JJSIM(N,M,D,U,V,X,Z,EPS,K,FS,XX,F,XT,XF)
     DIMENSION X(N),F(M),XX(N,M),XT(N),XF(N)
     DOUBLE PRECISION X,Z,F,XX,XT,XF,FR,FL,FG,FT,FF,FS
     INTEGER R,G
     K=1
     FR=SQRT(1.0D0*M)
     FL=D*(FR-1.0)/(1.414*N)
     FG=D*(FR+N-1.0)/(1.414*N)
     DO 10 I=2,M
     DO 10 J=1,N
10   XX(J,I)=FL
     DO 20 I=2,M
20   XX(I-1,I)=FG
     DO 40 I=1,M
        DO 30 J=1,N
```

```
30     X(J)=XX(J,I)
       F(I)=FS(N,X)
40  CONTINUE
50  FR=F(1)
    FL=F(1)
    R=1
    L=1
    DO 60 I=2,M
       IF (F(I).GT.FR) THEN
          R=I
          FR=F(I)
       END IF
       IF (F(I).LT.FL) THEN
          L=I
          FL=F(I)
```

```
      END IF
60  CONTINUE

    G=1
    FG=F(1)
    J=1
    IF (R.EQ.1) THEN
      G=2
      FG=F(2)
      J=2
    END IF
    DO 70 I=J+1,M
      IF ((I.NE.R).AND.(F(I).GT.FG)) THEN
        G=I
        FG=F(I)
```

```
      END IF
70  CONTINUE
    DO 90 J=1,N
      XF(J)=0.0
      DO 80 I=1,M
        IF (I.NE.R) XF(J)=XF(J)+XX(J,I)/N
80    CONTINUE
      XT(J)=2.0*XF(J)-XX(J,R)
90  CONTINUE
    FT=FS(N,XT)
    IF (FT.LT.F(L)) THEN
      DO 100 J=1,N
100       XF(J)=(1.0+U)*XT(J)-U*XF(J)
      FF=FS(N,XF)
      IF (FF.LT.F(L)) THEN
```

```
          DO 110 J=1,N
110          XX(J,R)=XF(J)
          F(R)=FF
        ELSE
          DO 120 J=1,N
120          XX(J,R)=XT(J)
          F(R)=FT
        END IF
      ELSE IF (FT.LE.F(G)) THEN
        DO 130 J=1,N
130        XX(J,R)=XT(J)
        F(R)=FT
      ELSE
        IF (FT.LE.F(R)) THEN
          DO 140 J=1,N
```

```
140          XX(J,R)=XT(J)
       F(R)=FT
     END IF
     DO 150 J=1,N
150        XF(J)=V*XX(J,R)+(1.0-V)*XF(J)
     FF=FS(N,XF)

     IF (FF.GT.F(R)) THEN
       DO 170 I=1,M
         DO 160 J=1,N
           XX(J,I)=(XX(J,I)+XX(J,L))/2.0
           X(J)=XX(J,I)
160            CONTINUE
         F(I)=FS(N,X)
170          CONTINUE
```

```
        ELSE
          DO 180 J=1,N
180            XX(J,R)=XF(J)
          F(R)=FF
        END IF
      END IF
      FF=0.0
      FT=0.0
      DO 190 I=1,M
        FF=FF+F(I)/M
        FT=FT+F(I)*F(I)
190     CONTINUE
      FT=(FT-M*FF*FF)/N
      IF (FT.GE.EPS) THEN
        K=K+1
```

```
        IF (K.LT.201) GOTO 50
      END IF
      DO 210 J=1,N
        X(J)=0.0
        DO 200 I=1,M
200         X(J)=X(J)+XX(J,I)/M
210     CONTINUE
      Z=FS(N,X)
      RETURN
      END
```