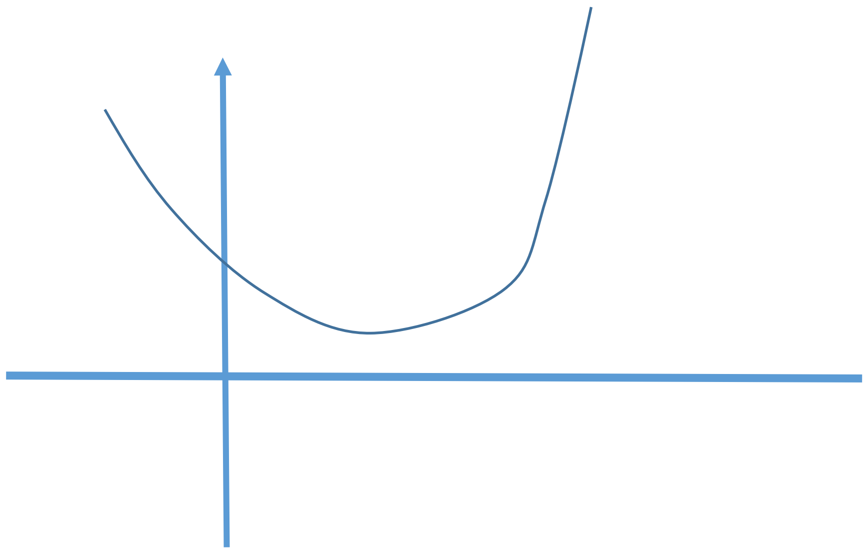
无约束最优化

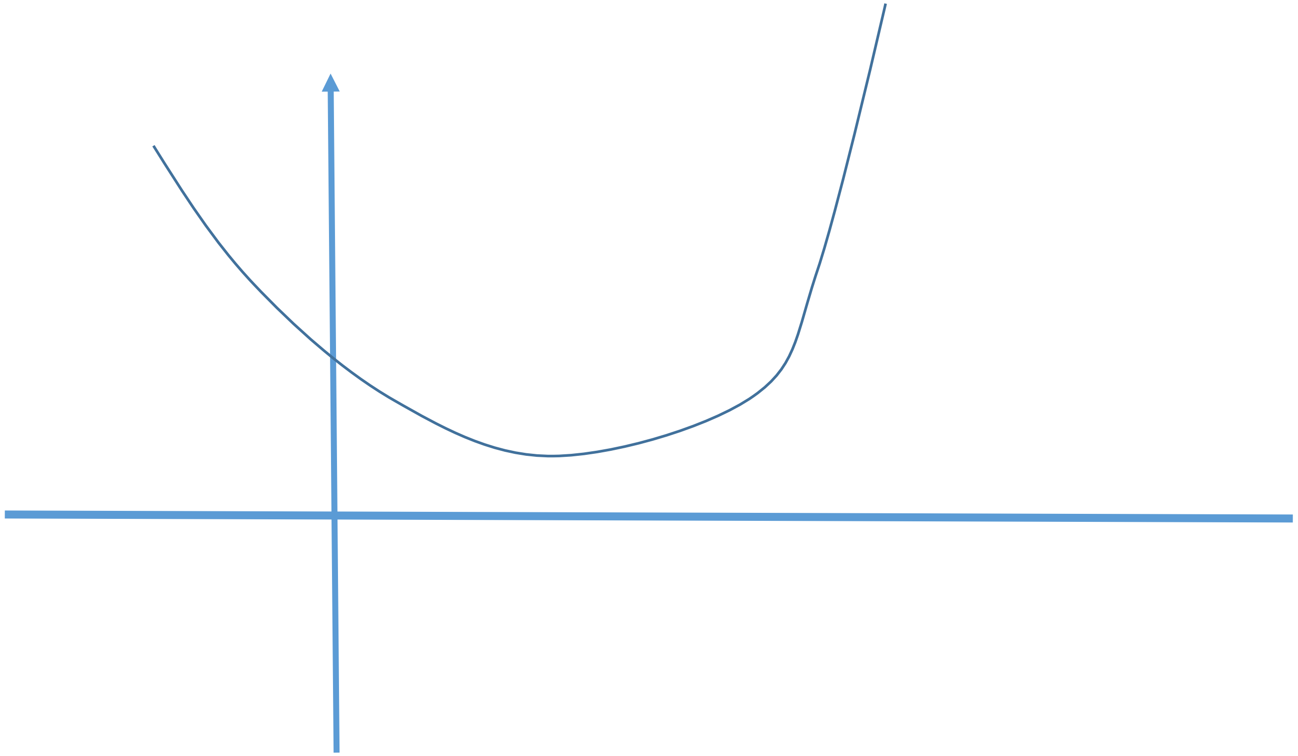
目录

- 单变量情形
  - 函数值可知
    - 直接搜索法
  - 导数可计算
    - 牛顿下降法
- 多变量情形
  - 函数值可知
    - 直接搜索法
    - 共轭方向法
  - 导数可知
    - 梯度法
    - 牛顿法



单变量情形

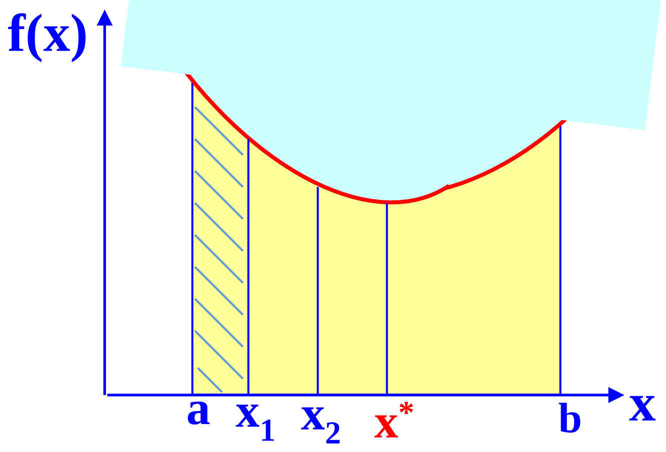
函数值已知



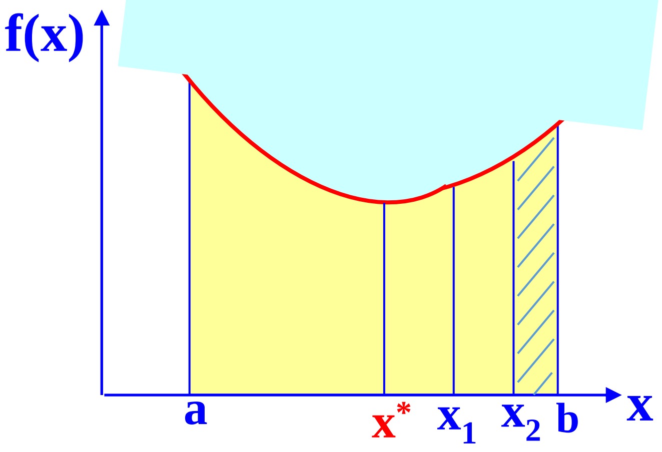
定理： 设f(x)是区间[a,b]上的一个单峰函数，$x^*\in[a,b]$是其极小点，x_1和x_2是[a, b]上的任意两点，且$a<x_1<x_2<b$，那么比较$f(x_1)$与$f(x_2)$的值后，可得出如下结论：

（I） 若$f(x_1)\geq f(x_2)$，$x^*\in[x_1,b]$

（II） 若$f(x_1) < f(x_2)$，$x^*\in[a,x_2]$



(I) 消去$[a, x_1]$



(II) 消去$[x_2, b]$

在单峰函数的区间内，计算两个点的函数值，比较大小后，就能把搜索区间缩小。在已缩小的区间内，仍含有一个函数值，如再计算另一点的函数值，比较后就可进一步缩小搜索区间 。

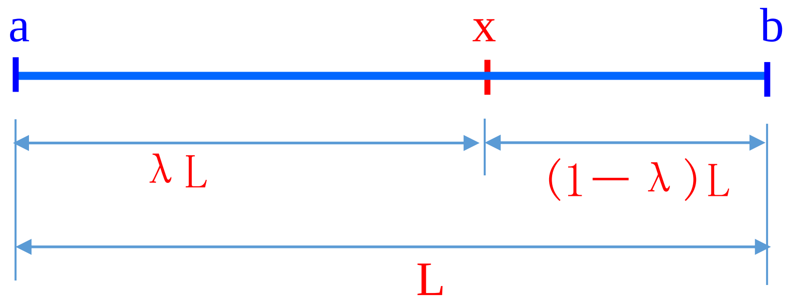
区间消去法—黄金分割法

消去法的思想：反复使用单峰函数的消去性质，不断缩小包含极小点的搜索区间，直到满足精度为止。

消去法的优点：只需计算函数值，通用性强

消去法的设计原则：（1）迭代公式简单；（2）消去效率高

一、黄金分割



$$\frac{b-x}{x-a}=\frac{x-a}{b-a}=\lambda$$

$$\frac{(1-\lambda)L}{\lambda L}=\lambda$$

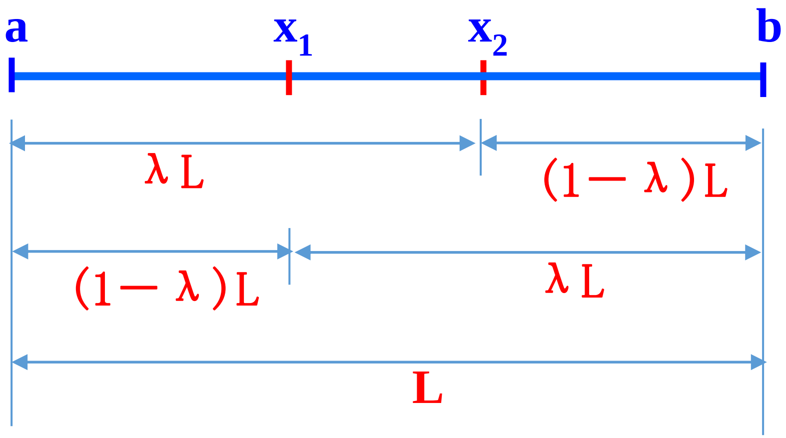
$$\lambda=\frac{-1\pm\sqrt{5}}{2}$$

取“＋”，λ=0.61803398874189

二、黄金分割法的基本思想

黄金分割重要的消去性质:

设x_1，x_2为[a, b] 中对称的两个黄金分割点



x_1为$[a, x_2]$的黄金分割点

$$\frac{x_1-a}{x_2-a}=\frac{(1-\lambda)L}{\lambda L}=\lambda$$

$$x_1+x_2=a+b$$

$$\frac{b-x_2}{b-x_1}=\frac{(1-\lambda)L}{\lambda L}=\lambda$$

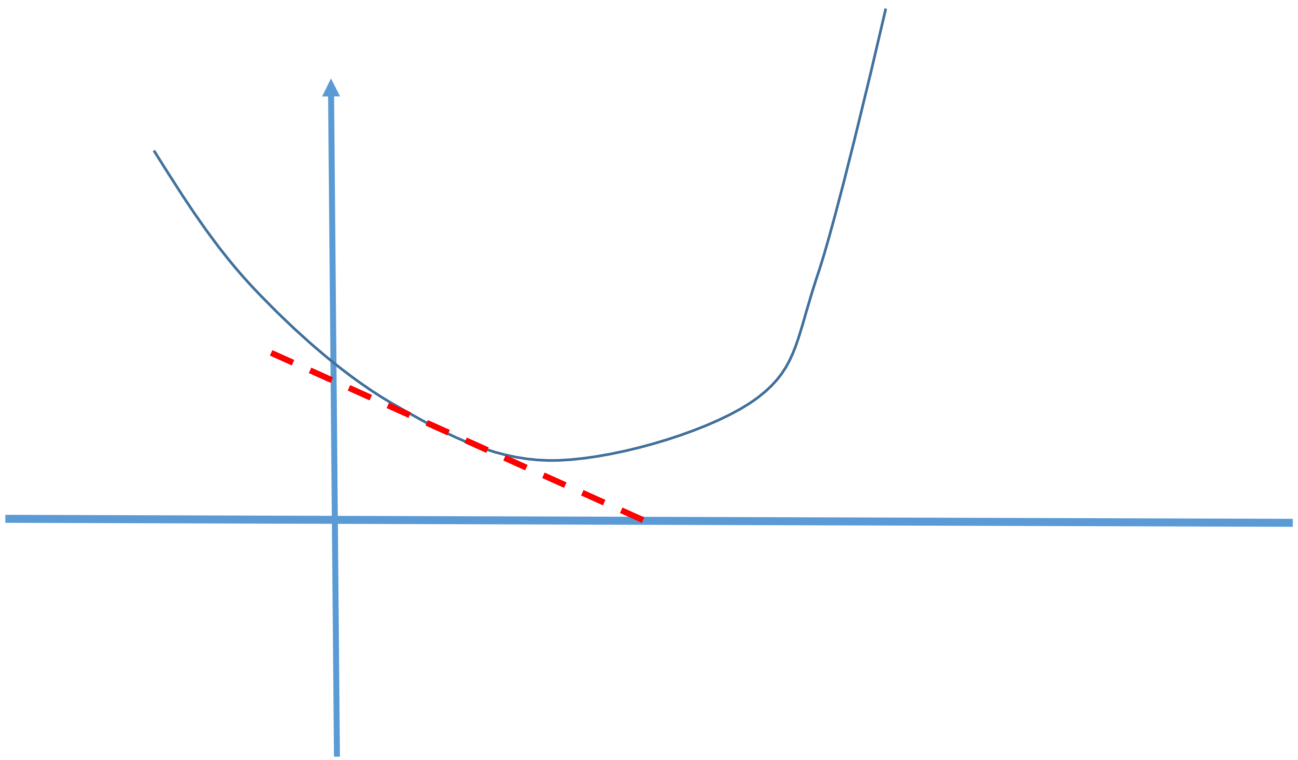
x_2为$[x_1, b]$的黄金分割点

黄金分割比λ≈0.618，所以此法也称为0.618法

在进行区间消去时，不管是消去$[a, x_1]$，还是消去$[x_2, b]$，留下来的区间中还含一个黄金分割点，只要在对称位置找另一个黄金分割点，又可以进行下一次区间消去。

每次消去后，新区间的长度约为原区间的0.618倍，经过n次消去后，保留下来的区间长度为$0.618^n L$，需计算函数值的次数为n+1

导数可计算



要求计算导数的迭代法

如目标函数f(x)可导，可通过解f ’(x)＝0求平稳点，进而求出极值点。

对高度非线性函数，要用逐次逼近求平稳点

Newton-Raphson法

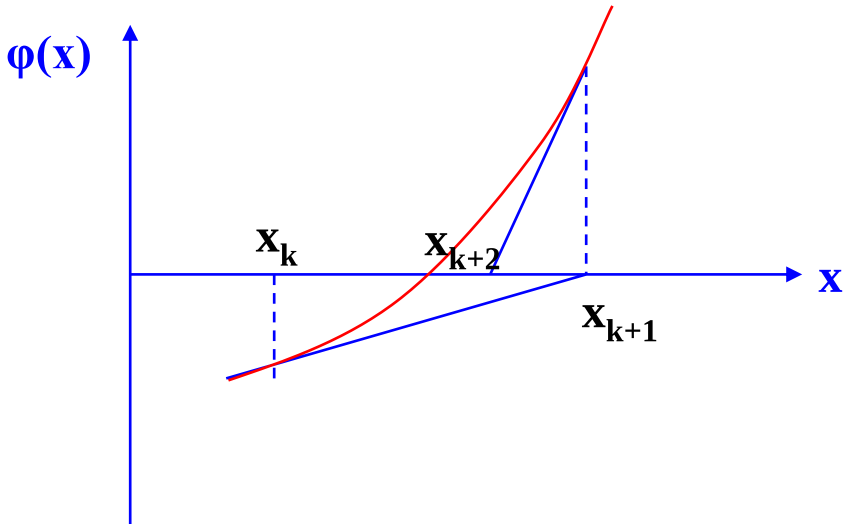
要求目标函数f(x)在搜索区间内具有二阶连续导数，且已知f ’(x)和f ”(x)的表达式

采用迭代法求φ(x) ＝f ’(x)＝0的根

$$\varphi(x)=\varphi(x_k)+\varphi'(x_k)\,(x-x_k)$$

$$x_{K+1}=x_k-\varphi(x_k)/\varphi'(x_k)$$

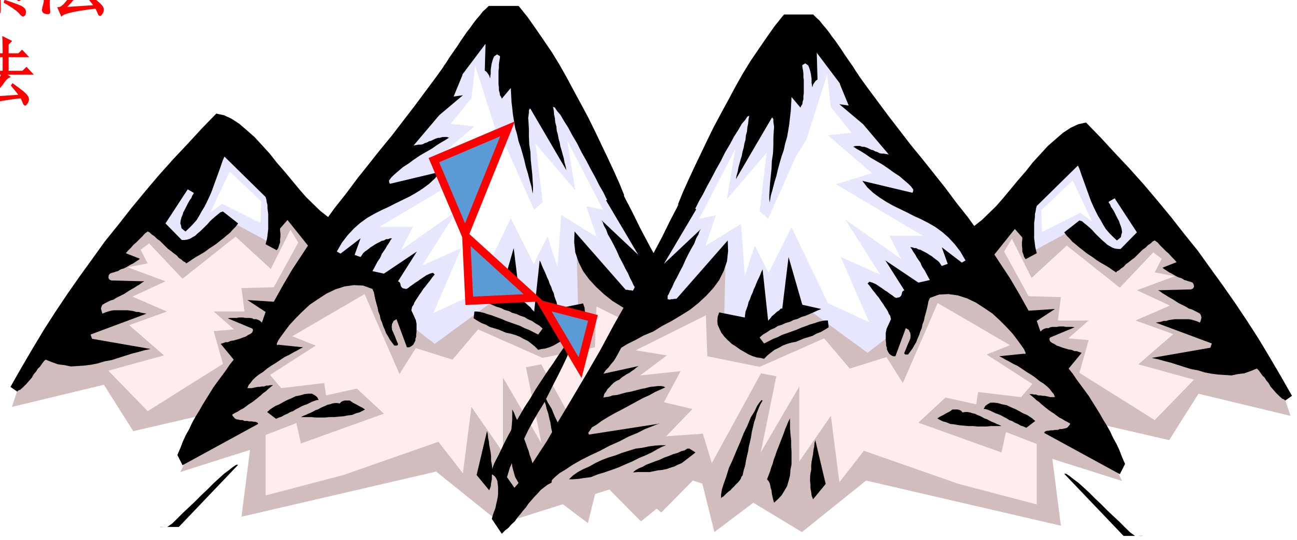
$$x_{K+1}=x_k-f'(x_k)/f''(x_k)$$



多变量情形

函数值已知

1. 单纯形搜索法
2. 共轭方向法

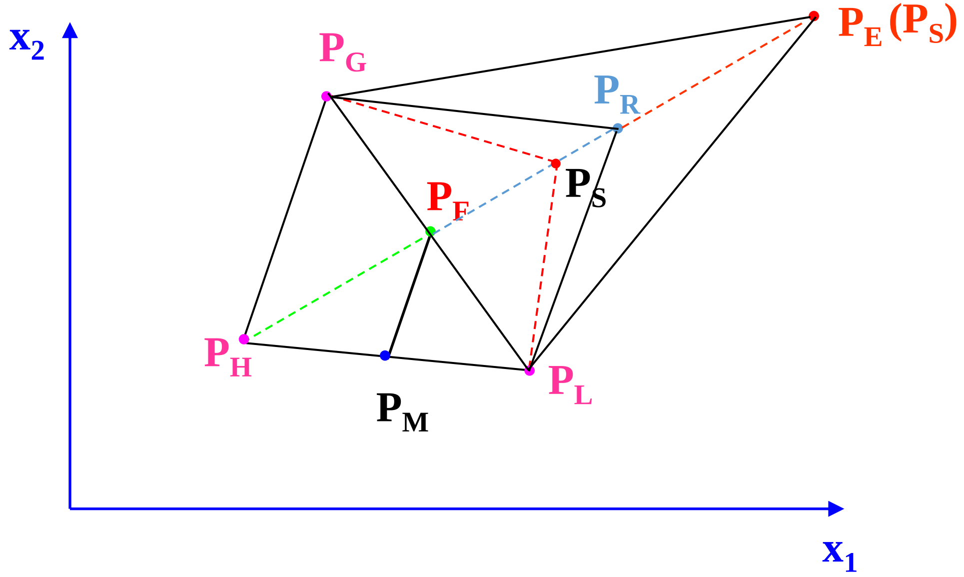


单纯形搜索法

单纯形法的基本思想

Nelder-Mead单纯形法

以 min $f(x_1, x_2)$为例， 说明迭代过程



$f_H > f_G > f_L$

反射P_R

- $f_R<f_G$ 反射成功 ⟹ 扩张P_E
  - $f_E \le f_R$ 扩张成功 $P_S = P_E$
  - $f_E > f_R$ 扩张失败 $P_S = P_R$
- $f_R \ge f_G$ 反射失败 ⟹ 压缩P_S
  - $f_S < f_G$ 压缩成功 P_S
  - $f_S \ge f_G$ 压缩失败 ⟹ 收缩 ⟹ 新单纯形 $P_F P_L P_M$

扩张成功、扩张失败、压缩成功 ⟹ 新单纯形 $P_G P_L P_S$

Powell共轭方向法（方向加速法）

- ♣ 1964年由Powell提出，后经Zangwoll（1967年）和Brent（1973年）改进，是迄今为止最有效的直接搜索法。
- ♥ 该算法有效地利用了迭代过程中的历史信息，建立起能加速收敛的方向
- ♠ 有理论基础，以二次对称函数 $f(x) = c + b^T x + 1/2\, x^T A x$ 为模型进行研究。

? 为什么选择二次函数作为模型 ?

1、在非线性目标函数中，最简单的是二次函数，故任何对一般函数有效的方法首先应对二次函数有效；

2、在最优点附近，非线性函数可用一个二次函数作近似，故对二次函数使用良好的方法，通常对一般函数也有效；

3、很多实际问题的目标函数是二次函数。

（一）共轭方向

- 设A是n×n阶对称正定矩阵，$p^{(0)}, p^{(1)}$为两个n维向量，若 成立

$$p^{(0)T} A\, p^{(1)} = 0$$

则称向量$p^{(0)}$与$p^{(1)}$为A共轭或A正交，称该两向量的方向为A共轭方向。

- 若 A＝I（单位矩阵），$p^{(0)T} p^{(1)} = 0$，即$p^{(0)}$与$p^{(1)}$是正交的。

$=\|p^{(0)}\|.\|p^{(1)}\|\cos\theta$

“正交”是“共轭”的特殊情况

例：

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad p^{(0)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad p^{(1)} = \begin{bmatrix} 1 \\ -2 \end{bmatrix},$$

则 $p^{(0)}$与$p^{(1)}$是A共轭的。因为

$$p^{(0)T} A\, p^{(1)} = [1,0]\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} 1 \\ -2 \end{bmatrix} = [2,1]\begin{bmatrix} 1 \\ -2 \end{bmatrix} = 0$$

? 共轭方向有什么用 ?

（二）共轭方向法的基本定理

定理1： 设A为n×n阶对称正定矩阵，$p^{(0)}, p^{(1)}, \ldots, p^{(n-1)}$为n个相互A共轭的n维非零向量（即$p^{(i)} \neq 0$, $i=0,1,\ldots,n-1$, 且$p^{(i)T} A p^{(j)} = 0$，$i \neq j$），则此向量系必线性无关。

推　论：　在n维空间中，互相共轭的非零向量的个数不超过n个。

定理2： 若$p^{(0)}, p^{(1)}, \ldots, p^{(n-1)}$是n个非零的A共轭向量，则二次目标函数

$f(x) = c + b^T x + 1/2\ x^T A x$的最优点 x^*为

$$x^* = \sum_{i=0}^{n-1}\left[-\frac{p^{(i)T} b}{p^{(i)T} A p^{(i)}}\, p^{(i)}\right]$$

? 上式用于非二次目标函数，可否得到最优点 ？

! 可得到非二次函数最优点的一个近似点；其中A其是目标函数的Hesse矩阵 !

定理3:

设A为n阶对称正定矩阵，对于二次目标函数$f(x) = c + b^T x + 1/2\, x^T A x$，

从任意初始点$x^{(0)}$出发，逐次进行一维搜索，即

$$\min f(x^{(i)} + t\, p^{(i)}) = f(x^{(i)} + t_i\, p^{(i)}) \qquad i \geqslant 0$$

若搜索方向$p^{(0)}, p^{(1)}, \ldots, p^{(n-1)}$是非零的A共轭向量，则至多进行n次迭代必可得到极小点x^*，即

$x^{(i+1)} = x^{(i)} + t_i\, p^{(i)}$，其中$t_i$为最优步长因子， $i = 0,1,\ldots,n-1$

$x^* = x^{(k)},\ 0 \leqslant k \leqslant n$

？ 对非二次函数，采用上述方法，n 次迭代是否也可得到极小点 ？

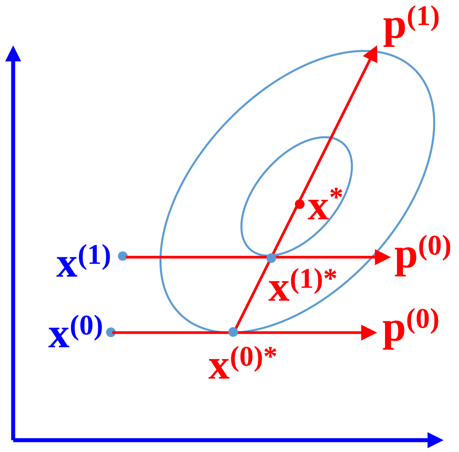
？ 如何简便地找出n个相互A共轭的向量？

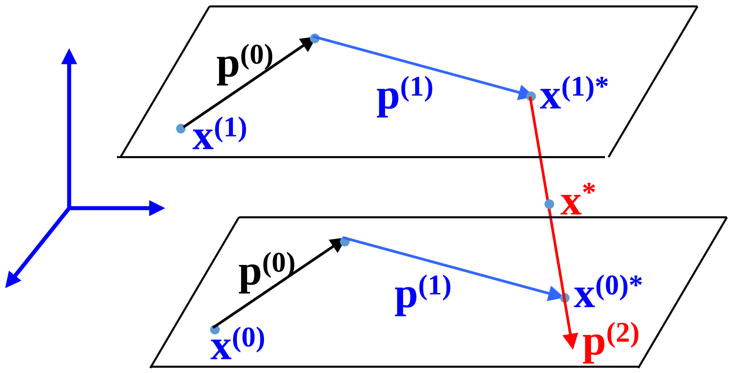
定理4:

假设

1. n元函数$f(x) = c + b^T x + 1/2\, x^T A x$ 中的矩阵A是对称正定的；
2. 向量$p^{(0)}, p^{(1)}, \ldots, p^{(m-1)}$ $(m<n)$是互相A共轭的；
3. $x^{(0)}, x^{(1)}$是不同的任意两点，分别从$x^{(0)}, x^{(1)}$出发，依次沿$p^{(0)}, p^{(1)}, \ldots, p^{(m-1)}$作一维精确搜索，设最后一次一维搜索的极小点分别为$x^{(0)*}$和$x^{(1)*}$，那么，向量 $p = x^{(0)*} - x^{(1)*}$与$p^{(0)}, p^{(1)}, \ldots, p^{(m-1)}$互为A共轭。

已知前m个共轭方向，
就可以找到第m+1个共轭方向





（三）Powell共轭方向法的基本思想

任意n个线性无关的方向

阶段起点$\mathbf{x}^{(k,0)}$	n+1次一维搜索过程	新共轭方向 $\tilde{p}^{(k)}$

$$x^{(0)} = x^{(1,0)} \xrightarrow{p^{(0)}} x^{(1,1)} \xrightarrow{p^{(1)}} x^{(1,2)} \cdots x^{(1,n-2)} \xrightarrow{p^{(n-2)}} x^{(1,n-1)} \xrightarrow{p^{(n-1)}} x^{(1,n)} \xrightarrow{\tilde{p}^{(1)}=x^{(1,n)}-x^{(1,0)}} x^{(1,n+1)}$$

$$x^{(1,n+1)} = x^{(2,0)} \xrightarrow{p^{(1)}} x^{(2,1)} \xrightarrow{p^{(2)}} x^{(2,2)} \cdots x^{(2,n-2)} \xrightarrow{p^{(n-1)}} x^{(2,n-1)} \xrightarrow{\tilde{p}^{(1)}} x^{(2,n)} \xrightarrow{\tilde{p}^{(2)}=x^{(2,n)}-x^{(2,0)}} x^{(2,n+1)}$$

$$x^{(2,n+1)} = x^{(3,0)} \xrightarrow{p^{(2)}} x^{(3,1)} \xrightarrow{p^{(3)}} x^{(3,2)} \cdots x^{(3,n-2)} \xrightarrow{\tilde{p}^{(1)}} x^{(3,n-1)} \xrightarrow{\tilde{p}^{(2)}} x^{(3,n)} \xrightarrow{\tilde{p}^{(3)}=x^{(3,n)}-x^{(3,0)}} x^{(3,n+1)}$$

……………………………………………………………………

$$x^{(n-1,n+1)} = x^{(n,0)} \xrightarrow{p^{(n-1)}} x^{(n,1)} \xrightarrow{\tilde{p}^{(1)}} x^{(n,2)} \cdots x^{(n,n-2)} \xrightarrow{\tilde{p}^{(n-2)}} x^{(n,n-1)} \xrightarrow{\tilde{p}^{(n-1)}} x^{(n,n)} \xrightarrow{\tilde{p}^{(n)}=x^{(n,n)}-x^{(n,0)}} x^{(n,n+1)}$$

共分n个阶段，每一阶段都进行n+1次搜索，最后产生一个共轭方向

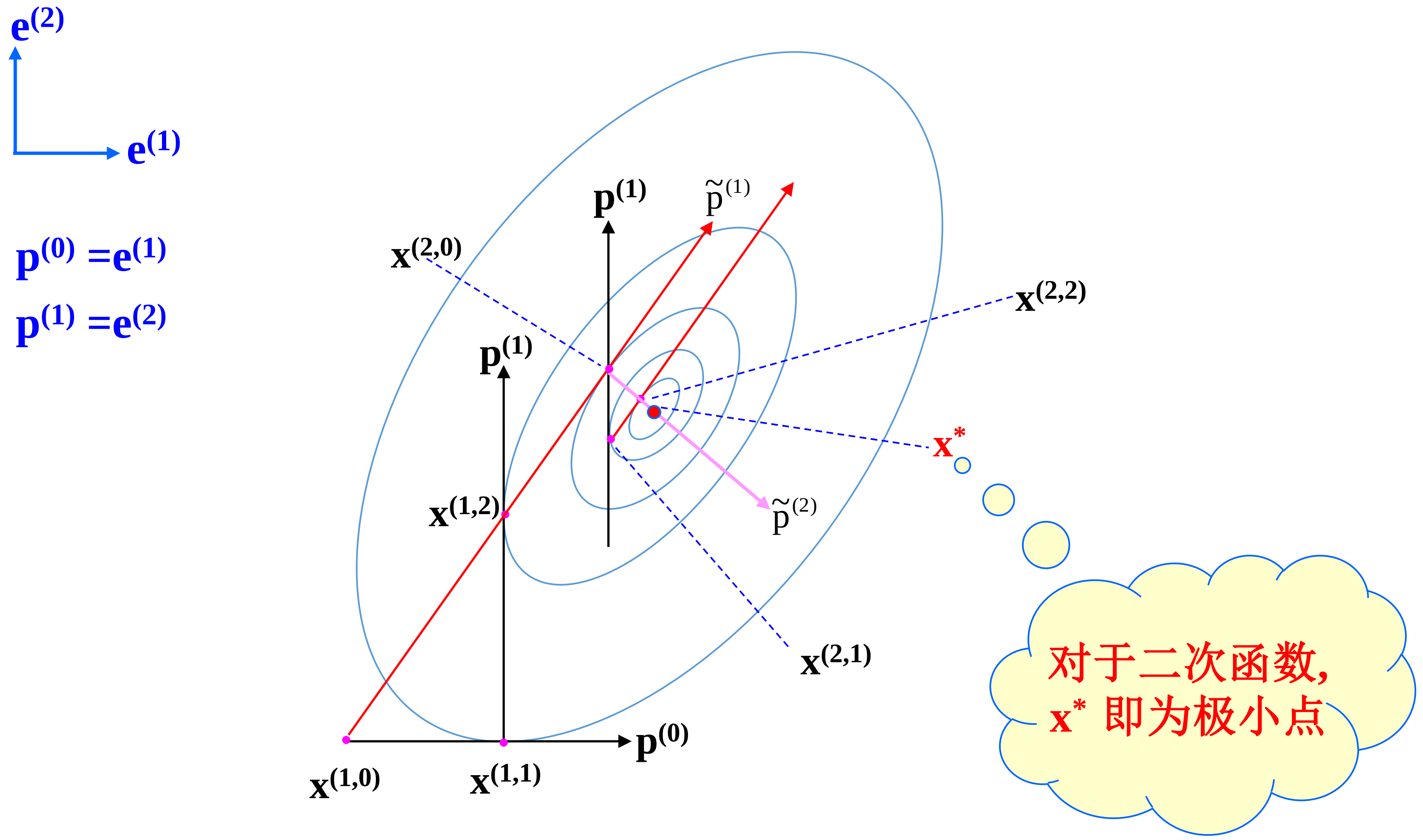
一边搜索，
一边找共轭方向

二维空间中二次函数的Powell共轭方向法示意

以二次函数$f(x_1, x_2)$为例

$p^{(0)}=e^{(1)}$

$p^{(1)}=e^{(2)}$



梯度方法

梯度法

直接搜索法收敛速度一般比较慢，需要计算大量的函数值。梯度反映了函数值变化的规律，充分利用梯度信息构造算法，能加速收敛。

使用函数的梯度（一阶导数）或Hesse矩阵（二阶导数）的优化算法称为梯度法

目标： 求出平稳点（满足$\nabla f(x)=0$的x^*）。

由于 $\nabla f(x)=0$ 一般是非线性方程组，解析法往往行不通，所以梯度法通常也是逐次逼近的迭代法

假定： $\nabla f(x)$和$\nabla^2 f(x)$连续存在

最速下降法(Cauchy法)

1847年Cauchy提出。特点是直观易懂，但收敛速度慢

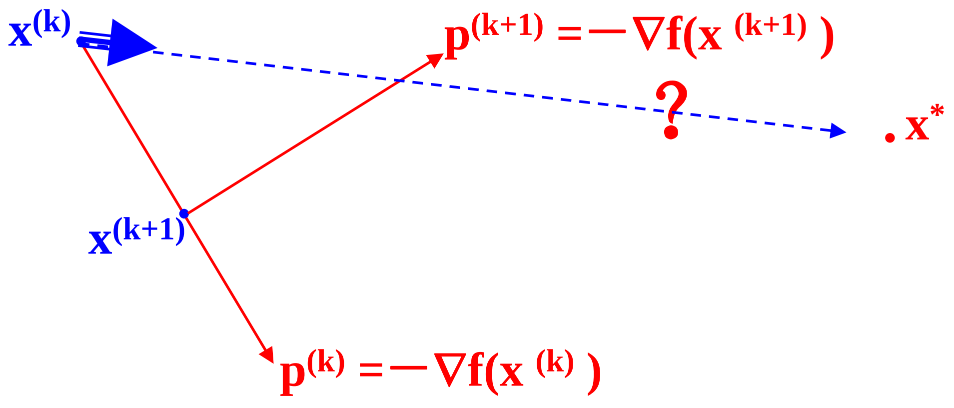
（一）基本思想

多变量最优化迭代解法的一般迭代公式

$$x^{(k+1)} = x^{(k)} + t_k \, p^{(k)}$$

可用一维搜索技术解决

关键是如何确定搜索方向$p^{(k)}$



瞎子下山：由于他看不到哪里是山谷，不可能沿直接指向山谷的路线走，他只能在当前位置上，靠手杖作局部探索，哪里最陡就往哪里前进一步，然后在新的位置上再用手杖寻找最陡方向，再下降一步。这就是最速下降法的形象比喻。

迭代公式 $x^{(k+1)} = x^{(k)} - t_k \nabla f(x^{(k)})$

Newton法（二阶方法）

? 由最速下降法可知，从全局角度来看，负梯度方向一般不是一个特别好的方向，有没有更好的方向 ?

（一）基本Newton法

$$f(x) = f(x^{(k)}) + \nabla f(x^{(k)})^T \Delta x + \frac{1}{2}\Delta x^T \nabla^2 f(x^{(k)}) \Delta x + \cancel{0(\Delta x^3)} \approx f(x; x^{(k)})$$

f(x)在$x^{(k)}$处的二次近似函数

取 $f(x; x^{(k)})$的平稳点作为f(x) 最优点的一个近似点

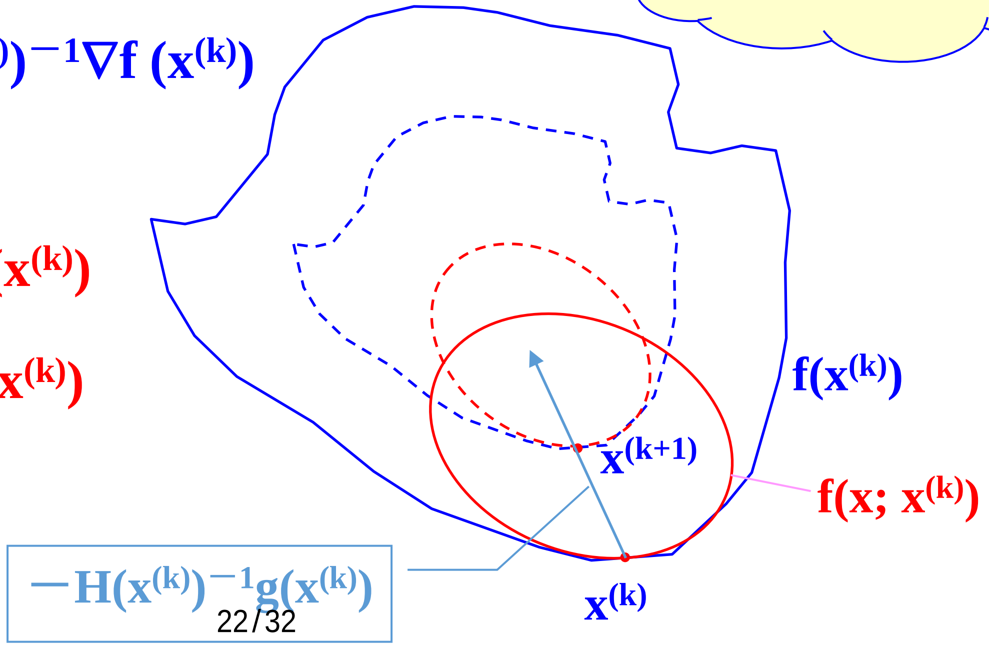
$$\nabla f(x; x^{(k)}) = \nabla f(x^{(k)}) + \nabla^2 f(x^{(k)})\Delta x = 0$$

$$x - x^{(k)} = \Delta x = -\nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)})$$

Newton迭代公式

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

或 $x^{(k+1)} = x^{(k)} - H(x^{(k)})^{-1}g(x^{(k)})$



Marquardt法

1963年Marquardt提出将最速下降法与Newton法结合，开始用最速下降法，在接近最优点时用Newton法。

（一）方法思想

在迭代公式$x^{(k+1)} = x^{(k)} + t_k\, p^{(k)}$中，取步长$t_k = 1$，搜索方向为

$$p^{(k)} = -[\nabla^2 f(x^{(k)}) + \lambda_k I]^{-1} \nabla f(x^{(k)})$$

其中 λ_k同时起控制搜索方向和步长的作用，I为单位矩阵

(1) 开始阶段取很大，如$\lambda_0 = 10^4$，

$$p^{(0)} = -[\nabla^2 f(x^{(0)}) + \lambda_0 I]^{-1} \nabla f(x^{(0)}) \approx -\frac{1}{\lambda_0} \nabla f(x^{(0)})$$ ⇒ 最速下降法

(2) 在迭代过程中，让$\lambda_k \to 0$，

$$p^{(k)} \to -\nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$$ ⇒ Newton法

具体在每一步是否缩小 λ_k，要通过检验目标函数值来决定：

若$f(x^{(k+1)}) < f(x^{(k)})$，取$\lambda_{k+1} < \lambda_k$；否则，取$\lambda_k = \beta\lambda_k$, $\beta > 1$，重作第k步迭代。

非线性最小二乘问题

（一）最小二乘问题

在工程实际问题中，经常遇到一类特殊的求极小值问题，其目标函数具有平方和形式：

$$F(x)=\sum_{i=1}^{m} f_i^2(x),\quad m \ge n$$

例 求解方程组$f_i(x)=0$，（$i=1,2,\ldots,m$)的问题可化为求解下列优化问题

$$\min\ F(x)=\sum_{i=1}^{m} f_i^2(x)$$

例 通过m组实验数据来建立物理量y与另外l个物理量$t_1, t_2,\ldots, t_l$ 之间的函数关系：

$$y = Y(t_1, t_2, \ldots, t_l; x_1, x_2, \ldots, x_n)$$

即要确定其中n个待定参数$x_1,x_2,\ldots,x_n$，使得经验公式的计算值$\tilde{y}^{(i)}$与实验值$y^{(i)}$尽可能地接近。“接近”的衡量标准常用平方和的形式

$$F(x)=\sum_{i=1}^{m}(\tilde{y}^{(i)}-y^{(i)})^2=\sum_{i=1}^{m}[Y(t_1^{(i)},t_2^{(i)},\ldots,t_l^{(i)};x_1,x_2,\ldots,x_2)-y^{(i)}]^2=\sum_{i=1}^{m} f_i^2(x)$$

求解min F(x)

为求解方便，引入向量函数 $f(x)=[f_1(x),f_2(x),\ldots,f_m(x)]^T$

最小二乘问题化为： $\min F(x) = \min f(x)^T f(x) = \min \|f(x)\|^2$

（二）最小二乘问题的求解

可直接用前面介绍的单纯形法、Powell共轭方向法、最速下降法、Newton法、Marquart法求解

！ 如用Newton法求解，则要求F(x)的Hesse矩阵，是否可以不求H(x)？

- 最小二乘法 （Gauss-Newton法）

Newton方向： $p^{(k)} = -\nabla^2 F(x^{(k)})^{-1} \nabla F(x^{(k)})$

由于最小二乘问题目标函数形式的特殊性，可用计算一阶导数来代替二阶导数的计算：

Jacobi 矩阵 $J(x)=[J_{ij}(x)]_{m\times n}$ $J_{ij}(x) = \dfrac{\partial f_i(x)}{\partial x_j}$

$$\nabla F(x^{(k)})=2J(x^{(k)})^T f(x^{(k)})$$

$$\nabla^2 F(x^{(k)}) \approx 2J(x^{(k)})^T J(x^{(k)})$$

迭代公式：$x^{(k+1)} = x^{(k)} - [J(x^{(k)})^T J(x^{(k)})]^{-1} J(x^{(k)})^T f(x^{(k)})$

共轭梯度法

由Powell共轭方向法可知，共轭方向是好方向，是否有比Powell共轭方向法更简单的方法构建共轭方向？

（一）Fletcher－Reeves共轭梯度法的基本思想

任取初始点$x^{(0)}$，然后沿着逐次迭代产生的共轭方向$p^{(k)}$进行一维搜索：

$$x^{(k+1)} = x^{(k)} + t_k\, p^{(k)}$$

得到下一个迭代点。

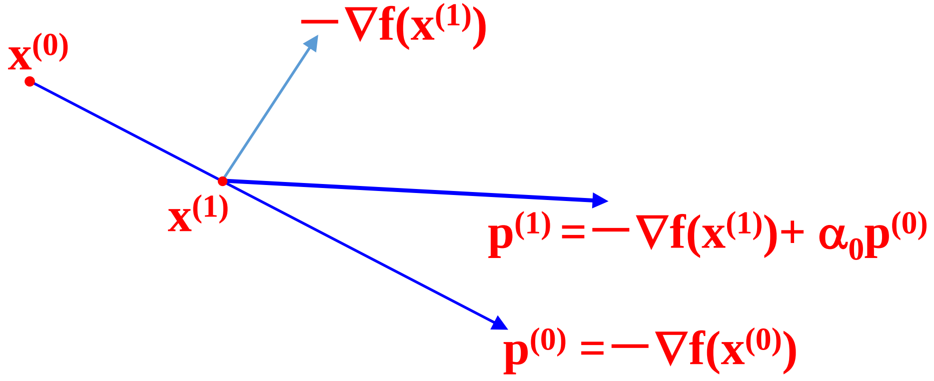
当f(x)为二次函数时，至多经过n次迭代就可得到极小点

构造共轭方向$p^{(0)},\ldots,p^{(n-1)}$的方法：

$$p^{(0)} = -\nabla f(x^{(0)})$$

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

即下一个共轭方向是当前点处负梯度方向与已求得的最后一个共轭方向的线性组合。



！ 要使得$p^{(0)},\ldots,p^{(n-1)}$相互共轭，

显然α_k不能随便取 ！

以二次目标函数为模型，经推导得：

$$\alpha_k=\frac{\nabla f(x^{(k+1)})^T\nabla f(x^{(k+1)})}{\nabla f(x^{(k)})^T\nabla f(x^{(k)})}=\frac{\left\|\nabla f(x^{(k+1)})\right\|^2}{\left\|\nabla f(x^{(k)})\right\|^2}$$

记 $g^{(k)}=\nabla f(x^{(k)})$

F－R共轭梯度法的迭代公式：

$$\begin{cases} x^{(k+1)}=x^{(k)}+t_k p^{(k)} \\ p^{(k+1)}=-g^{(k)}+\alpha_k p^{(k)} \\ \alpha_k=\dfrac{\|g^{(k+1)}\|^2}{\|g^{(k)}\|^2} \\ p^{(0)}=-g^{(0)} \end{cases}$$

例 用共轭梯度法解 $\min f(x) = 60 - 10x_1 - 4x_2 + x_1^2 + x_2^2 - x_1x_2$

初始点取为$x^{(0)}=[0, 0]^T$。

解：$\nabla f(x) = [-10 + 2x_1 - x_2,\ -4 + 2x_2 - x_1]^T$

$p^{(0)} = -g^{(0)} = -\nabla f(x^{(0)}) = [10, 4]^T$

进行一维搜索，对简单f(x)，可用解析法求解：

$f(x^{(1)}) = f(x^{(0)} + t\,p^{(0)}) = f(x)|_{x=[10t,\ 4t]^T} = 60 - 116t + 76t^2$

$f'(t) = 1520t - 116 = 0 \qquad t_0 = 0.76315789$

$x^{(1)} = x^{(0)} + t_0\,p^{(0)} = [7.63157894, 3.052631578]^T$

$g^{(1)} = \nabla f(x^{(1)}) = [2.21052631, -5.52631579]^T$

$a_0 = \|g^{(1)}\|^2 / \|g^{(0)}\|^2 = 35.42659277 / 116$

$p^{(1)} = -g^{(1)} + \alpha_0 p^{(0)} = [0.84349308, 6.747922437]^T$

再用解析法求最优步长 t_1 得

$t_1 = 0.436781609$

$x^{(2)} = x^{(1)} + t_1\,p^{(1)} = [7.999999993, 5.999999997]^T$

$g^{(2)} = \nabla f(x^{(2)}) = [0, 0]^T$

所以，$x^* = [8, 6]^T \qquad f^* = 8$

（二）拟Newton法的基本思想

Newton方向　$p^{(k)}=-\nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)})$

？不想求Hesse矩阵及其逆矩阵，有什么办法？

从形式上模仿，构造一个方向：　$p^{(k)}=-H^{(k)}\nabla f(x^{(k)})$

尺度矩阵$H^{(k)}$既近似$\nabla^2 f(x^{(k)})^{-1}$，计算又要方便！

$H^{(k)}$应满足的条件：

(1) 满足拟Newton条件：$x^{(k+1)}-x^{(k)} = H^{(k+1)}[\nabla f(x^{(k+1)})-\nabla f(x^{(k)})]$

(2) $H^{(k)}$为正定矩阵，这样$p^{(k)}$为下降方向

(3) 由$H^{(k)}$出发计算$H^{(k+1)}$应简便：$H^{(k+1)}=H^{(k)}+\underline{E^{(k)}}$

(4) 应使算法具有二次收敛性。

校正矩阵

总 结

多变量最优化迭代解法的一般迭代公式

$$x^{(k+1)} = x^{(k)} + \boxed{t_k}\ \boxed{p^{(k)}}$$

最优步长
可用一维搜索技术解决

不同的搜索方向$p^{(k)}$,
构成不同的算法

算法名称	搜索方向$p^{(k)}$
共轭方向法	共轭方向
最速下降法	$p^{(k)} = -\nabla f(x^{(k)})$
Newton法	$p^{(k)} = -\nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)})$
Marquart法	$p^{(k)} = -[\nabla^2 f(x^{(k)}) + \lambda_k I]^{-1}\nabla f(x^{(k)})$
F-R共轭梯度法	$p^{(0)} = -\nabla f(x^{(0)})$ $p^{(k+1)} = -\nabla f(x^{(k+1)}) + \alpha_k p^{(k)}$
（拟Newton法）	$p^{(0)} = -\nabla f(x^{(0)})$ $p^{(k)} = -H^{(k)}\nabla f(x^{(k)})$

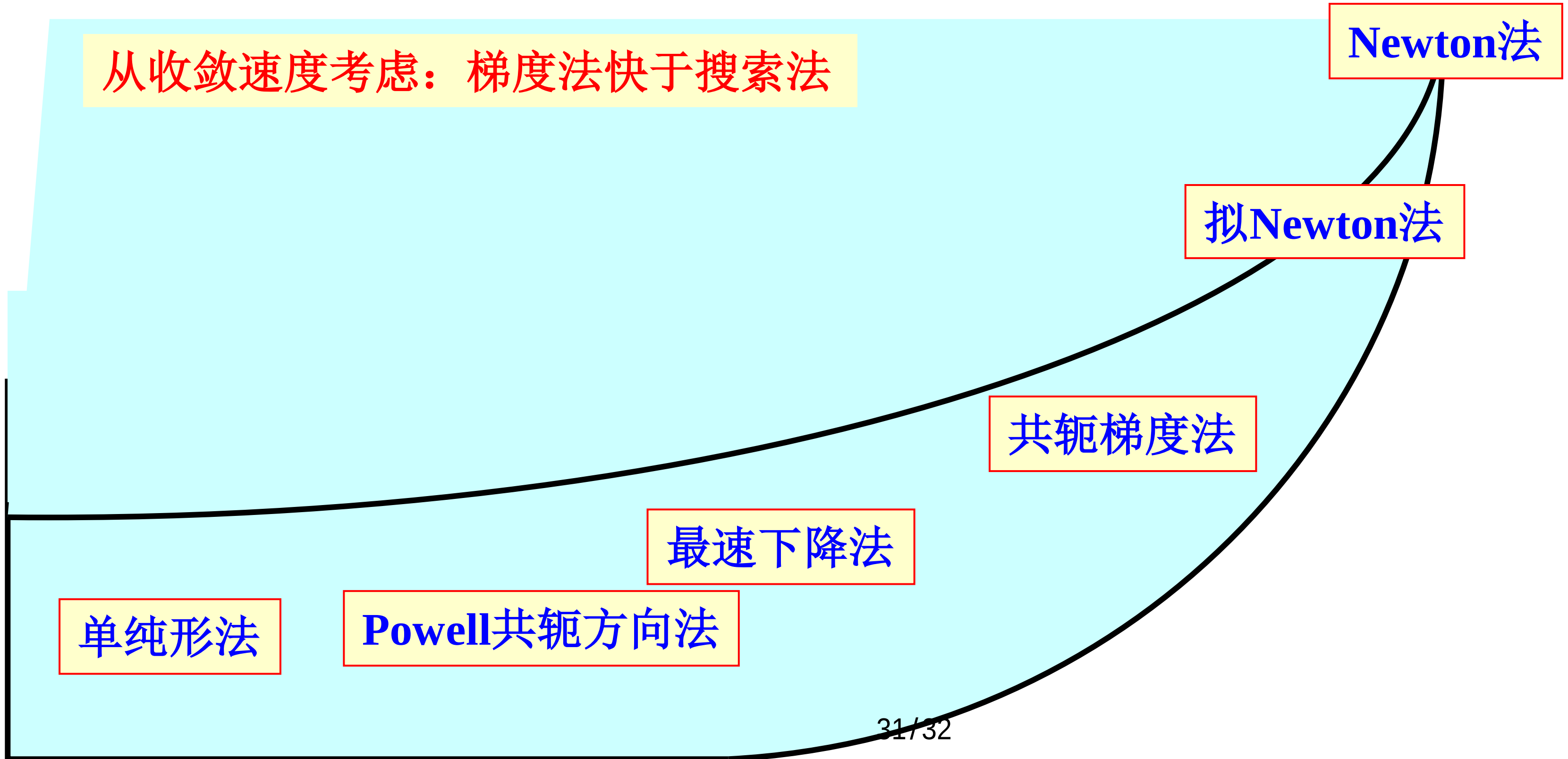
方法的比较与选择

1、直接法：对简单问题，求解必要条件或充分条件；

2、迭代算法：

- 零阶法：只需计算函数值 f(x) → 搜索法
- 一阶法：需计算一阶导数 $\nabla f(x)$
- 二阶法：需计算二阶导数 $\nabla^2 f(x)$

一阶法、二阶法 → 梯度法



Adam: A Method for Stochastic Optimization

Diederik P. Kingma*
University of Amsterdam, OpenAI
dpkingma@openai.com

Jimmy Lei Ba*
University of Toronto
jimmy@psi.utoronto.ca

Algorithm 1: *Adam*, our proposed algorithm for stochastic optimization. See section 2 for details, and for a slightly more efficient (but less clear) order of computation. g_t^2 indicates the elementwise square $g_t \odot g_t$. Good default settings for the tested machine learning problems are $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. All operations on vectors are element-wise. With β_1^t and β_2^t we denote β_1 and β_2 to the power t.

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector
  $m_0 \leftarrow 0$ (Initialize 1st moment vector)
  $v_0 \leftarrow 0$ (Initialize 2nd moment vector)
  $t \leftarrow 0$ (Initialize timestep)
  while θ_t not converged **do**
    $t \leftarrow t + 1$
    $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)
    $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
    $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
    $\widehat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
    $\widehat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
    $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \widehat{m}_t / (\sqrt{\widehat{v}_t} + \epsilon)$ (Update parameters)
  end while
  return θ_t (Resulting parameters)

3rd International Conference for Learning Representations, San Diego, 2015