第 5 章　无约束非线性规划

第 1 节　基本概念

第 2 节　最速下降法

第 3 节　一维搜索

第一节　基本概念

一、非线性规划问题的数学模型

非线性规划数学模型：

$$\min f(\mathbf{x}) \qquad (1)$$

$$\text{s.t.}\quad h_i(\mathbf{x}) = 0,\ i = 1,2,\ldots,m \qquad (2)$$

$$g_j(\mathbf{x}) \geq 0,\ j = 1,2,\ldots,l \qquad (3)$$

$\mathbf{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$是$n$维欧氏空间$\mathbb{E}^n$中的向量[1]（点），而$f(\mathbf{x})$、$h_i(\mathbf{x})$、$g_j(\mathbf{x})$至少应有一个为非线性函数。

[1] 所谓“欧氏空间”，是两点距离定义为欧几里得（Euclidean）距离（$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$）的线性空间。

二维非线性规划问题可用作图法求解。

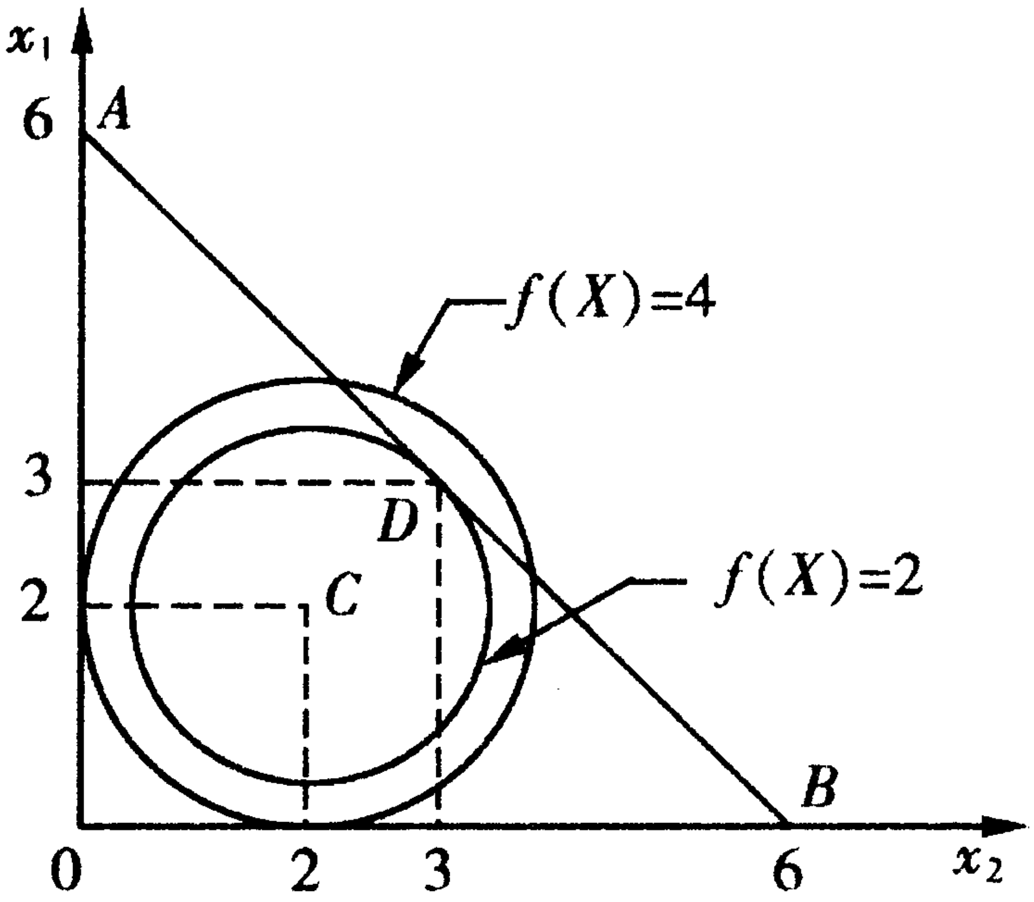
例，

$$\min f(\mathbf{x})=(x_1-2)^2+(x_2-2)^2 \qquad （\mathbf{4}）$$

$$\mathrm{s.t.}\ \ h(\mathbf{x})=x_1+x_2-6=0 \qquad （\mathbf{5}）$$

满足$f(\mathbf{x})=C$的点的集合，一般为一条曲线或一张曲面，称为等值线或等值面。

$(x_1-2)^2+(x_2-2)^2=C$，是一系列以$(2,2)^{\mathrm{T}}$点为圆心、以$\sqrt{C}$为半径的同心圆。



等值线$f(\mathbf{x}) = 2$和约束直线 **AB** 相切，切点 **D** 为最优解：

$$x_1^* = x_2^* = 3$$

最优目标值：

$$f(\mathbf{x}^*) = 2$$

若以约束

$$g(\mathbf{x}) = x_1 + x_2 - 6 \le 0$$

代替约束（**5**），则最优解是$x_1^* = x_2^* = 2$，即 **C** 点，此时$f(\mathbf{x}^*) = 0$。

注 1：如果线性规划问题有最优解，则最优解只能在其可行域边界上达到（特别是在可行域的顶点达到）；

而非线性规划问题最优解（如果最优解存在）则可能在其可行域中的任意一点达到。

二、函数极值

✧ 局部极值

设$f(\mathbf{x})$为定义在n维欧氏空间某一区域$\mathbf{R}$上的n元实函数。对于$\mathbf{x}^* \in \mathbf{R}$，若存在某个$\varepsilon > 0$，使所有与$\mathbf{x}^*$的距离小于$\varepsilon$的$\mathbf{x} \in \mathbf{R}$（即$\mathbf{x} \in \mathbf{R}$且$\|\mathbf{x} - \mathbf{x}^*\| < \varepsilon$）均满足$f(\mathbf{x}) \geq f(\mathbf{x}^*)$，则称$\mathbf{x}^*$为$f(\mathbf{x})$在$\mathbf{R}$上的局部极小点，$f(\mathbf{x}^*)$为局部极小值。

若对于所有$\mathbf{x} \neq \mathbf{x}^*$且与$\mathbf{x}^*$的距离小于$\varepsilon$的$\mathbf{x} \in \mathbf{R}$，严格不等式$f(\mathbf{x}) > f(\mathbf{x}^*)$成立，则称$\mathbf{x}^*$为$f(\mathbf{x})$在$\mathbf{R}$上的严格局部极小点，$f(\mathbf{x}^*)$为严格局部极小值。

✧ 全局极值

若$\mathbf{x}^* \in \mathbf{R}$，对于所有$\mathbf{x} \in \mathbf{R}$都有$f(\mathbf{x}) \geq f(\mathbf{x}^*)$，则称$\mathbf{x}^*$为$f(\mathbf{x})$在$\mathbf{R}$上的全局极小点，$f(\mathbf{x}^*)$为全局极小值。

若对于所有$\mathbf{x} \in \mathbf{R}$且$\mathbf{x} \neq \mathbf{x}^*$，都有严格不等式$f(\mathbf{x}) > f(\mathbf{x}^*)$成立，则称$\mathbf{x}^*$为$f(\mathbf{x})$在$\mathbf{R}$上的严格全局极小点，$f(\mathbf{x}^*)$为严格全局极小值。

✧ 极值条件

定理 1（必要条件） 设$\mathbf{R}$是$\mathbb{E}^n$上的开集，若$f(\mathbf{x})$在$\mathbf{R}$上有一阶连续偏导，且$\mathbf{x}^* \in \mathbf{R}$为其局部极值，则必有

$$\partial f(\mathbf{x}^*)/\partial x_1 = \cdots = \partial f(\mathbf{x}^*)/\partial x_n = 0 \qquad (6)$$

即

$$\nabla f(\mathbf{x}^*) = \mathbf{0} \qquad (7)$$

其中，$\nabla f(\mathbf{x}^*) = (\partial f(\mathbf{x}^*)/\partial x_1, \ldots, \partial f(\mathbf{x}^*)/\partial x_n)^{\mathrm{T}}$为函数$f(\mathbf{x}^*)$在点$\mathbf{x}^*$处的梯度。

注 2：$\nabla f(\mathbf{x})$是$f(\mathbf{x})$的等值面（等值线）在点x处的法线方向，沿这个方向函数值增加最快。

梯度为0的点称为平稳点或驻点；在开集R，极值点必为平稳点，但平稳点不一定是极值点。

定理 2（充分条件） 设R是$\mathbb{E}^n$上的开集，$f(\mathbf{x})$在R上具有二阶连续偏导因而存在 Hesse 矩阵$\mathbf{H}(\mathbf{x})$。对$\mathbf{x}^* \in$ R，若$\nabla f(\mathbf{x}^*) = \mathbf{0}$，且对任何非零向量$\mathbf{z} \in \mathbb{E}^n$，有

$$\mathbf{z}^{\mathrm{T}}\mathbf{H}(\mathbf{x}^*)\mathbf{z} > 0 \qquad (8)$$

则$\mathbf{x}^*$为$f(\mathbf{x})$的严格局部极小点。其中，$\mathbf{H}(\mathbf{x}^*)$为$f(\mathbf{x})$在点$\mathbf{x}^*$处的海赛（Hesse）矩阵。

$$\mathbf{H}(\mathbf{x}^*) = \nabla^2 f(\mathbf{x}^*)$$

$$= \begin{bmatrix} \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1 \, \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1 \, \partial x_n} \\ \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2 \, \partial x_1} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2 \, \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n \, \partial x_1} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n \, \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n^2} \end{bmatrix}$$

Ludwig Otto Hesse

1811-1874，German

例，函数$f(\mathbf{x}) = x_1^2 + 2x_2^3 + x_1x_3 + x_1x_2x_3$，梯度为

$$\nabla f(\mathbf{x}) = (2x_1 + x_3 + x_2x_3, 6x_2^2 + x_1x_3, x_1 + x_1x_2)^{\mathrm{T}}$$

Hesse 矩阵为：

$$\mathbf{H}(\mathbf{x}) = \begin{bmatrix} 2 & x_3 & 1 + x_2 \\ x_3 & 12x_2 & x_1 \\ 1 + x_2 & x_1 & 0 \end{bmatrix}$$

定 2 理的证明需运用多元函数的一阶、二阶 Taylor 展开：

$$f(\mathbf{x}^* + \lambda\mathbf{d}) = f(\mathbf{x}^*) + \lambda\nabla f(\mathbf{x}^*)^{\mathrm{T}}\mathbf{d} + o(\|\lambda\mathbf{d}\|) \quad (9)$$

以及

$$f(\mathbf{x}^* + \lambda\mathbf{d}) = f(\mathbf{x}^*) + \lambda\nabla f(\mathbf{x}^*)^{\mathrm{T}}\mathbf{d} + \frac{1}{2}\lambda^2\mathbf{d}^{\mathrm{T}}\nabla^2 f(\mathbf{x}^*)\mathbf{d} + o(\|\lambda\mathbf{d}\|^2) \quad (10)$$

证明：略。

定理 2 的条件（8）式：$\mathbf{z}^{\mathrm{T}}\mathbf{H}(\mathbf{x}^*)\mathbf{z} > \mathbf{0}$，意思是海塞矩阵 $\mathbf{H}(\mathbf{x}^*)$在$\mathbf{x}^*$处为正定矩阵。

✧ 正定矩阵

矩阵$\mathbf{A} \in \mathbb{R}^{n\times n}$为对称矩阵，若对任意向量$\mathbf{d} \in \mathbb{R}^{n}$，有

$$\mathbf{d}^{\mathrm{T}}\mathbf{A}\mathbf{d} \geq 0$$

则称A为半正定矩阵；若对$\mathbf{d} \neq \mathbf{0}$有严格不等式成立：

$$\mathbf{d}^{\mathrm{T}}\mathbf{A}\mathbf{d} > 0, \mathbf{d} \neq \mathbf{0}$$

则称A为正定矩阵。

若对任意向量$\mathbf{d} \in \mathbb{R}^{n}$，有：$\mathbf{d}^{\mathrm{T}}\mathbf{A}\mathbf{d} \leq 0$，则称 A 为半负定矩阵；若对任意$\mathbf{d} \neq \mathbf{0}$有严格不等式成立：$\mathbf{d}^{\mathrm{T}}\mathbf{A}\mathbf{d} < 0$，则称 A 为负定矩阵。

✓ 如何判断矩阵正定或负定——各阶主子式

矩阵各阶主子式，是指各阶左上角元素构成的行列式：

一阶正；二阶正；三阶正；四阶正；五阶正

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{bmatrix}$$

如果各阶主子式严格大于零，则矩阵正定。

例， $\mathbf{A}=\begin{bmatrix}1 & 1 & 0\\ 1 & 2 & 1\\ 0 & 1 & 4\end{bmatrix}$

$|1|=1>0$， $\begin{vmatrix}1 & 1\\ 1 & 2\end{vmatrix}=1>0$， $\begin{vmatrix}1 & 1 & 0\\ 1 & 2 & 1\\ 0 & 1 & 4\end{vmatrix}=3>0$

因此**A**是正定矩阵。

注意，各阶主子式严格大于零（>0），则矩阵正定；但若有某阶主子式$=0$（≥ 0），则不能保证半正定。

例，

矩阵，$\mathbf{A}=\begin{bmatrix}1 & 1 & -3\\ 1 & 1 & -3\\ -3 & -3 & 5\end{bmatrix}$，一阶主子式 >0，二阶、三阶 $=0$，但它并不是半正定的——因为存在向量：

$$(1,1,1)\begin{bmatrix}1 & 1 & -3\\ 1 & 1 & -3\\ -3 & -3 & 5\end{bmatrix}\begin{bmatrix}1\\ 1\\ 1\end{bmatrix}=-3<0。$$

因此，从主子式 ≥ 0，不能推出矩阵半正定；只能从各阶主子式 >0，推出矩阵正定

若各阶主子式负（< 0）、正（> 0）相间（奇负偶正），则矩阵负定。

一阶负;二阶正;三阶负;四阶正;五阶负

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{bmatrix}$$

例，$\mathbf{A}=\begin{bmatrix}-2 & 2 & 0\\ 2 & -3 & -1\\ 0 & -1 & -5\end{bmatrix}$，一阶：$|-2|=-2<0$，

二阶：$\begin{vmatrix}-2 & 2\\ 2 & -3\end{vmatrix}=2>0$，三阶：$\begin{vmatrix}-2 & 2 & 0\\ 2 & -3 & -1\\ 0 & -1 & -5\end{vmatrix}=-8<0$，

因此矩阵**A**负定。

与半正定情况相同，如果某阶主子式= 0，就无法从主子式的值判断半负定。

例，$\mathbf{A}=\begin{bmatrix}-1 & -1 & 3\\ -1 & -1 & 3\\ 3 & 3 & -5\end{bmatrix}$

一阶 <0、二阶、三阶 $=0$。但是存在向量$(1,1,1)^{\mathrm{T}}$，有：

$$(1,1,1)\begin{bmatrix}-1 & -1 & 3\\ -1 & -1 & 3\\ 3 & 3 & -5\end{bmatrix}\begin{bmatrix}1\\ 1\\ 1\end{bmatrix}=3>0$$

因此，矩阵**A**不是（半）负定的。

✓ 求高阶行列式的值——右上三角阵

以高斯消元将行列式化为右上三角形式，则对角线元素的乘积就是行列式的值。

$$\begin{vmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 4 \end{vmatrix} = \begin{vmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 4 \end{vmatrix} = \begin{vmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 3 \end{vmatrix} = 3$$

$$\begin{vmatrix} 1 & 1 & 2 & 4 \\ 1 & 2 & 1 & 1 \\ 1 & 2 & 4 & -1 \\ -1 & 2 & 3 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 1 & 2 & 4 \\ 0 & 1 & -1 & -3 \\ 0 & 0 & 3 & -2 \\ 0 & 0 & 0 & 58/3 \end{vmatrix} = 58$$

注 3：**定理 2** 中的充分条件（8）并不是必要条件，即：极小点处的 Hesse 矩阵并不一定正定。

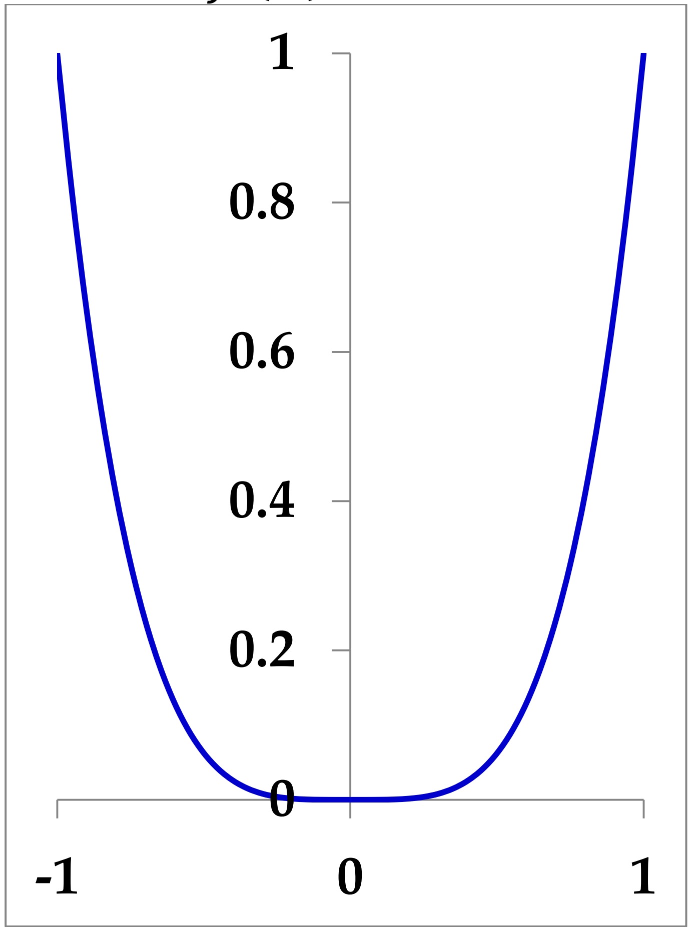
$f(x)=x^4$，极小点$x^*=0$，但$\nabla^2 f(x^*)=0$。

注 4：**定理 2** 中（8）式要成为充分条件，必须是严格不等式成立（正定）。

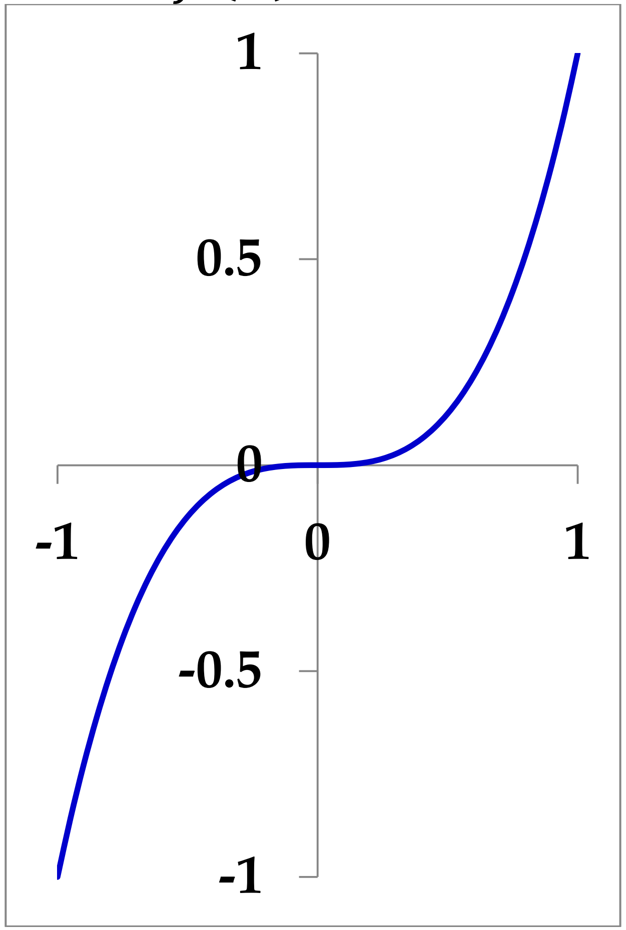
$f(x)=x^3$，在$x=0$处，$\nabla^2 f(x)=0$且$\nabla^2 f(x)=0$（半正定），但$x=0$并不是极值点，只是个平稳点。

而$f(x)=x^3/3-x$在$x=1$处满足定理 2（$\nabla^2 f(x)=2>0$），是极小点。

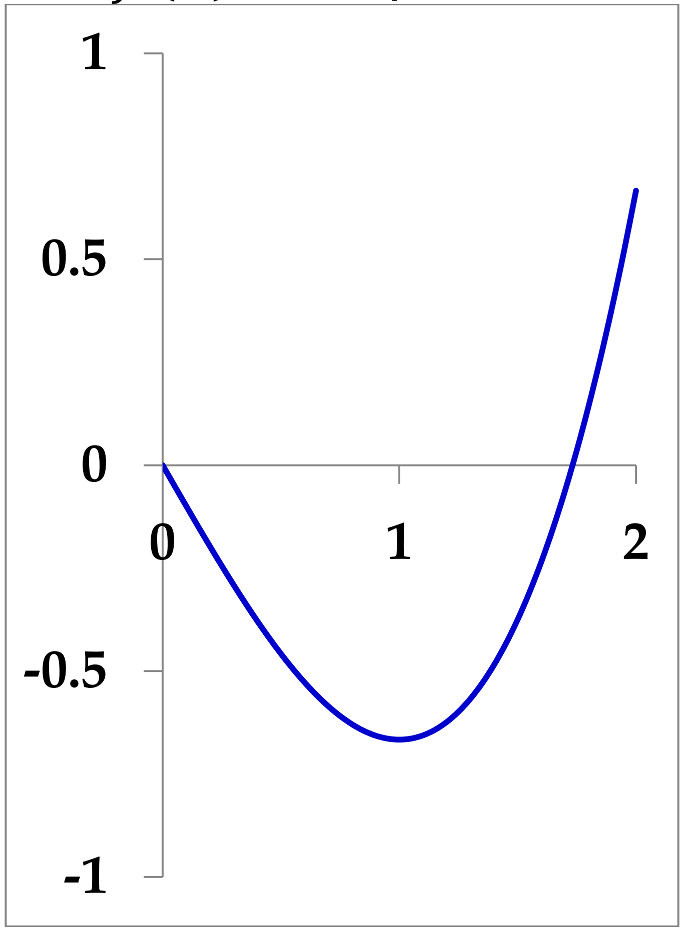
$f(x) = x^4$



$f(x) = x^3$



$f(x) = x^3/3 - x$



三、凸函数与凸规划

✧ 凸函数的定义

设$f(\mathbf{x})$为n维欧氏空间$\mathbb{E}^n$中某个凸集$\mathbf{R}$上的函数，若对任何实数$\alpha\in(0,1)$，以及$\mathbf{x}^{(1)},\mathbf{x}^{(2)}\in\mathbf{R}$，恒有

$$f(\alpha\mathbf{x}^{(1)}+(1-\alpha)\mathbf{x}^{(2)})\leq\alpha f(\mathbf{x}^{(1)})+(1-\alpha)f(\mathbf{x}^{(2)})$$

则称$f(\mathbf{x})$为定义在$\mathbf{R}$上的凸函数。

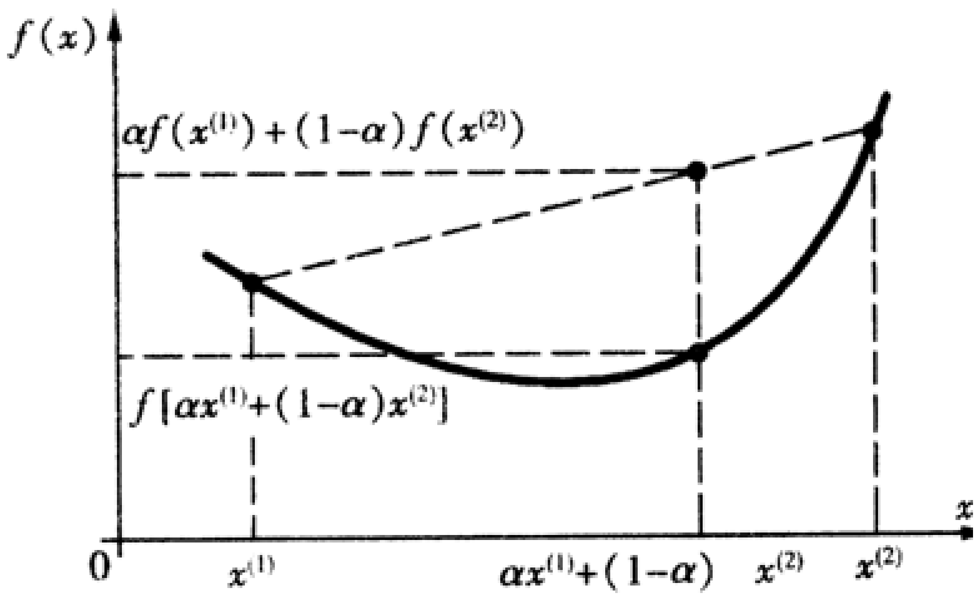
若恒有严格不等式成立：

$$f(\alpha\mathbf{x}^{(1)}+(1-\alpha)\mathbf{x}^{(2)})<\alpha f(\mathbf{x}^{(1)})+(1-\alpha)f(\mathbf{x}^{(2)})$$

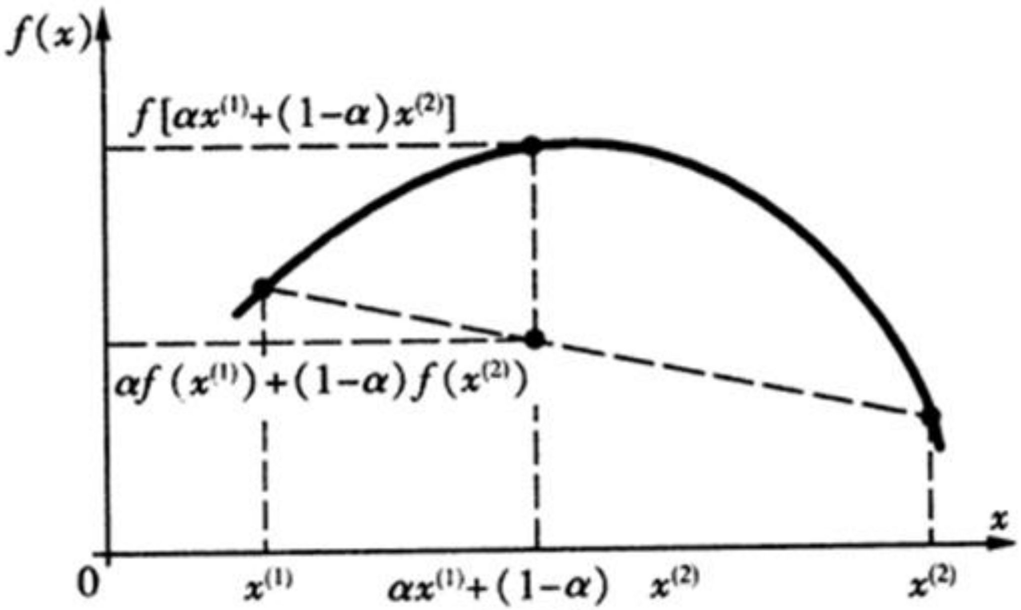
则称$f(\mathbf{x})$为定义在$\mathbf{R}$上的严格凸函数。

若函数$f(\mathbf{x})$是凸函数（严格凸函数），则$-f(\mathbf{x})$是凹函数（严格凹函数）。

凸/凹函数的几何意义：凸函数上任意两点的连线在函数上方；凹函数上任意两点的连线在函数下方。



凸函数



凹函数

线性函数既可看作凸函数，也可看作凹函数。

✧ 凸函数的性质

性质 1：设$f(\mathbf{x})$为定义在凸集**R**上的凸函数，则对任意实数$\beta \geq 0$，函数$\beta f(\mathbf{x})$也是定义在**R**上的凸函数。

性质 2：设$f_1(\mathbf{x})$和$f_2(\mathbf{x})$为定义在凸集**R**上的两个凸函数，则$f(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x})$仍为定义在**R**上的凸函数。

性质 3：设$f(\mathbf{x})$为定义在凸集**R**上的凸函数，则对任一实数β，集合$\mathbf{S}_\beta = \{\mathbf{x}|\mathbf{x} \in \mathbf{R}, f(\mathbf{x}) \leq \beta\}$是凸集（$\mathbf{S}_\beta$称为水平集/截集）。

✧ 凸函数的判定

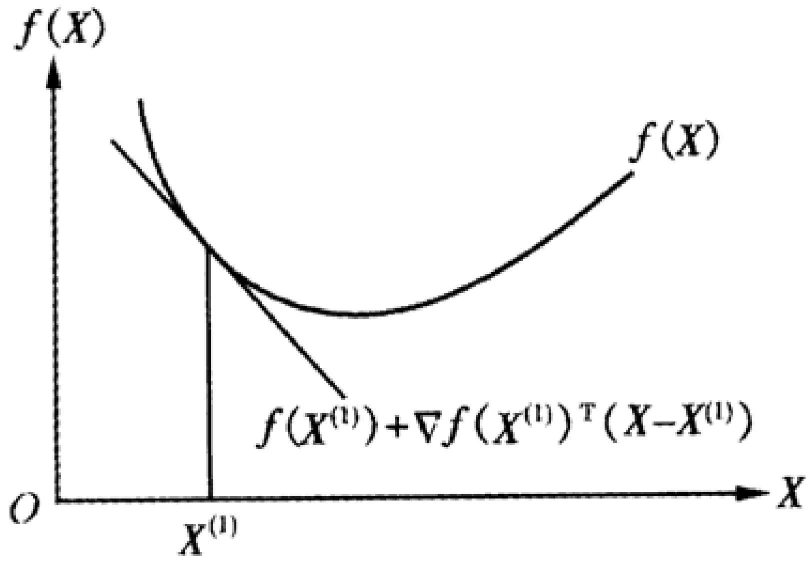
可直接依据凸函数的定义判断是否为凸函数；对于可微函数，可利用下述判别定理。

定理 3（一阶条件） 设$\mathbf{R}$为n维欧氏空间$\mathbb{E}^n$的开凸集，$f(\mathbf{x})$在$\mathbf{R}$上具有一阶连续偏导，则$f(\mathbf{x})$为$\mathbf{R}$上的凸函数的充要条件是，对任意不同两点$\mathbf{x}^{(1)} \neq \mathbf{x}^{(2)} \in \mathbf{R}$，恒有

$$f(\mathbf{x}^{(2)}) \geq f(\mathbf{x}^{(1)}) + \nabla f(\mathbf{x}^{(1)})^{\mathrm{T}}(\mathbf{x}^{(2)} - \mathbf{x}^{(1)})$$

若上式为严格不等式，就是严格凸函数的充要条件。

定理 3 表明，凸函数上某点发出的切线，总是不高于该凸函数——凸函数的切线总是位于函数下方。



定理 4（二阶条件） 设**R**为n维欧氏空间$\mathbb{E}^n$上的开凸集，$f(\mathbf{x})$在**R**上具有二阶连续偏导数，则$f(\mathbf{x})$为**R**上的凸函数的充要条件是：$f(\mathbf{x})$的海赛矩阵$\mathbf{H}(\mathbf{x})$在**R**上处处半正定。[2]

如果处处正定，则是严格凸函数。

[2] 仅在某个点处半正定，则在再该点周围，函数不一定具有凸性，如$f(x)=x^3$在$x=0$附近。

凹函数：如果函数$f(\mathbf{x})$在开凸集**R**上的海赛矩阵处处半负定，则函数是凹函数；

如果函数的海森矩阵负定（奇数阶主子式严格小于零，偶数阶严格大于零），则函数是严格凹函数。

若 **Hesse** 矩阵不定，则函数在所定义的集合上不具有一致的凹性或凸性（在有些区域凹，有些区域凸）。

判断下面各函数是凹函数还是凸函数：

$$f(\mathbf{x}) = e^{x_1+x_2} + x_3^2$$

$$f(\mathbf{x}) = x_1^2 + x_2^2 + x_1x_2$$

$$f(\mathbf{x}) = -x_1^2 - x_2^2 + x_1x_2$$

$$f(\mathbf{x}) = x_1^2 + x_2^2 + 3x_1x_2$$

$$f(\mathbf{x}) = -x_1^2 - x_2^2 - 3x_1x_2$$

$$f(\mathbf{x}) = x_1^2 + x_2^2 + x_3^4$$

$$f(\mathbf{x}) = -x_1^2 - x_2^2 - x_3^4 + x_4^2 + x_1x_2$$

✧ 凸函数的极值

函数局部极小并不一定是其全局最小（全局极小值）。但对定义在凸集上的凸函数，其极小值就是全局最小。

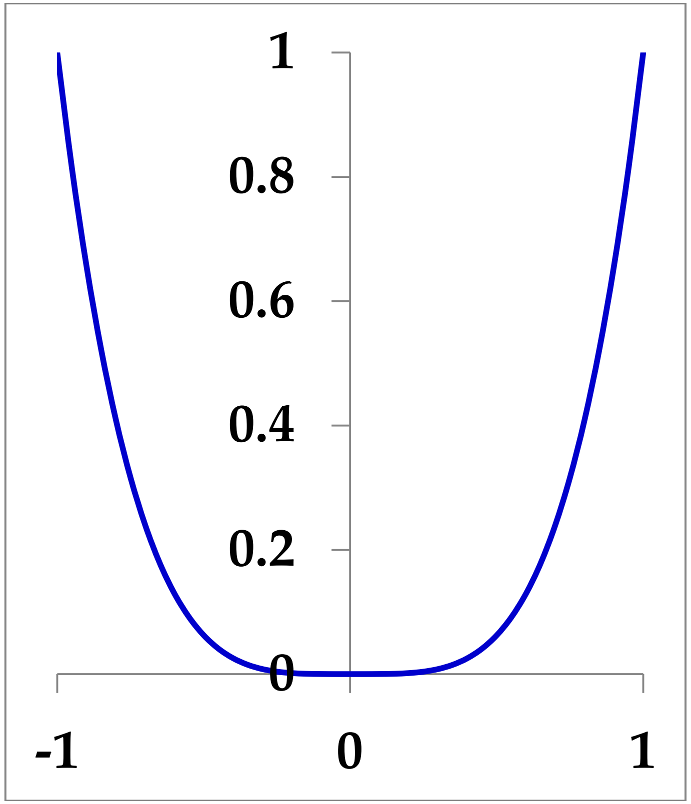
定理 5　若$f(\mathbf{x})$是凸集R上的凸函数，则其任一极小点就是它在R上的最小点，且极小点形成一个凸集。

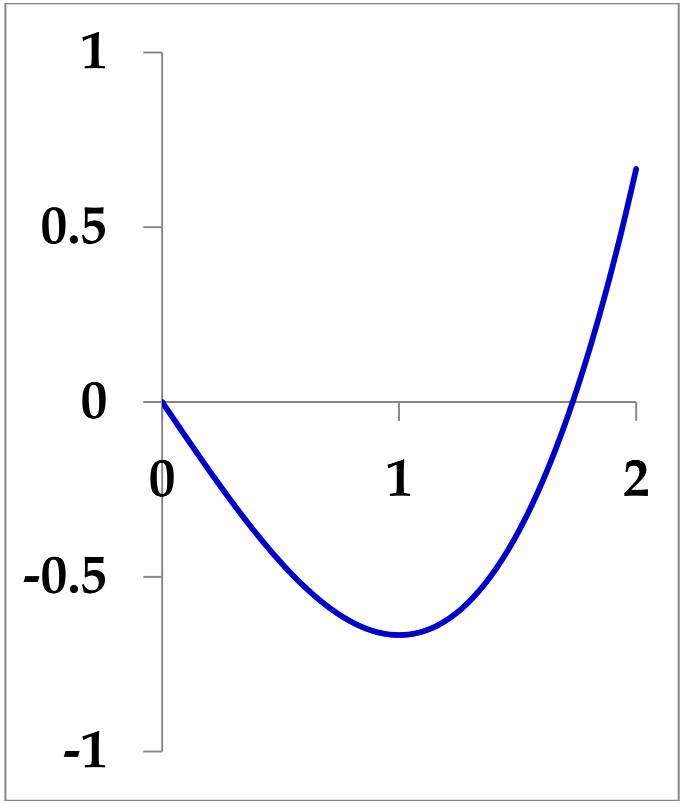
定理 6　设$f(\mathbf{x})$是定义在凸集R上的可微凸函数，若存在点$\mathbf{x}^* \in \mathbf{R}$，使得对于所有$\mathbf{x} \in \mathbf{R}$，有

$$\nabla f(\mathbf{x}^*)^{\mathrm{T}}(\mathbf{x}-\mathbf{x}^*) \geq 0$$

则$\mathbf{x}^*$是$f(\mathbf{x})$在R上的最小点（全局极小点）。

注 4：定理 5 和 6 说明，定义在凸集上的凸函数的平稳点，就是其全局极小点。全局极小点并不一定是唯一的，但若为严格凸函数，则全局极小点就是唯一的。





✧ 凸规划

考虑集合：$\mathbf{R} = \{\mathbf{x} | g_j(\mathbf{x}) \geq 0, j = 1,2, \dots, l\}$。如果$g_j(\mathbf{x})$为凹函数，或者$-g_j(\mathbf{x})$为凸函数，那么根据前述性质 **3**，集合$\mathbf{R} = \{\mathbf{x} | g_j(\mathbf{x}) \geq 0, j = 1,2, \dots, l\}$是凸集。

凸规划就是在凸集上最小化一个凸函数，即

$$\min_{\mathbf{x}\in\mathbf{R}} f(\mathbf{x})$$
$$\text{s.t. } \mathbf{R} = \{\mathbf{x} | g_j(\mathbf{x}) \geq 0, j = 1,2, \dots, l\}$$

若$f(\mathbf{x})$为凸函数，$g_j(\mathbf{x}), (j = 1,2, \dots, l)$为凹函数（或$-g_j(\mathbf{x})$为凸函数），这样的非线性规划称为凸规划。

注 5：凸规划可行域为凸集，局部最优解即为全局最优解；若有多个最优解，则最优解的集合仍然形成一个凸集。

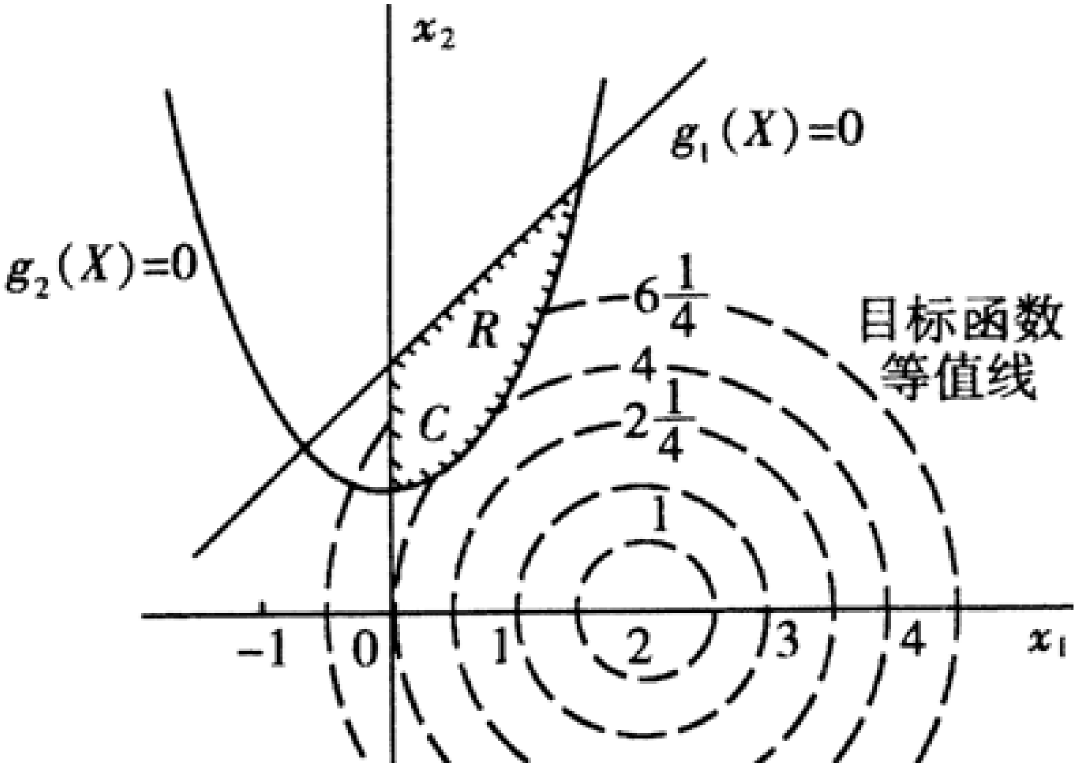
当凸规划的目标函数为严格凸函数时，若最优解存在，则最优解必定唯一。

由于线性函数既可视为凸函数，又可视为凹函数，故线性规划也属于一种凸规划。

例 1，非线性规划问题

$$\min f(\mathbf{x}) = x_1^2 + x_2^2 - 4x_1 + 4$$

$$\begin{cases} g_1(\mathbf{x}) = \quad x_1 \ - x_2 + 2 \geq 0 \\ g_2(\mathbf{x}) = -x_1^2 + x_2 - 1 \geq 0 \\ x_1, x_2 \geq 0 \end{cases}$$



目标函数$f(\mathbf{x})$严格凸，$g_2(\mathbf{x})$凹，其它约束均为线性函数，所以是凸规划问题。

图解可知，C 为最优解：

$$\mathbf{x}^* = (0.58, 1.34)^{\mathrm{T}}$$

最优值：

$$f(\mathbf{x}^*) = 3.8$$

四、下降迭代算法

对一般的无约束n元函数$f(\mathbf{x})$，由梯度条件$\nabla f(\mathbf{x})=\mathbf{0}$求最优解，得到的是一个非线性方程组，求解可能相当困难。若函数不可微，则更无法使用这种方法。

迭代法：先给定初始估计$\mathbf{x}^{(0)}$，然后按某种规则（称为算法）找出比$\mathbf{x}^{(0)}$更好的解$\mathbf{x}^{(1)}$（对极小化问题，即要求满足$f(\mathbf{x}^{(1)})<f(\mathbf{x}^{(0)})$），再按此算法找出比$\mathbf{x}^{(1)}$更好的解$\mathbf{x}^{(2)},\ldots,$如此得到一个解序列$\{\mathbf{x}^{(k)}\}$。

若该解序列存在极限$\mathbf{x}^*$，即$\lim_{k\to\infty}\left\|\mathbf{x}^{(k)}-\mathbf{x}^*\right\|=0$，则称它收敛于$\mathbf{x}^*$。

所产生的解序列$\{\mathbf{x}^{(k)}\}$使目标函数值逐步减少的迭代算法称为下降迭代算法。

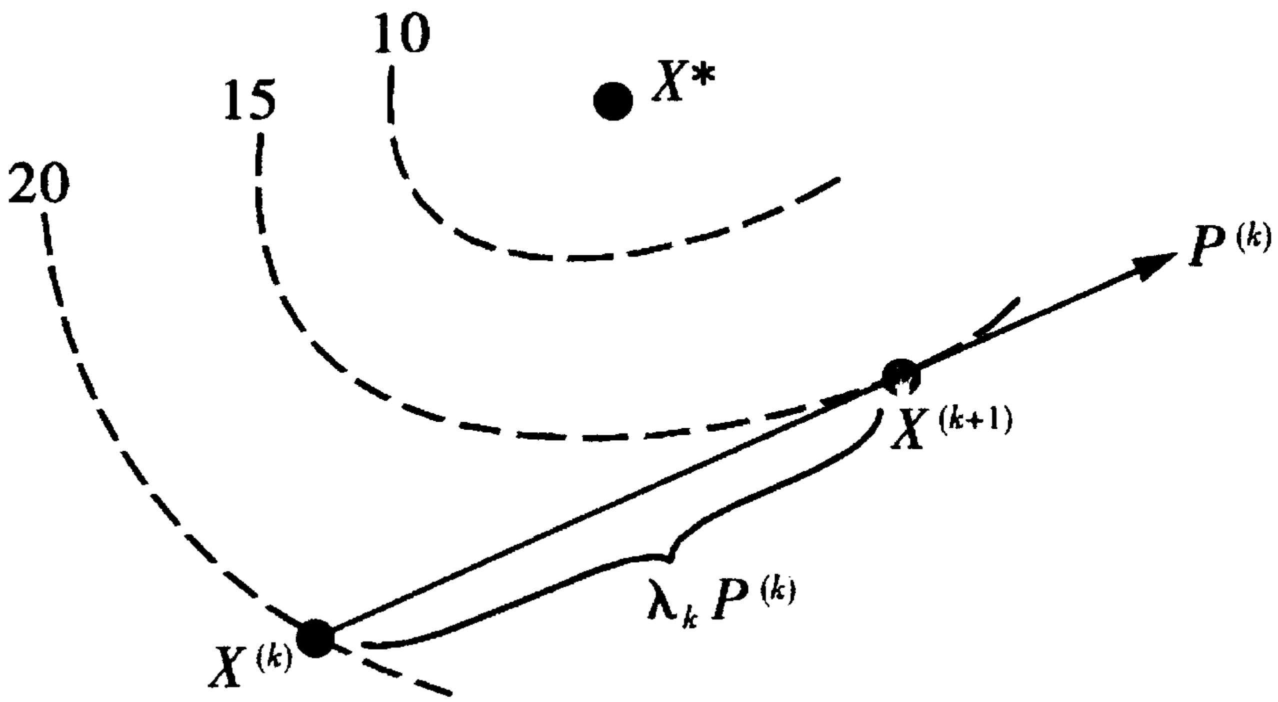
假定已迭代到点$\mathbf{x}^{(k)}$，若从$\mathbf{x}^{(k)}$出发沿任何方向移动都不能使目标值下降，则$\mathbf{x}^{(k)}$就是局部极小，迭代停止。

若至少存在一个方向$\mathbf{p}^{(k)}$使目标函数值有所下降，则可选定该下降方向$\mathbf{p}^{(k)}$，沿该方向适当前进一步，使$f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$，就得到下一迭代点$\mathbf{x}^{(k+1)}$。

这相当于在射线$\mathbf{x}=\mathbf{x}^{(k)}+\lambda\mathbf{p}^{(k)}$上选定新点

$$\mathbf{x}^{(k+1)}=\mathbf{x}^{(k)}+\lambda_k\mathbf{p}^{(k)}$$

$\mathbf{p}^{(k)}$称为搜索方向，λ_k称为步长或步长因子。



✧ 下降迭代算法

（1）选定某一初始点$\mathbf{x}^{(0)}$，令$k=0$；

（2）假设迭代到第k步，则确定一个搜索方向$\mathbf{p}^{(k)}$；

（3）从$\mathbf{x}^{(k)}$出发沿$\mathbf{p}^{(k)}$求步长λ_k，产生下一个迭代点$\mathbf{x}^{(k+1)}$；

（4）检查新点$\mathbf{x}^{(k+1)}$是否为极小点或近似极小。若是，则停止迭代。否则，令$k=k+1$，转（2）继续迭代。

注 6：选取搜索方向$\mathbf{p}^{(k)}$是关键步骤，各种下降迭代算法的区分，主要就在于确定搜索方向的方法不同。

✧ 步长的确定方法

（1）最简单方法：令它等于某一常数（如$\lambda_k=1$），但这不能保证目标函数值一定下降。

（2）可接受点法：任选使目标值下降的步长λ_k。

（3）一维搜索法：沿搜索方向使目标值下降最多，即沿射线$\mathbf{x}=\mathbf{x}^{(k)}+\lambda\mathbf{p}^{(k)}$，求目标函数的极小值：

$$\lambda_k: \min_{\lambda} f(\mathbf{x}^{(k)}+\lambda\mathbf{p}^{(k)})$$

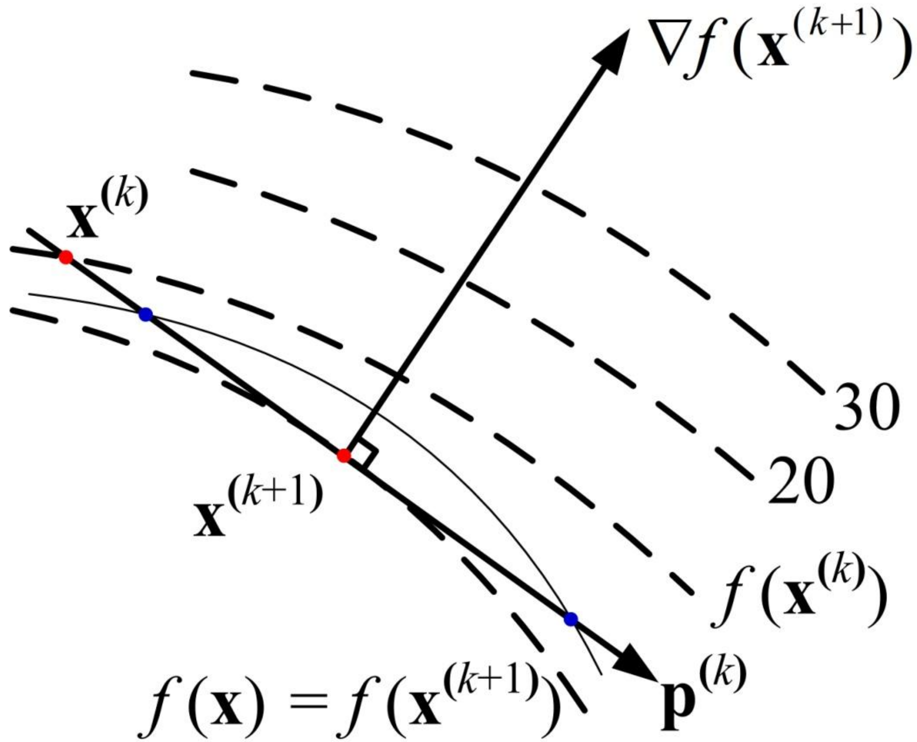
求关于λ的一元函数$f(\mathbf{x}^{(k)}+\lambda\mathbf{p}^{(k)})$的极小点$\lambda_k$，称为一维搜索或线搜索，所确定的步长$\lambda_k$称为最佳步长。

✧ 一维搜索的重要性质

定理 7 设函数 $f(\mathbf{x})$ 具有一阶连续偏导，迭代点 $\mathbf{x}^{(k+1)}$ 按下述规则产生

$$\begin{cases}\lambda_k : \min\limits_{\lambda} f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}) \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}\end{cases}$$

则有：$\nabla f(\mathbf{x}^{(k+1)})^{\mathrm{T}} \mathbf{p}^{(k)} = 0$。



定理 7 表明，在搜索方向上得到的最优点处的梯度和该搜索方向正交。

✧ 迭代终止准则（收敛准则）

真正的最优解事先并不知道，何时停止迭代，可根据相继两次迭代的结果判断。常用的终止准则有：

（1）相继两次迭代的绝对误差足够小

$$\left\|\mathbf{x}^{(k+1)}-\mathbf{x}^{(k)}\right\|<\varepsilon_1 \text{ or } \left|f\left(\mathbf{x}^{(k+1)}\right)-f\left(\mathbf{x}^{(k)}\right)\right|<\varepsilon_2$$

（2）相继两次迭代的相对误差足够小

$$\frac{\left\|\mathbf{x}^{(k+1)}-\mathbf{x}^{(k)}\right\|}{\left\|\mathbf{x}^{(k)}\right\|}<\varepsilon_3 \text{ or } \frac{\left|f\left(\mathbf{x}^{(k+1)}\right)-f\left(\mathbf{x}^{(k)}\right)\right|}{\left|f\left(\mathbf{x}^{(k)}\right)\right|}<\varepsilon_4$$

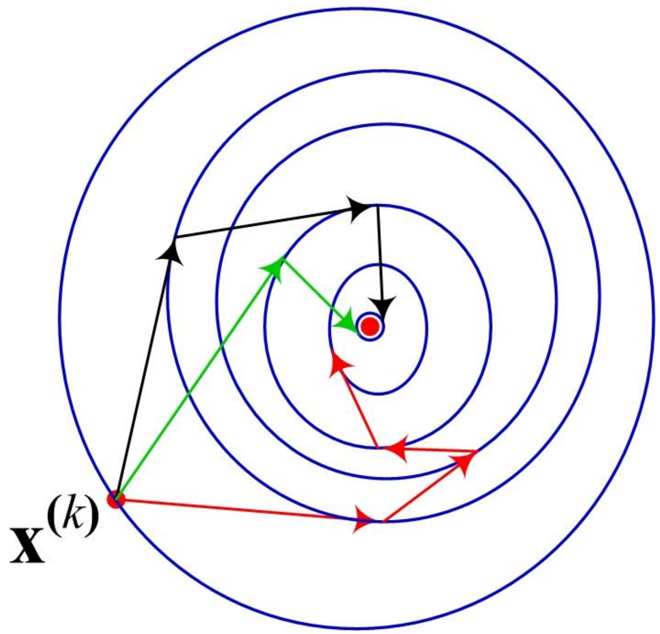
（3）目标函数梯度的模足够小：$\left\|\nabla f\left(\mathbf{x}^{(k)}\right)\right\|<\varepsilon_5$。

第二节　最速下降法

✧ 基本原理

考虑无约束优化问题

$$\min f(\mathbf{x}), \mathbf{x} \in \mathbb{E}^n$$



下降迭代算法：从某一点出发，选择一个使目标函数值下降的方向前进，就可以优化目标函数。

下降方向有很多，选择哪一个？

1847 年，法国数学家 Cauchy（柯西）提出使用目标函数的负梯度作为下降方向：

$$\mathbf{p} = -\nabla f(\mathbf{x})$$

负梯度方向是函数在当前点下降最快的方向，因此基于负梯度的下降迭代算法称为最速下降法。

Augustin-Louis Cauchy

1879-1857，French

✧ 最速下降法的迭代公式

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)} = \mathbf{x}^{(k)} - \lambda_k \nabla f(\mathbf{x}^{(k)})$$

λ_k为迭代步长，满足：

$$f(\mathbf{x}^{(k)} - \lambda_k \nabla f(\mathbf{x}^{(k)})) = \min_{\lambda \geq 0} f(\mathbf{x}^{(k)} - \lambda \nabla f(\mathbf{x}^{(k)}))$$

这是函数$f(\mathbf{x}^{(k)} - \lambda \nabla f(\mathbf{x}^{(k)}))$关于单变量$\lambda$的优化问题。

若函数关于λ凸，且一阶条件可解，则容易获得最优λ从而实现最优下降，否则就要使用数值型一维搜索（线搜索）来求λ的近似最优值。

✧ 最速下降法计算步骤

（1）给定初点$\mathbf{x}^{(0)} \in \mathbb{E}^n$，允许误差$\varepsilon > 0$，置$k = 0$。

（2）设迭代到第k步，令搜索方向$\mathbf{p}^{(k)} = -\nabla f(\mathbf{x}^{(k)})$。

（3）若$\|\mathbf{p}^{(k)}\| \le \varepsilon$，则停止计算；否则，从$\mathbf{x}^{(k)}$出发，沿$\mathbf{p}^{(k)}$进行一维搜索，求$\lambda_k$，使

$$f(\mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}) = \min_{\lambda \ge 0} f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)})$$

（4）令$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}$，置$k := k + 1$，转（2）。

例 3，求$f(\mathbf{x}) = 2x_1^2 + x_2^2$的最小点。其中初始点：$\mathbf{x}^{(0)} = (1,1)^{\mathrm{T}}$，精度要求$\|\nabla f(\mathbf{x})\| \le \varepsilon = 1/10$。

第 1 次迭代：目标函数梯度$\nabla f(\mathbf{x}) = \begin{bmatrix} 4x_1 \\ 2x_2 \end{bmatrix}$，令搜索方向

$\mathbf{p}^{(0)} = -\nabla f(\mathbf{x}^{(0)}) = \begin{bmatrix} -4 \\ -2 \end{bmatrix}$，有

$$\|\mathbf{p}^{(0)}\| = \sqrt{14+4} = 2\sqrt{5} > \varepsilon$$

不满足精度要求。

从$\mathbf{x}^{(0)}$出发，令$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \lambda \mathbf{p}^{(0)} = \begin{bmatrix} 1-4\lambda \\ 1-2\lambda \end{bmatrix}$，沿方向$\mathbf{p}^{(0)}$进行一维搜索，求最优步长$\lambda_0$，即

$$\min_{\lambda \geq 0} \phi(\lambda) = f(\mathbf{x}^{(0)} + \lambda \mathbf{p}^{(0)}) = 2(1-4\lambda)^2 + (1-2\lambda)^2$$

$\phi(\lambda)$关于λ凸，运用一阶条件，令$\phi'(\lambda) = 0$，解得：

$$\lambda_0 = 5/18$$

因此沿搜索方向$\mathbf{p}^{(0)}$，得到极小点为：

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \lambda_0 \mathbf{p}^{(0)} = \begin{bmatrix} -1/9 \\ 4/9 \end{bmatrix}$$

$$\mathbf{p}^{(1)} = -\nabla f(\mathbf{x}^{(1)}) = \begin{bmatrix} 4/9 \\ -8/9 \end{bmatrix}, \ \|\mathbf{p}^{(1)}\| = 4\sqrt{5}/9 \approx 0.99 > \varepsilon$$

不满足精度要求，需要进一步迭代。

第 2 次迭代：

$$\min_{\lambda\geq 0}\phi(\lambda)=f(\mathbf{x}^{(1)}+\lambda\mathbf{p}^{(1)})=2(-\frac{1}{9}+\frac{4}{9}\lambda)^2+(\frac{4}{9}-\frac{8}{9}\lambda)^2$$

解得：

$$\lambda_1=5/12,\ \mathbf{x}^{(2)}=\mathbf{x}^{(1)}+\lambda_1\mathbf{p}^{(1)}=\begin{bmatrix}2/27\\2/27\end{bmatrix}$$

$$\mathbf{p}^{(2)}=-\nabla f(\mathbf{x}^{(2)})=\begin{bmatrix}-8/27\\-4/27\end{bmatrix}$$

$$\left\|\mathbf{p}^{(2)}\right\|=4\sqrt{5}/27\approx 0.33>\varepsilon$$

仍然不满足精度要求。

第 3 次迭代：

$$\min_{\lambda\geq 0}\phi(\lambda)=f(\mathbf{x}^{(2)}+\lambda\mathbf{p}^{(2)})=\frac{8}{27^2}(1-4\lambda)^2+\frac{4}{27^2}(1-2\lambda)^2$$

解得：

$$\lambda_2=5/18,\ \ \mathbf{x}^{(3)}=\mathbf{x}^{(2)}+\lambda_2\mathbf{p}^{(2)}=\frac{2}{243}\begin{bmatrix}-1\\4\end{bmatrix}$$

$$\nabla f(\mathbf{x}^{(3)})=\frac{2}{243}\begin{bmatrix}-4\\8\end{bmatrix}$$

此时有，$\|\nabla f(\mathbf{x}^{(3)})\|=\frac{8}{243}\sqrt{5}\approx 0.07<\varepsilon$，满足精度要求，得到近似最优解：

$$\mathbf{x}^*=\frac{2}{243}\begin{bmatrix}-1\\4\end{bmatrix}。$$

✧ 单变量问题的优化

下降迭代算法每一步都需要求解一个单变量问题，若难以用微分方法求最优步长λ_k，则需考虑近似方法。

方法 1：近似最佳步长

若$f(\mathbf{x})$具有二阶连续偏导，则可以在点$\mathbf{x}^{(k)}$处用二阶泰勒展开近似函数$f(\mathbf{x}^{(k)}+\lambda\mathbf{p}^{(k)})$：

$$f(\mathbf{x}^{(k)}+\lambda\mathbf{p}^{(k)})=f(\mathbf{x}^{(k)}-\lambda\nabla f(\mathbf{x}^{(k)}))\approx$$

$$f(\mathbf{x}^{(k)})-\lambda\nabla f(\mathbf{x}^{(k)})^{\mathrm{T}}\nabla f(\mathbf{x}^{(k)})+\frac{1}{2}\lambda^2\nabla f(\mathbf{x}^{(k)})^{\mathrm{T}}\mathbf{H}(\mathbf{x}^{(k)})\nabla f(\mathbf{x}^{(k)})$$

其中$\mathbf{H}(\mathbf{x}^{(k)})=\nabla^2 f(\mathbf{x}^{(k)})$是函数在$\mathbf{x}^{(k)}$处的海塞矩阵。

对λ求导并令导数等于零，可得近似最佳步长：

$$\lambda_k = \frac{\nabla f(\mathbf{x}^{(k)})^{\mathrm{T}} \nabla f(\mathbf{x}^{(k)})}{\nabla f(\mathbf{x}^{(k)})^{\mathrm{T}} \mathbf{H}(\mathbf{x}^{(k)}) \nabla f(\mathbf{x}^{(k)})} \qquad (11)$$

下一迭代点：$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}$

有时将搜索方向$\mathbf{p}^{(k)}$的模规格化为 1，此时

$$\mathbf{p}^{(k)} = -\frac{\nabla f(\mathbf{x}^{(k)})}{\left\|\nabla f(\mathbf{x}^{(k)})\right\|} \qquad (12)$$

那么，式（11）相应变为

$$\lambda_k = \frac{\nabla f(\mathbf{x}^{(k)})^{\mathrm{T}} \nabla f(\mathbf{x}^{(k)}) \left\|\nabla f(\mathbf{x}^{(k)})\right\|}{\nabla f(\mathbf{x}^{(k)})^{\mathrm{T}} \mathbf{H}(\mathbf{x}^{(k)}) \nabla f(\mathbf{x}^{(k)})} \qquad (13)$$

例 4，从$\mathbf{x}^{(0)}=(0.5,0.8)^{\mathrm{T}}$出发，使用近似最佳步长，以最速下降法求$f(\mathbf{x})=4x_1^2+4x_2^2+e^{x_1+x_2}$的极小点。

解，梯度和 Hesse 矩阵为：

$$\nabla f(\mathbf{x})=\begin{bmatrix}8x_1+e^{x_1+x_2}\\8x_2+e^{x_1+x_2}\end{bmatrix},\quad \mathbf{H}(\mathbf{x})=\begin{bmatrix}8+e^{x_1+x_2} & e^{x_1+x_2}\\ e^{x_1+x_2} & 8+e^{x_1+x_2}\end{bmatrix}$$

使用规格化搜索方向：$\mathbf{p}^{(k)}=-\nabla f(\mathbf{x}^{(k)})/\|\nabla f(\mathbf{x}^{(k)})\|$。

在初始点$\mathbf{x}^{(0)}=(0.5,0.8)^{\mathrm{T}}$，梯度和 Hesse 矩阵为：

$$\nabla f(\mathbf{x}^{(0)})=(7.6693,10.0693)^{\mathrm{T}},\quad \mathbf{H}(\mathbf{x}^{(0)})=\begin{bmatrix}11.67 & 3.67\\3.67 & 11.67\end{bmatrix}$$

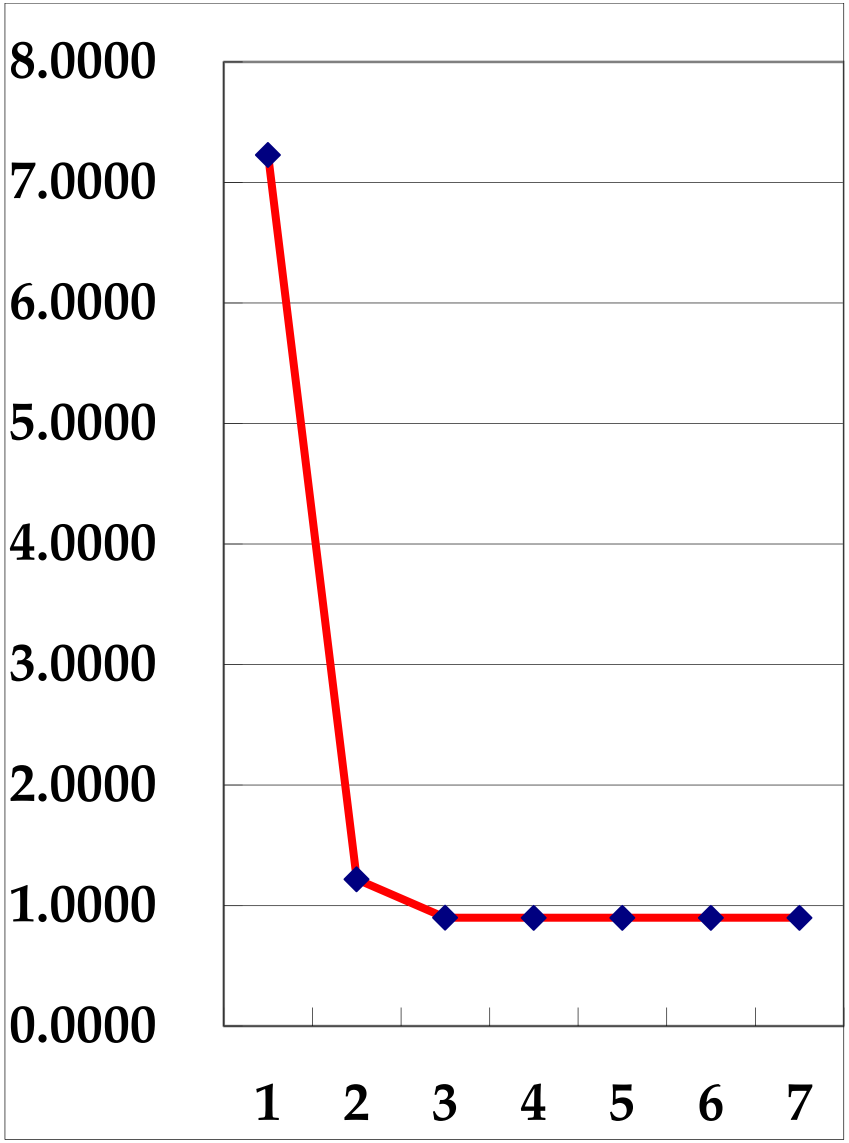
步长：
$$\lambda_0=\frac{\nabla f(\mathbf{x}^{(0)})^{\mathrm{T}}\nabla f(\mathbf{x}^{(0)})\|\nabla f(\mathbf{x}^{(0)})\|}{\nabla f(\mathbf{x}^{(0)})^{\mathrm{T}}\mathbf{H}(\mathbf{x}^{(0)})\nabla f(\mathbf{x}^{(0)})}=0.8324$$

搜索方向：$\mathbf{p}^{(0)} = -\frac{\nabla f(\mathbf{x}^{(0)})}{\|\nabla f(\mathbf{x}^{(0)})\|} = -(0.6059,0.7955)^{\mathrm{T}}$

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \lambda_0 \mathbf{p}^{(0)} = (-0.0043,0.1378)^{\mathrm{T}}$$

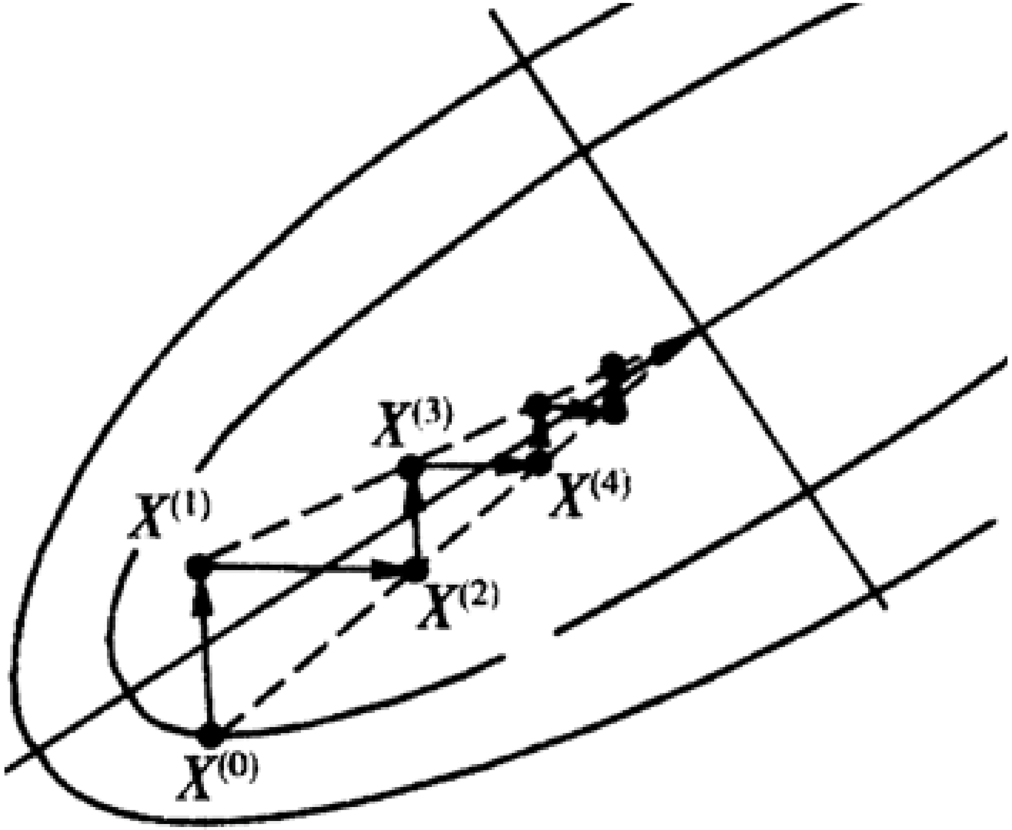
在$\mathbf{x}^{(1)}$点计算梯度、**Hesse** 矩阵和步长，迭代如下：

迭代点	$\nabla f(\mathbf{x}^{(k)})$	$\mathbf{H}(\mathbf{x}^{(k)})$	λ_k	$f(\mathbf{x}^{(k)})$
$\mathbf{x}^{(0)} = (0.5000,0.8000)^{\mathrm{T}}$	$(7.67,10.07)^{\mathrm{T}}$	$\begin{bmatrix} 11.67 & 3.67 \\ 3.67 & 11.67 \end{bmatrix}$	0.8324	7.2293
$\mathbf{x}^{(1)} = (-0.0043,0.1378)^{\mathrm{T}}$	$(1.11,2.25)^{\mathrm{T}}$	$\begin{bmatrix} 9.14 & 1.14 \\ 1.14 & 9.147 \end{bmatrix}$	0.2492	1.2189
$\mathbf{x}^{(2)} = (-0.1146,-0.0856)^{\mathrm{T}}$	$(-0.10,0.13)^{\mathrm{T}}$	$\begin{bmatrix} 8.82 & 0.82 \\ 0.82 & 8.82 \end{bmatrix}$	0.0206	0.9004
$\mathbf{x}^{(3)} = (-0.1024,-0.1022)^{\mathrm{T}}$	$(0.00,0.00)^{\mathrm{T}}$	$\begin{bmatrix} 8.81 & 0.81 \\ 0.81 & 8.81 \end{bmatrix}$	0.0005	0.8987
$\mathbf{x}^{(4)} = (-0.1020,-0.1019)^{\mathrm{T}}$	$(0.00,0.00)^{\mathrm{T}}$	$\begin{bmatrix} 8.82 & 0.82 \\ 0.82 & 8.82 \end{bmatrix}$	≈ 0	0.8987
$\mathbf{x}^{(5)} = (-0.1019,-0.1019)^{\mathrm{T}}$	$(0.00,0.00)^{\mathrm{T}}$	$\begin{bmatrix} 8.82 & 0.82 \\ 0.82 & 8.82 \end{bmatrix}$	≈ 0	0.8987
$\mathbf{x}^{(6)} = (-0.1019,-0.1019)^{\mathrm{T}}$	$(0.00,0.00)^{\mathrm{T}}$	$\begin{bmatrix} 8.82 & 0.82 \\ 0.82 & 8.82 \end{bmatrix}$	≈ 0	0.8987



$$f(\mathbf{x}) = 4x_1^2 + 4x_2^2 + e^{x_1+x_2}$$

注 8：负梯度方向，仅在当前点附近才具有“最速下降”性，



对于整个极小化过程来说，负梯度并非收敛最快的方向。

从全局看，精确一维搜索得到的前后两次迭代方向相互垂直，因而搜索路线总体呈锯齿状，当趋近极小点时，即使向极小点移动很小距离，由于锯齿现象，也要经历不小的弯路，收敛速度大为减慢。[3]

[3] 如例 3：$\mathbf{p}^{(0)}=\begin{bmatrix}-4\\-2\end{bmatrix}$; $\mathbf{p}^{(1)}=\begin{bmatrix}4/9\\-8/9\end{bmatrix}$; $\mathbf{p}^{(2)}=\begin{bmatrix}-8/27\\-4/27\end{bmatrix}$; $\mathbf{p}^{(3)}=-\nabla f(\mathbf{x}^{(3)})=\frac{2}{243}\begin{bmatrix}4\\-8\end{bmatrix}$，相邻点积均为 0——垂直。

在实用中，常将最速下降法和其它方法联合起来应用，在前期使用最速下降法，而在接近极小点时，则使用收敛较快的其它方法。

无约束优化更好的求解方法有：

（1）牛顿法

（2）共轭梯度法

（3）拟牛顿法（变尺度法）

方法 2：数值型一维搜索

求近似最佳步长，除了需要使用梯度，还需要使用 **Hesse** 矩阵，对高阶问题，计算起来比较麻烦；

而函数的二次近似也有可能因为步长不精确而导致不收敛，结果在最优点附近震荡；

在实际计算中，可使用数值型一维搜索技术（**0.618** 法、斐波那契法等）确定迭代步长。

第三节　一维搜索

所谓一维搜索，就是求解单变量（一维）函数最优值的一类数值方法，在下降迭代算法[4]中经常碰到。

考虑以最速下降法求解：

$$\min f(\mathbf{x}) = 4x_1^2 + 4x_2^2 + e^{x_1+x_2}$$

假设迭代到点$\mathbf{x}^{(k)} = (0.5,0.8)^{\mathrm{T}}$，那么规格化下降方向为：

$$\mathbf{p}^{(k)} = -\frac{\nabla f(\mathbf{x}^{(k)})}{\|\nabla f(\mathbf{x}^{(k)})\|} = -(0.6059,0.7955)^{\mathrm{T}}$$

下一步迭代点为：

[4] 下降迭代算法需不断求解关于步长的单变量问题：$\min_{\lambda\geq 0} f(\mathbf{x}^{(k+1)}) = f(\mathbf{x}^{(k)} + \lambda\mathbf{p}^{(k)})$。

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)} = (0.5 - 0.6059\lambda, 0.8 - 0.7955\lambda)^{\mathrm{T}}$$

于是需要求解如下问题：

$$\min \phi(\lambda) = 4(0.5 - 0.6059\lambda)^2 + 4(0.8 - 0.7955\lambda)^2 + e^{1.3 - 1.4014\lambda}$$

使用一阶条件求解，并不容易。

若在点$\mathbf{x}^{(k)}$对原函数做二阶近似，则可得到近似最佳步长——但要使用 **Hesse** 矩阵，高阶问题计算较麻烦。

若原问题性质较好，则所碰到的上述单变量问题的目标函数，都是所谓单峰函数，可以考虑使用一维搜索方法进行数值求解。

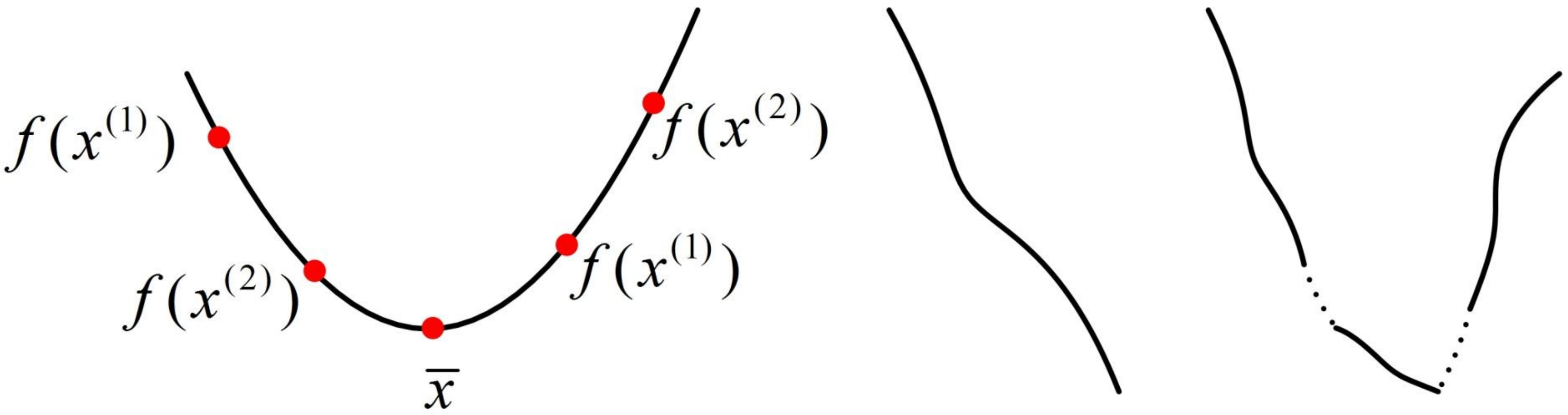
一、单峰函数及其性质

定义 1 设$f(x)$是定义在闭区间$[a,b]$上的一元实函数，$\bar{x}$是$f(x)$在$[a,b]$上的极小点。

若对任意$x^{(1)} < x^{(2)} \in [a,b]$，成立：

$$\begin{cases} f(x^{(1)}) > f(x^{(2)}) & if \;\; x^{(2)} \le \bar{x} \\ f(x^{(1)}) < f(x^{(2)}) & if \;\; \bar{x} \le x^{(1)} \end{cases}$$

则称函数f是闭区间$[a,b]$上的单峰函数。

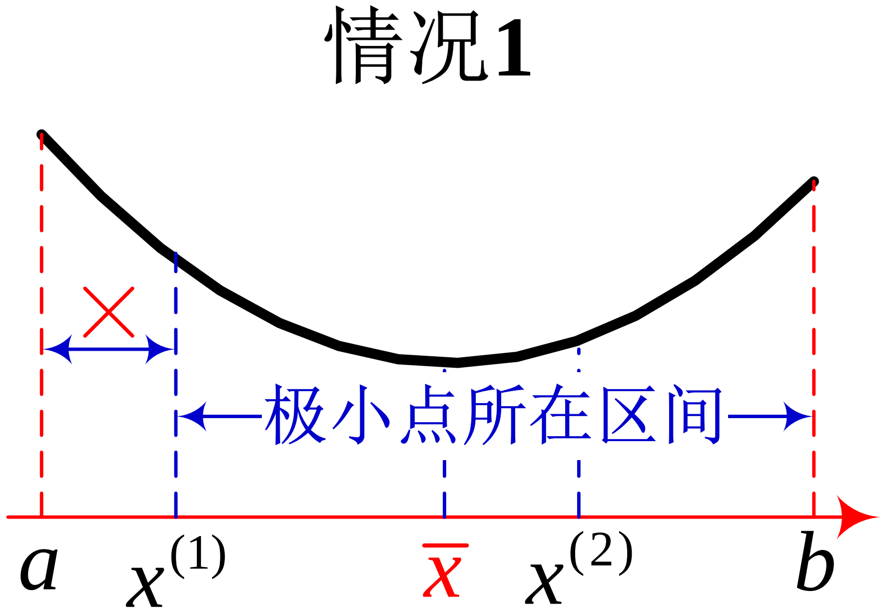


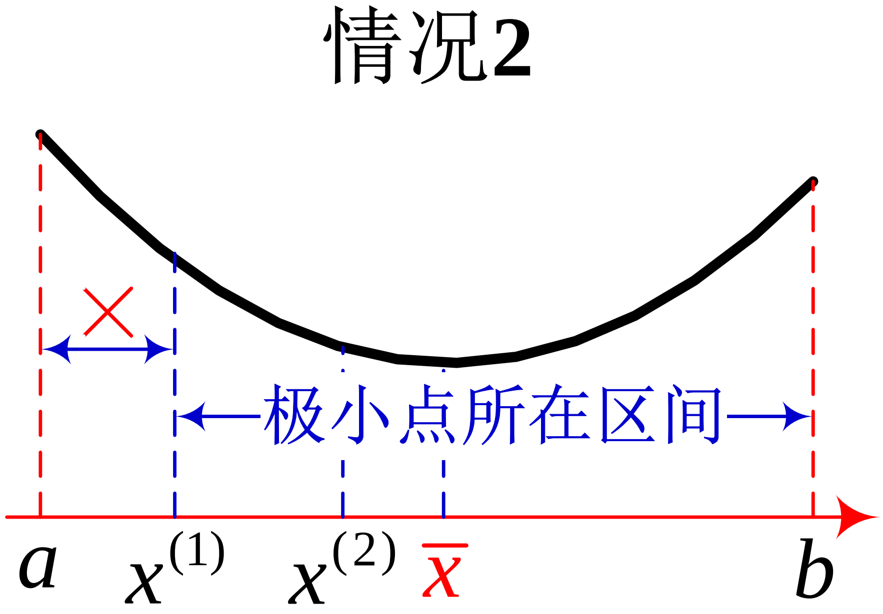
✧ 单峰函数的性质

通过计算区间$[a,b]$内两个不同点处的函数值，就能确定一个包含极小点的子区间。

定理 8 设$f(x)$是区间$[a,b]$上的单峰函数，$x^{(1)},x^{(2)}\in[a,b]$，且$x^{(1)}<x^{(2)}$。如果$f(x^{(1)})>f(x^{(2)})$，则对每一个$x\in[a,x^{(1)}]$，必有$f(x)>f(x^{(2)})$；如果$f(x^{(1)})\le f(x^{(2)})$，则对每一个$x\in[x^{(2)},b]$，必有$f(x)\ge f(x^{(1)})$。

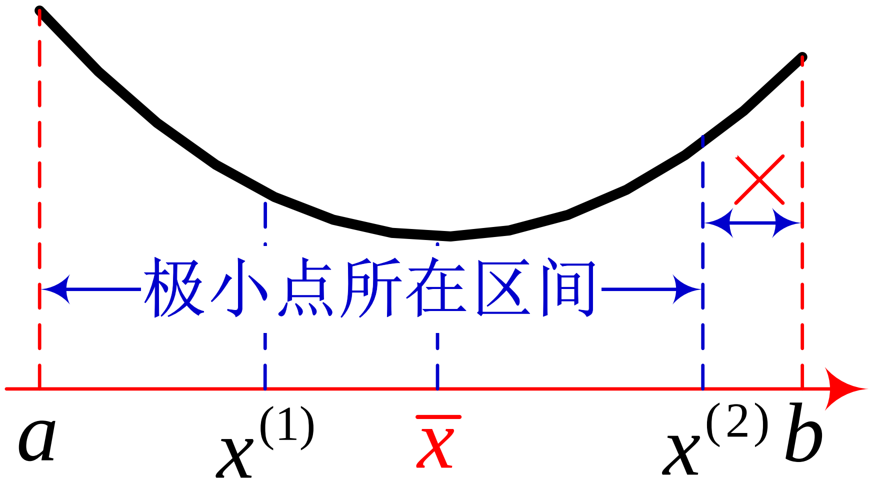
根据定理 **8**，如果$f(x^{(1)}) > f(x^{(2)})$，则意味着最小点不可能在区间$[a, x^{(1)}]$内，那么只能在$[x^{(1)}, b]$内：



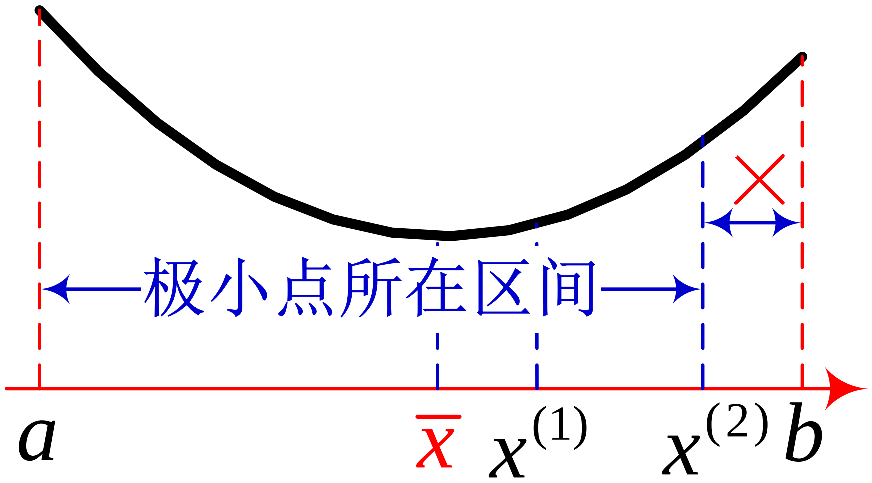


反过来，若$f(x^{(1)}) \le f(x^{(2)})$，则最小点不可能在区间$[x^{(2)},b]$内，只能在$[a,x^{(2)}]$内。

情况1



情况2



所以，只需选择两个试探点，就可将包含极小点的区间缩短为$[x^{(1)},b]$，或$[a,x^{(2)}]$。

二、0.618 法（黄金分割法）

基本思想

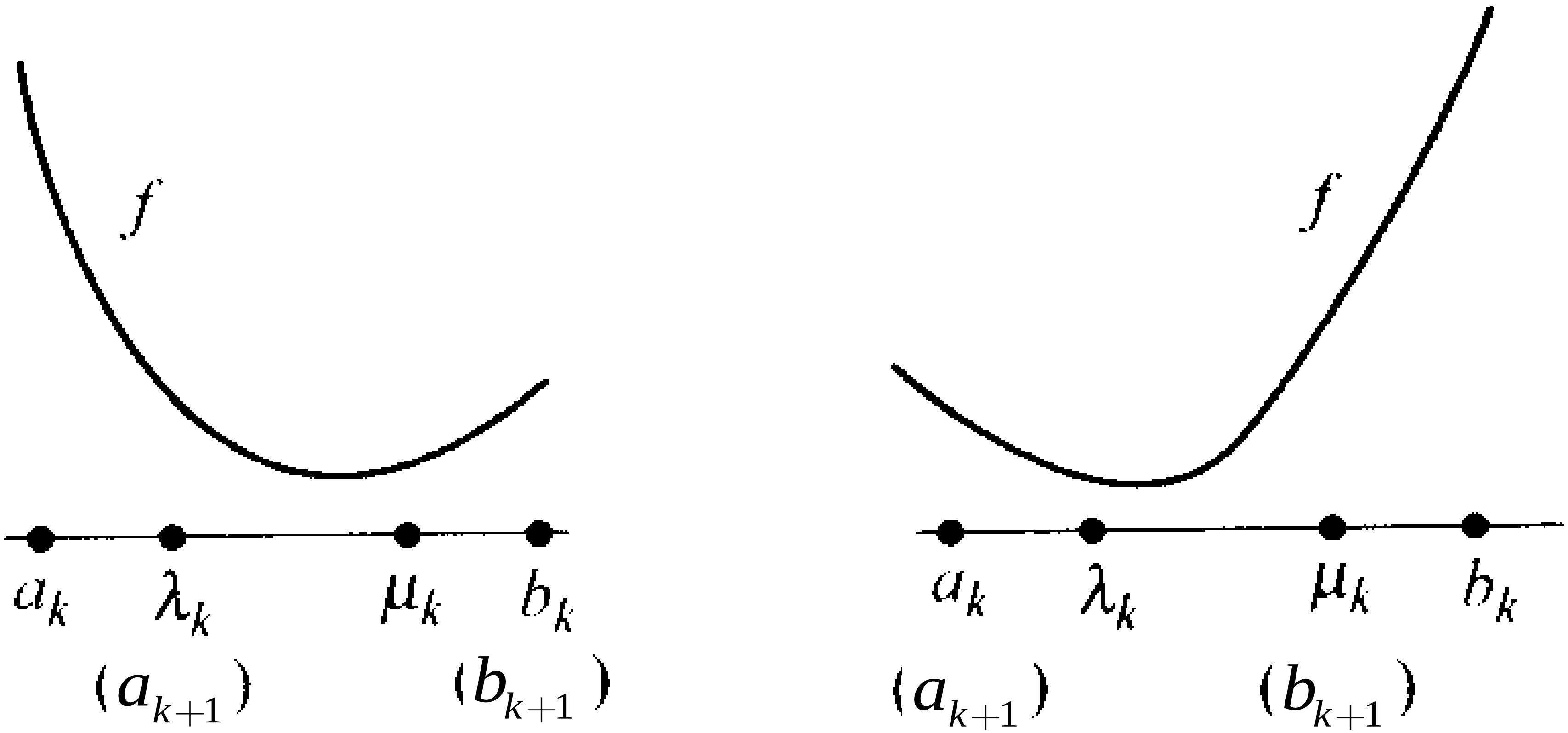
不断取试探点可使包含极小点的区间不断缩短，直到满足精度要求；取试探点的方案很多，**0.618** 法的基本思路是：每次都取黄金分割点作为试探点。

试探点计算公式

设$f(x)$是$[a_1,b_1]$上的单峰函数，极小点$\bar{x}\in[a_1,b_1]$。设进行第k次迭代时，有$\bar{x}\in[a_k,b_k]$。为缩短包含极小点的区间，取两个试探点$\lambda_k<\mu_k\in[a_k,b_k]$，如下：

$$\lambda_k = a_k + 0.382(b_k - a_k) \quad \textbf{(14)}$$

$$\mu_k = a_k + 0.618(b_k - a_k) \quad \textbf{(15)}$$



计算步骤:

（1）置初始区间$[a_1, b_1]$及精度要求$L > 0$，计算试探点λ_1和μ_1，令$k = 1$，计算函数值$f(\lambda_1)$和$f(\mu_1)$，其中

$$\lambda_1 = a_1 + 0.382(b_1 - a_1)，\mu_1 = a_1 + 0.618(b_1 - a_1)$$

（2）若$b_k - a_k < L$，则停止计算。否则，当$f(\lambda_k) > f(\mu_k)$时，转步骤（3）；当$f(\lambda_k) \leq f(\mu_k)$时，转步骤（4）。

（3）置$a_{k+1} = \lambda_k$，$b_{k+1} = b_k$，取新的试探点：

$$\lambda_{k+1} = \mu_k，\mu_{k+1} = a_{k+1} + 0.618(b_{k+1} - a_{k+1})$$

计算函数值$f(\mu_{k+1})$，转步骤（5）。

（4）置$a_{k+1} = a_k$，$b_{k+1} = \mu_k$，取新的试探点：

$$\lambda_{k+1} = a_{k+1} + 0.382(b_{k+1} - a_{k+1}),\ \mu_{k+1} = \lambda_k,$$

计算函数值$f(\lambda_{k+1})$，转步骤（5）。

（5）置$k = k + 1$，返回步骤（2）。

例 2，求解极小化问题

$$\min \phi(\lambda) = 4(0.5 - 0.6059\lambda)^2 + 4(0.8 - 0.7955\lambda)^2 + e^{1.3-1.4014\lambda}$$

设定初始区间$[a_1, b_1] = [0.5,1.5]$，精度$L = 0.05$。

使用 **0.618** 法做一维搜索，则迭代过程如下（见 **PPT**）：

k	a_k	b_k	λ_k	μ_k	$\phi(\lambda_k)$	$\phi(\mu_k)$	$b_k - a_k$
1	0.5000	1.5000	0.8820	1.1180	1.1095	0.9237	1.0000
2	0.8820	1.5000	1.1180	1.2639	0.9237	1.0757	0.6180
3	0.8820	1.2639	1.0279	1.1180	0.9305	0.9237	0.3819
4	1.0279	1.2639	1.1180	1.1738	0.9237	0.9582	0.2360
5	1.0279	1.1738	1.0836	1.1180	0.9171	0.9237	0.1459
6	1.0279	1.1180	1.0623	1.0836	0.9187	0.9171	0.0901
7	1.0623	1.1180	1.0836	1.0967	0.9171	0.9183	0.0557
8	1.0623	1.0967	1.0755	1.0836	0.9172	0.9171	0.0344

经过 7 次迭代有

$$b_8 - a_8 = 0.0344 < L = 0.05$$

近似极小点：$\bar{\lambda} \in [1.06238, 1.0968]$，可取近似最优解为

$$\bar{\lambda} = \frac{a_8 + b_8}{2} = \frac{1.06238 + 1.0968}{2} = 1.08$$

这个值和例 1 第一步的近似最佳步长 **0.8324** 不同。

注，使用 **0.618** 法时，每次在新区间中只需计算一个新点，另一个点复用自上次迭代所用的某个点，即可在新区间中构造出两个点来进一步缩短区间。

如何确定初始区间？

使用试探法，寻找三点，使函数值呈现“高-低-高”的关系，则函数值高的两个端点包含最小点，如下：

从任一点a出发（函数值$f(a)$），前进（或后退）一步δ，得到$f(a+\delta)$，如果$f(a) > f(a+\delta)$，则再前进2δ，不断加倍前进，直到得到某两点a', b'，其关系为：

$$f(a') < f(b')$$

则初始区间确定为$[a_0, b_0] = [a, b']$。

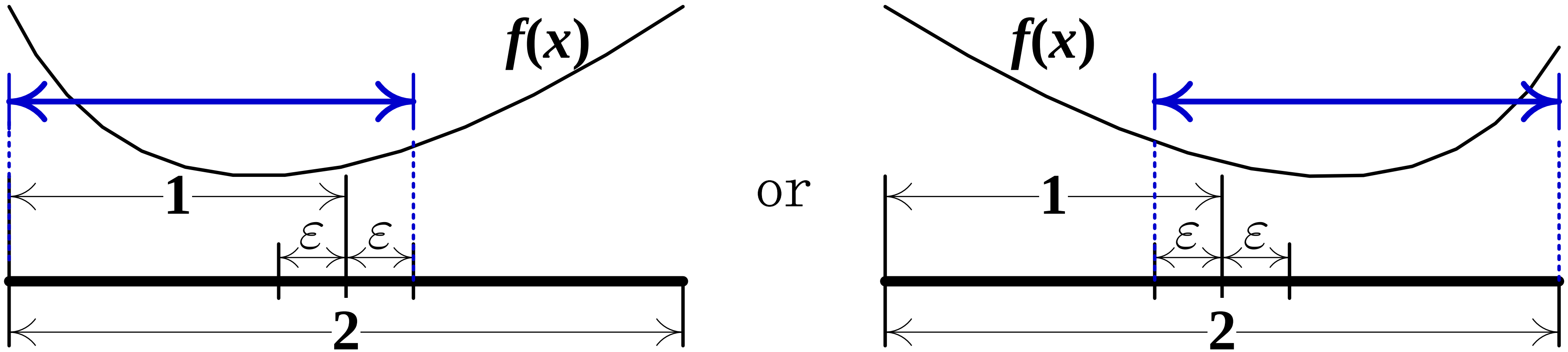
三、斐波那契（Fibonacci）法

问题：基于两点试探原理来缩短区间，则世界上速度最快的一维搜索方法是什么？

这个问题等价于：在给定试探点数的情况下，最长能将多长的区间缩短到长度为 1?

假设只允许搜索 1 次，显然只能将长度为 1 的区间“缩短”到 1：因为不能做 2 点比较而改变区间长度。

如果允许搜索 2 个点呢？如下图：

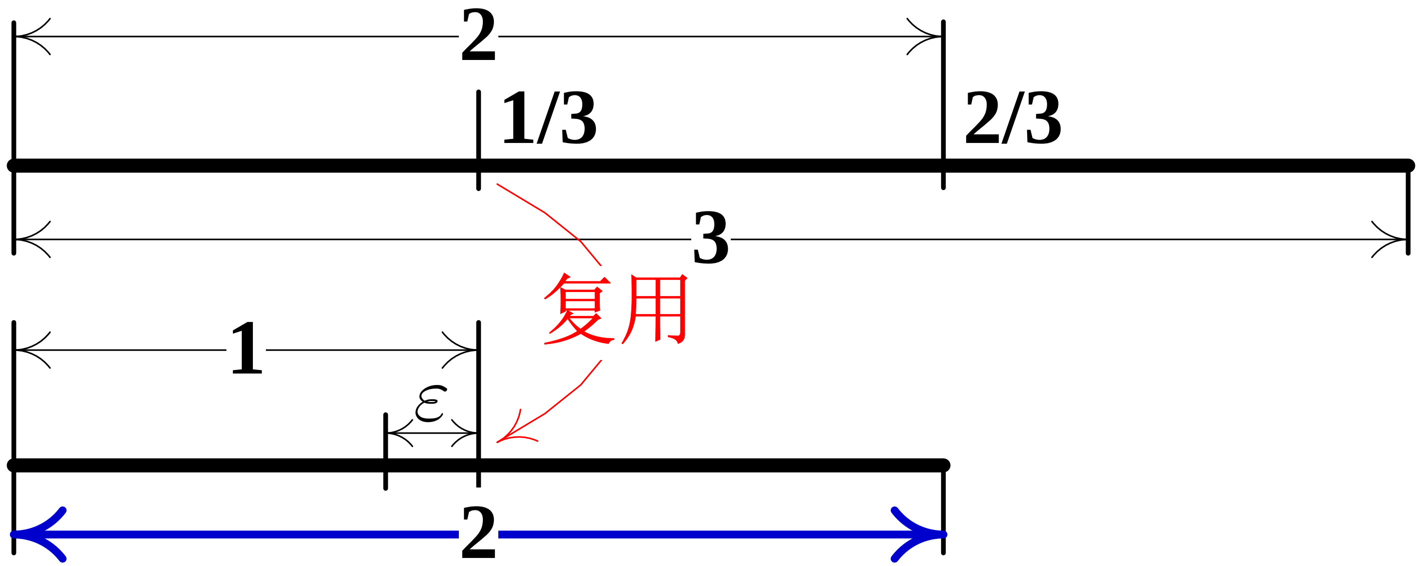


比较两个点的函数值：$f(1-\varepsilon)$和$f(1+\varepsilon)$，可将长度为 2 的区间缩短为$[0,1+\varepsilon]$或$[1-\varepsilon,2]$。当ε很小时，缩短后的区间长度近似为 1。

所以，若只允许选择两个试探点，那么最长可将长度为 2 的区间缩短到长度为 1。

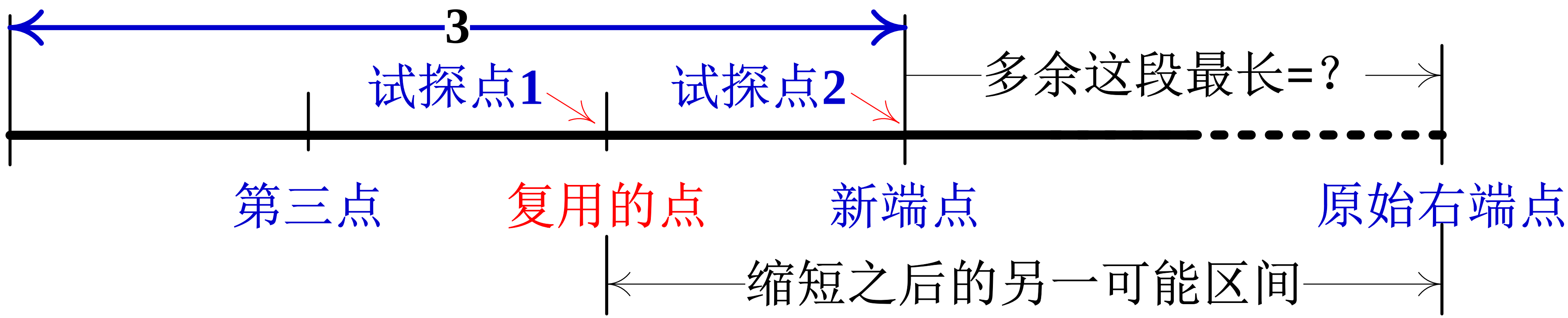
如果允许选 3 个点呢？

先通过 1/3 和 2/3 位置的两点将区间缩短为 2，然后再用第三个点将其缩短到 1——上次的两点之一可复用：



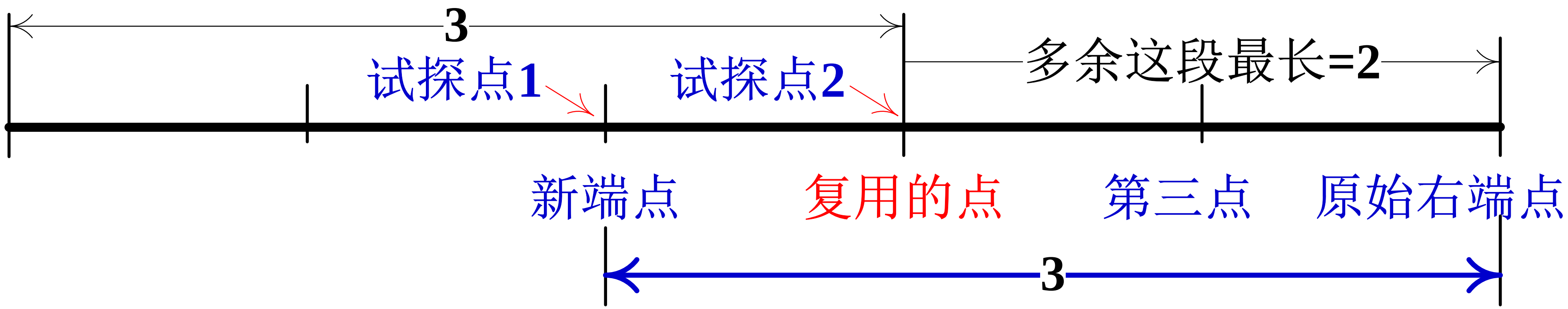
如果允许试探 4 个点呢？

可先用两个点将区间缩短为 3，再用第三点缩短为 2，再用第四点缩短为 1——上次迭代的两个点，一个成为新端点，另一个必须复用而参与下次函数值评估：



复用的点必须处于新区间的 2/3 处，才能和第三点共同将区间长度进一步缩短为 2。

新区间也有可能是下面的情况：

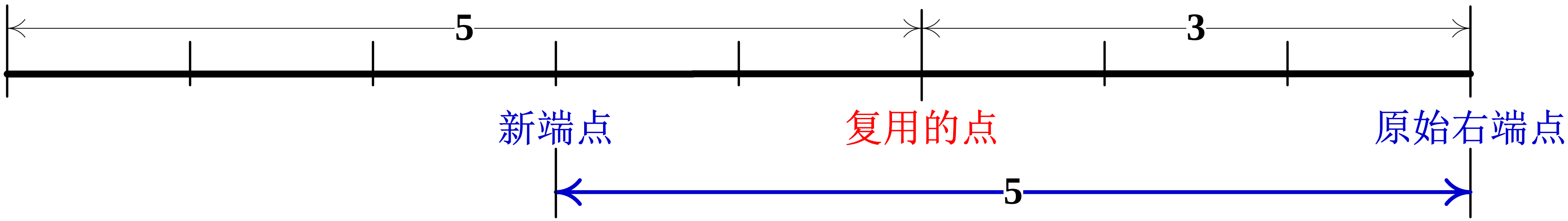


因为缩短前并不知道缩短后的区间取左还是取右，所以必须是两种可能的结果都是 3，才能保证必然能将区间缩短为 3，结果多余的这段最大长度只能为 2。

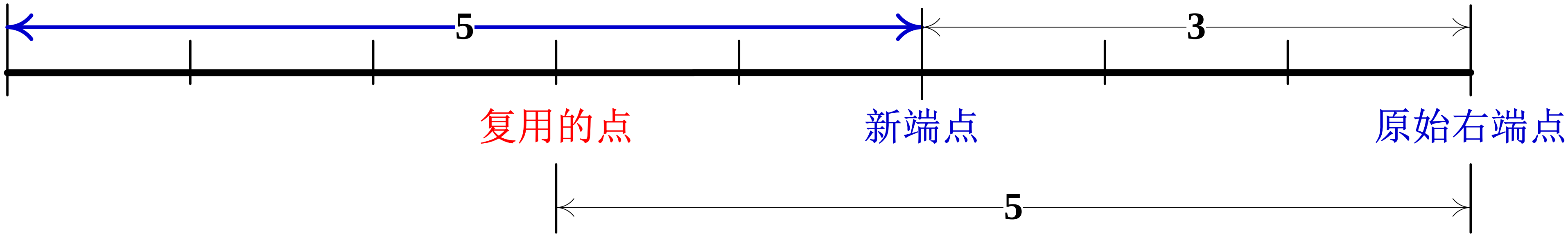
因此试探4个点,可将最长为5的区间经3→2缩短为1,且最初两个点位置分别在 2/5 和 3/5 处。

如果允许使用 5 个试探点呢?

先用两个点将区间缩短为 5，再缩为 3→2→1：



或



所以，5 个点可将最长为 8 的区间缩短到 1，且初始两点位置分别为 3/8 和 5/8。

接着两点位置必须为 2/5 和 3/5（包括一个复用点），然后两点位置是 1/3 和 2/3，最后是区间为 2 的需要缩短为 1，两点位置就是 1/2 和$(1+\varepsilon)/2$。

上述给出的点数和可缩区间最大长度的关系为：

点数k	0	1	2	3	4	5	6	7	8	9	10	…
最长区间F_k	1	1	2	3	5	8	13	21	34	55	89	…

定义 2 上表的数字序列满足：

$$F_0=F_1=1;\ F_{k+1}=F_k+F_{k-1},(k=1,2,\ldots)$$

称为 **Fibonacci 数列**。

可以看到，运用 **Fibonacci** 数列选n个点，就可将区间从F_n缩短到 1，缩短比率为$1/F_n$。

一维搜索：给定初始区间$[a_1, b_1]$，若以 **Fibonacci** 数列选试探点的位置，则最少需要多少个试探点才能将区间缩短为L？也即缩短比率为$L/(b_1 - a_1)$。

因为 **Fibonacci** 数列选点法的缩短比率为$1/F_n$，于是应有$1/F_n \leq L/(b_1 - a_1)$。即，试探点数n需满足：

$$F_n \geq \frac{b_1 - a_1}{L}$$

例，区间长度为3，要求缩短到0.05，求试探点数以及各试探点的位置。

解，需要$F_n \geq \frac{b_1 - a_1}{L} = 60$。查 **Fibonacci** 数列表，应该取$F_{10} = 89$，那么试探点数为10，位置分别为：

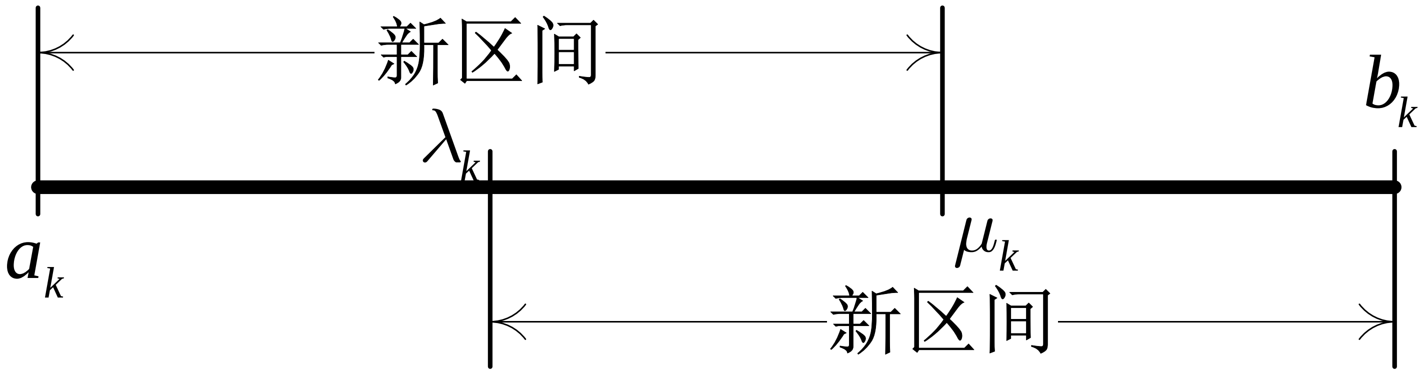
第一次迭代：34/89和55/89	第六次迭代：3/8和5/8
第二次迭代：21/55和34/55	第七次迭代：2/5和3/5
第三次迭代：13/34和21/34	第八次迭代：1/3和2/3
第四次迭代：8/21和13/21	第九次迭代：1/2和$\frac{1+\varepsilon}{2}$
第五次迭代：5/13和8/13	

而 **Fibonacci 法**在迭代中计算**试探点**的公式为[5]：

$$\lambda_k = a_k + \frac{F_{n-k-1}}{F_{n-k+1}}(b_k - a_k), k = 1, \dots, n-1 \quad \textbf{(16)}$$

$$\mu_k = a_k + \frac{F_{n-k}}{F_{n-k+1}}(b_k - a_k), k = 1, \dots, n-1 \quad \textbf{(17)}$$

n是计算函数值的次数；第k次迭代后，区间长度的**缩短比率**为F_{n-k}/F_{n-k+1}。



[5] 注，Fibonacci 法确定试探点的位置时，是从后往前倒着使用 Fibonacci 数列的。

✧ Fibonacci 法与 0.618 法的关系

（1）两者都用于单峰函数的一维搜索。

（2）0.618 法可作为 Fibonacci 法的极限形式

$$\lim_{n\to\infty} F_{n-1}/F_n = 0.61803398874\ldots$$

（3）Fibonacci 法精度高于 0.618 法。给定相同试探次数，0.618 法的最终区间约比 Fibonacci 法长 17%。

（4）Fibonacci 法的缺点：需事先知道计算函数值（试探点）的次数（0.618 法不需要）。

在解决实际问题时，一般采用 0.618 法。