最优化方法及其 Matlab 程序设计

马昌凤

2010 年 3 月 22 日

目　录

第五章　拟牛顿法 **1**

5.1　拟牛顿法及其性质 2

5.2　BFGS 算法及其 Matlab 实现 9

5.3　DFP 算法及其 Matlab 实现 19

5.4　Broyden 族算法及其 Matlab 实现 24

5.5　拟牛顿法的收敛性 37

第五章　拟牛顿法

第 3 章所介绍的牛顿法的优点是具有二阶收敛速度, 但当 Hesse 阵 $G(x_k) = \nabla^2 f(x_k)$ 不正定时, 不能保证所产生的方向是目标函数在 x_k 处的下降方向. 特别地, 当 $G(x_k)$ 奇异时, 算法就无法继续进行下去. 尽管修正牛顿法可以克服这一缺陷, 但其中的修正参数 μ_k 的选取很难把握, 过大或过小都会影响到收敛速度. 此外, 牛顿法的每一迭代步都需要目标函数的二阶导数, 即 Hesee 阵, 对于大规模问题其计算量是惊人的.

本章即将介绍的拟牛顿法克服了这些缺点, 并且在一定条件下这类算法仍然具有较快的收敛速度 — 超线性收敛速度.

5.1　拟牛顿法及其性质

拟牛顿法的基本思想是在基本牛顿法的步 2 中用 Hesee 阵 $G_k = \nabla^2 f(x_k)$ 的某个近似矩阵 B_k 取代 G_k. 通常, B_k 应具有下面的三个特点:

(1) 在某种意义下有 $B_k \approx G_k$, 使相应的算法产生的方向近似于牛顿方向, 以确保算法具有较快的收敛速度.

(2) 对所有的 k, B_k 是对称正定的, 从而使得算法所产生的方向是函数 f 在 x_k 处下降方向.

(3) 矩阵 B_k 更新规则相对比较简单, 即通常采用一个秩 1 或秩 2 矩阵进行校正.

下面介绍满足这三个特点的矩阵 B_k 的构造. 设 $f : \mathbb{R}^n \to \mathbb{R}$ 在开集 $D \subset \mathbb{R}^n$ 上二次连续可微. 那么, f 在 x_{k+1} 处的二次近似模型为

$$f(x) \approx f(x_{k+1}) + g_{k+1}^T(x - x_{k+1}) + \frac{1}{2}(x - x_{k+1})^T G_{k+1}(x - x_{k+1}).$$

对上式求导数得

$$g(x) \approx g_{k+1} + G_{k+1}(x - x_{k+1}).$$

令 $x = x_k$, 位移 $s_k = x_{k+1} - x_k$, 梯度差 $y_k = g_{k+1} - g_k$, 则有

$$G_{k+1} s_k \approx y_k.$$

注意到, 对于二次函数 f, 上式是精确成立的. 现在, 我们要求在拟牛顿法中构造出 Hesse 阵的近似矩阵 B_k 满足这种关系式, 即

$$B_{k+1} s_k = y_k. \tag{5.1}$$

上式通常称作拟牛顿方程或拟牛顿条件. 令 $H_{k+1} = B_{k+1}^{-1}$, 则得到拟牛顿方程的另一个形式:

$$H_{k+1} y_k = s_k, \tag{5.2}$$

其中 H_{k+1} 是 Hesse 阵逆的近似. 搜索方向由 $d_k = -H_k g_k$ 或 $B_k d_k = -g_k$ 确定. 根据 B_k(或 H_k) 的第三个特点, 可令

$$B_{k+1} = B_k + E_k, \quad H_{k+1} = H_k + D_k, \tag{5.3}$$

其中 E_k, D_k 是秩 1 或秩 2 矩阵. 通常将由拟牛顿方程 (5.1) (或 (5.2)) 和校正规则 (5.3) 所确定的方法称为拟牛顿法.

下面我们介绍一个对称秩 1 校正公式. 在 (5.3) 中取 $E_k = \alpha u_k u_k^T$ (秩 1 矩阵), 其中 $\alpha \in \mathbb{R}$, $u_k \in \mathbb{R}^n$. 由拟牛顿方程 (5.1) 得

$$(B_k + \alpha u_k u_k^T) s_k = y_k,$$

即有

$$\alpha(u_k^T s_k) u_k = y_k - B_k s_k. \tag{5.4}$$

上式表明向量 u_k 平行于 $(y_k - B_k s_k)$, 即存在常数 β 使得 $u_k = \beta(y_k - B_k s_k)$. 因此有

$$E_k = \alpha\beta^2 (y_k - B_k s_k)(y_k - B_k s_k)^T.$$

于是, 由 (5.4) 得

$$\alpha\beta^2 [(y_k - B_k s_k)^T s_k](y_k - B_k s_k) = (y_k - B_k s_k).$$

由此, 若 $(y_k - B_k s_k)^T s_k \neq 0$, 可取 $\alpha\beta^2 [(y_k - B_k s_k)^T s_k] = 1$, 即

$$\alpha\beta^2 = \frac{1}{(y_k - B_k s_k)^T s_k}, \quad E_k = \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{(y_k - B_k s_k)^T s_k}.$$

故得对称秩 1 校正公式如下:

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{(y_k - B_k s_k)^T s_k}. \tag{5.5}$$

类似地, 利用拟牛顿方程 (5.2), 对 H_k 进行对称秩 1 修正可得

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^T}{(s_k - H_k y_k)^T y_k}. \tag{5.6}$$

有了对称秩 1 校正公式后, 利用它可以构造求解无约束优化问题的一个拟牛顿算法, 步骤如下:

算法 5.1 (**对称秩** 1 **算法**)

步 0 给定初始点 $x_0 \in \mathbb{R}^n$, 终止误差 $0 \le \varepsilon \ll 1$. 初始对称正定阵 H_0 (通常取单位阵 I_n). 令 $k := 0$.

步 1 若 $\|g_k\| \le \varepsilon$, 停算, 输出 x_k 作为近似极小点.

步 2 计算搜索方向 $d_k = -H_k g_k$.

步 3 用线搜索技术求步长 α_k.

步 4　令 $x_{k+1}=x_k+\alpha_k d_k$, 由对称秩 1 校正公式 (5.6) 确定 H_{k+1}.

步 5　令 $k:=k+1$, 转步 1.

下面给出基于 Armijo 搜索的对称秩 1 算法的 Matlab 程序.

程序 5.1 (**对称秩 1 算法程序**)

```
function [x,val,k]=sr1(fun,gfun, x0)
%功能: 用对称秩1算法求解无约束问题:  min f(x)
%输入: x0是初始点, fun, gfun分别是目标函数及其梯度
%输出:  x, val分别是近似最优点和最优值,  k是迭代次数.
maxk=500;   %给出最大迭代次数
rho=0.55;sigma=0.4; epsilon=1e-5;
k=0;   n=length(x0); Hk=eye(n);
while(k<maxk)
   gk=feval(gfun,x0); %计算梯度
   dk=-Hk*gk; %计算搜索方向
```

```
    if(norm(gk)<epsilon), break; end  %检验终止准则
    m=0; mk=0;
    while(m<20)   % 用Armijo搜索求步长
     if(feval(fun,x0+rho^m*dk)<feval(fun,x0)+sigma*rho^m*gk'*dk)
         mk=m;  break;
     end
    m=m+1;
    end
    x=x0+rho^mk*dk;
    sk=x-x0;  yk=feval(gfun,x)-gk;
    Hk=Hk+(sk-Hk*yk)*(sk-Hk*yk)'/((sk-Hk*yk)'*yk); %秩1校正
    k=k+1;   x0=x;
  end
  val=feval(fun,x0);
```

例 5.1 利用程序 5.1 求解无约束优化问题

$$\min\ f(x)=100(x_1^2-x_2)^2+(x_1-1)^2,\ \ x=(x_1,x_2)^T\in\mathbb{R}^2.$$

该问题有精确解 $x^*=(1,1)^T$, $f(x^*)=0$.

解 取终止准则值为 $\|\nabla f(x_k)\|\le 10^{-5}$, 利用程序 5.1, 取不同的初始点, 数值结果如下表.

表 5.1 对称秩 1 校正算法的数值结果.

初始点 (x_0)	迭代次数 (k)	目标函数值 ($f(x_k)$)
$(0,0)^T$	22	7.0304×10^{-19}
$(0.5,0.5)^T$	19	3.8208×10^{-16}
$(2,2)^T$	38	3.3992×10^{-20}
$(-1,-1)^T$	45	8.2927×10^{-16}
$(1,10)^T$	98	1.9321×10^{-16}
$(10,10)^T$	142	2.1578×10^{-15}

说明 上述程序的调用方式是:

```
x0=[-1.2 1]';
[x,val,k]=sr1('fun','gfun',x0)
```

其中 fun, gfun 分别是求目标函数值及其梯度的 M 函数文件.

5.2　BFGS 算法及其 Matlab 实现

BFGS 校正是目前最流行也是最有效的拟牛顿校正, 它是由 Broyden, Fletcher, Goldfarb 和 Shanno 在 1970 年各自独立提出的拟牛顿法, 故称为 BFGS 算法. 其基本思想是: 在 (5.3) 中取修正矩阵 E_k 为秩 2 矩阵:

$$E_k = \alpha u_k u_k^T + \beta v_k v_k^T,$$

其中 $u_k, v_k \in \mathbb{R}^n$ 是待定向量, $\alpha, \beta \in \mathbb{R}$ 是待定实数. 于是由拟牛顿方程 (5.1) 可得

$$(B_k + \alpha u_k u_k^T + \beta v_k v_k^T)s_k = y_k,$$

或等价地

$$\alpha(u_k^T s_k)u_k + \beta(v_k^T s_k)v_k = y_k - B_k s_k. \tag{5.7}$$

不难发现, 满足上式的向量 u_k 和 v_k 不唯一, 可取 u_k 和 v_k 分别平行于 $B_k s_k$ 和 y_k, 即令 $u_k = \gamma B_k s_k$, $v_k = \theta y_k$, 其中 γ, θ 是待定的参数. 于是我们有

$$E_k = \alpha\gamma^2 B_k s_k s_k^T B_k + \beta\theta^2 y_k y_k^T.$$

将 u_k, v_k 的表达式代入 (5.7) 得

$$\alpha[(\gamma B_k s_k)^T s_k](\gamma B_k s_k) + \beta[(\theta y_k)^T s_k)(\theta y_k) = y_k - B_k s_k,$$

整理得

$$[\alpha\gamma^2(s_k^T B_k s_k) + 1]B_k s_k + [\beta\theta^2(y_k^T s) - 1]y_k = 0.$$

故此, 可令 $\alpha\gamma^2(s_k^T B_k s_k) + 1 = 0$ 及 $\beta\theta^2(y_k^T s) - 1 = 0$, 即

$$\alpha\gamma^2 = -\frac{1}{s_k^T B_k s_k}, \quad \beta\theta^2 = \frac{1}{y_k^T s_k}.$$

从而得到如下的 BFGS 秩 2 修正公式如下

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}. \tag{5.8}$$

显然, 由 (5.8) 可知, 若 B_k 对称, 校正后的 B_{k+1} 也对称, 并且可以证明 BFGS 校正公式的如下性质:

引理 5.1 设 B_k 对称正定, B_{k+1} 由 BFGS 校正公式 (5.8) 确定, 那么 B_{k+1} 对称正定的充要条件是 $y_k^T s_k > 0$.

证 必要性是显然的. 因 $y_k^T s_k = s_k^T B_{k+1} s_k$, 故若 B_{k+1} 正定, 则显然有 $y_k^T s_k > 0$.

下面证明充分性. 设 $y_k^T s_k > 0$ 且 B_k 正定. 由校正公式 (5.8), 对任意的 $0 \neq d \in \mathbb{R}^n$, 我们有

$$d^T B_{k+1} d = d^T B_k d - \frac{(d^T B_k s)^2}{s_k^T B_k s_k} + \frac{(d^T y_k)^2}{y_k^T s_k}. \tag{5.9}$$

因 B_k 对称正定, 故存在对称正定阵 $B_k^{1/2}$, 使得 $B_k = B_k^{1/2} B_k^{1/2}$. 从而, 利用 Cauchy-Schwarz 不等式得

$$\begin{aligned}(d^T B_k s_k)^2 &= [(B_k^{1/2} d)^T (B_k^{1/2} s_k)]^2 \le \|B_k^{1/2} d\|^2 \|B_k^{1/2} s_k\|^2 \\ &= (B_k^{1/2} d)^T (B_k^{1/2} d) \cdot (B_k^{1/2} s_k)^T (B_k^{1/2} s_k) \\ &= (d^T B_k d)(s_k^T B_k s_k). \end{aligned} \tag{5.10}$$

不难发现, 上式成立等式的充要条件是存在实数 $\tau_k \neq 0$, 使得 $B_k^{1/2} d = \tau_k B_k^{1/2} s_k$, 即 $d = \tau_k s_k$.

故而, 若不等式 (5.10) 严格成立, 则由 (5.9) 得

$$d^T B_{k+1} d > d^T B_k d - \frac{(d^T B_k d)(s_k^T B_k s_k)}{s_k^T B_k s_k} + \frac{(d^T y_k)^2}{y_k^T s_k} > 0.$$

否则, 若 (5.10) 等式成立, 即存在 τ_k, 使得 $d = \tau_k s_k$, 则由 (5.9), (5.10) 得

$$d^T B_{k+1} d = \frac{(d^T y_k)^2}{y_k^T s_k} = \frac{\tau_k^2 (s_k^T y_k)^2}{y_k^T s_k} = \tau_k^2 y_k^T s_k > 0.$$

故对任意的 $d \in \mathbb{R}^n$, $d \neq 0$, 总有 $d^T B_{k+1} d > 0$. 证毕. □

上面的引理表明, 若初始矩阵 B_0 对称正定且在迭代过程中保持 $y_k^T s_k > 0$, $\forall\, k \geq 0$, 则由 BFGS 校正公式产生的矩阵序列 $\{B_k\}$ 是对称正定的. 从而方程组 $B_k d = -g_k$ 有唯一解 d_k, 且 d_k 是函数 f 在 x_k 处的下降方向.

引理 5.2 若在 BFGS 算法中采用精确线搜索或 Wolfe 搜索准则, 则有 $y_k^T s_k > 0$.

证 注意到对于精确线搜索有 $g_{k+1}^T d_k = 0$. 故

$$y_k^T s_k = \alpha_k (g_{k+1} - g_k)^T d_k = -\alpha_k g_k^T d_k > 0.$$

对于 Wolfe 搜索准则, 利用该准则的第二个不等式 (即 $\nabla f(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k$), 得

$$\begin{aligned} y_k^T s_k &= \alpha_k (g_{k+1} - g_k)^T d_k \geq \alpha_k (\sigma - 1) g_k^T d_k \\ &= -\alpha_k (1 - \sigma) g_k^T d_k > 0. \end{aligned}$$

证毕. □

已有证明表示, Armijo 搜索准则一般不能保证 $y_k^T s_k > 0$. 但 Armijo 准则因其简单且易于程序实现而深得人们的喜爱, 因此, 为了保证采用 Armijo 准则时矩阵序列 $\{B_k\}$ 的对称正定性, 可采用如下的校正方式

$$B_{k+1} = \begin{cases} B_k, & \text{若 } y_k^T s_k \le 0, \\ B_k - \dfrac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \dfrac{y_k y_k^T}{y_k^T s_k}, & \text{若 } y_k^T s_k > 0. \end{cases} \tag{5.11}$$

不难发现, 只要 B_0 对称正定, 上述校正公式可以保证矩阵序列 $\{B_k\}$ 的对称正定性. 下面给出基于 Armijo 搜索准则的 BFGS 算法的详细步骤.

算法 5.2 (BFGS 算法)

步 0 给定参数 $\delta \in (0,1)$, $\sigma \in (0, 0.5)$, 初始点 $x_0 \in \mathbb{R}^n$, 终止误差 $0 \le \varepsilon \ll 1$. 初始对称正定阵 B_0 (通常取为 $G(x_0)$ 或单位阵 I_n). 令 $k := 0$.

步 1 计算 $g_k = \nabla f(x_k)$. 若 $\|g_k\| \le \varepsilon$, 停算, 输出 x_k 作为近似极小点.

步 2 解线性方程组得解 d_k:

$$B_k d = -g_k. \tag{5.12}$$

步 3 设 m_k 是满足下列不等式的最小非负整数 m:

$$f(x_k + \delta^m d_k) \le f(x_k) + \sigma\delta^m g_k^T d_k. \tag{5.13}$$

令 $\alpha_k = \delta^{m_k}$, $x_{k+1} = x_k + \alpha_k d_k$.

步 4 由校正公式 (5.11) 确定 B_{k+1}.

步 5 令 $k := k+1$, 转步 1.

下面给出基于 Armijo 搜索的 BFGS 算法的 Matlab 程序.

程序 5.2 (BFGS 算法程序)

```
function [x,val,k]=bfgs(fun,gfun,x0,varargin)
%功能: 用BFGS算法求解无约束问题:  min f(x)
%输入: x0是初始点, fun, gfun分别是目标函数及其梯度;
% varargin是输入的可变参数变量, 简单调用bfgs时可以忽略它,
```

```
% 但若其它程序循环调用该程序时将发挥重要的作用
%输出： x, val分别是近似最优点和最优值, k是迭代次数.
maxk=500;   %给出最大迭代次数
rho=0.55;sigma=0.4; epsilon=1e-5;
k=0;   n=length(x0);
Bk=eye(n);  %Bk=feval('Hess',x0);
while(k<maxk)
  gk=feval(gfun,x0,varargin{:}); %计算梯度
  if(norm(gk)<epsilon), break; end  %检验终止准则
  dk=-Bk\gk;  %解方程组, 计算搜索方向
  m=0; mk=0;
  while(m<20)   % 用Armijo搜索求步长
    newf=feval(fun,x0+rho^m*dk,varargin{:});
    oldf=feval(fun,x0,varargin{:});
    if(newf<oldf+sigma*rho^m*gk'*dk)
```

```
        mk=m; break;
      end
      m=m+1;
    end
    %BFGS校正
    x=x0+rho^mk*dk;
    sk=x-x0;  yk=feval(gfun,x,varargin{:})-gk;
    if(yk'*sk>0)
      Bk=Bk-(Bk*sk*sk'*Bk)/(sk'*Bk*sk)+(yk*yk')/(yk'*sk);
    end
    k=k+1;   x0=x;
  end
  val=feval(fun,x0,varargin{:});
```

例 5.2 利用程序 5.2 求解无约束优化问题

$$\min\ f(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2,\ \ x = (x_1, x_2)^T \in \mathbb{R}^2.$$

该问题有精确解 $x^* = (1,1)^T$, $f(x^*) = 0$.

解　取终止准则值为 $\|\nabla f(x_k)\| \le 10^{-5}$, 利用程序 5.2, 取不同的初始点, 数值结果如下表.

表 5.2 BFGS 校正算法的数值结果.

初始点 (x_0)	迭代次数 (k)	目标函数值 ($f(x_k)$)
$(0,0)^T$	20	2.2005×10^{-11}
$(0.5,0.5)^T$	15	1.946×10^{-16}
$(2,2)^T$	24	2.1171×10^{-15}
$(-1,-1)^T$	31	1.3594×10^{-12}
$(1,10)^T$	36	1.3757×10^{-15}
$(10,10)^T$	66	6.3531×10^{-15}
$(-1.2,1)^T$	32	6.7539×10^{-16}

从上表可以看出, BFGS 算法比对称秩 1 算法更为有效.

5.3 DFP 算法及其 Matlab 实现

DFP 校正是第一个拟牛顿校正, 是 1959 年由 Davidon 提出的, 后经 Fletcher 和 Powell 解释和改进, 故名之为 DFP 算法, 它也是求解无约束优化问题最有效的算法之一. 类似于 BFGS 校正公式的推导, 可得 DFP 校正公式如下:

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{s_k^T y_k}. \tag{5.14}$$

同样, 不难发现, 由 (5.14), 若 H_k 对称, 校正后的 H_{k+1} 也对称, 并且类似于引理 5.1 的证明, 可得 DFP 校正公式的如下性质:

引理 5.3 设 H_k 对称正定, H_{k+1} 由 DFP 校正公式 (5.14) 确定, 那么 H_{k+1} 对称正定的充要条件是 $s_k^T y_k > 0$.

类似于引理 5.2, 可以证明, 当采用精确线搜索或 Wolfe 搜索准则时, 矩阵序列 $\{H_k\}$ 的正定性条件 $s_k^T y_k > 0$ 可以被满足. 但一般来说, Armijo

搜索准备不能满足这一条件, 需要作如下修正:

$$H_{k+1}=\begin{cases}H_k, & \text{若 } s_k^T y_k \le 0,\\ H_k-\dfrac{H_k y_k y_k^T H_k}{y_k^T H_k y_k}+\dfrac{s_k s_k^T}{s_k^T y_k}, & \text{若 } y_k^T s_k > 0.\end{cases} \tag{5.15}$$

下面给出基于 Armijo 搜索准则的 DFP 算法的详细步骤.

算法 5.3 (**DFP 算法**)

步 0 给定参数 $\delta \in (0,1)$, $\sigma \in (0,0.5)$, 初始点 $x_0 \in \mathbb{R}^n$, 终止误差 $0 \le \varepsilon \ll 1$. 初始对称正定阵 H_0 (通常取为 $G(x_0)^{-1}$ 或单位阵 I_n). 令 $k := 0$.

步 1 计算 $g_k = \nabla f(x_k)$. 若 $\|g_k\| \le \varepsilon$, 停算, 输出 x_k 作为近似极小点.

步 2 计算搜索方向:

$$d_k = -H_k g_k. \tag{5.16}$$

步 3　设 m_k 是满足下列不等式的最小非负整数 m:

$$f(x_k + \delta^m d_k) \le f(x_k) + \sigma \delta^m g_k^T d_k. \tag{5.17}$$

令 $\alpha_k = \delta^{m_k}$, $x_{k+1} = x_k + \alpha_k d_k$.

步 4　由校正公式 (5.15) 确定 H_{k+1}.

步 5　令 $k := k + 1$, 转步 1.

下面给出基于 Armijo 搜索的 DFP 算法的 Matlab 程序.

程序 5.3 (**DFP 算法程序**)

```
function [x,val,k]=dfp(fun,gfun,x0)
%功能: 用DFP算法求解无约束问题: min f(x)
%输入: x0是初始点, fun, gfun分别是目标函数及其梯度
%输出: x, val分别是近似最优点和最优值, k是迭代次数.
maxk=1e5;  %给出最大迭代次数
rho=0.55;sigma=0.4; epsilon=1e-5;
```

```
k=0;   n=length(x0);
Hk=inv(feval('Hess',x0));   %Hk=eye(n);
while(k<maxk)
  gk=feval(gfun,x0); %计算梯度
  if(norm(gk)<epsilon), break; end  %检验终止准则
  dk=-Hk*gk;  %解方程组，计算搜索方向
  m=0; mk=0;
  while(m<20)   % 用Armijo搜索求步长
  if(feval(fun,x0+rho^m*dk)<feval(fun,x0)+sigma*rho^m*gk'*dk)
    mk=m; break;
  end
  m=m+1;
  end
  %DFP校正
  x=x0+rho^mk*dk;
```

```
    sk=x-x0;  yk=feval(gfun,x)-gk;
    if(sk'*yk>0)
      Hk=Hk-(Hk*yk*yk'*Hk)/(yk'*Hk*yk)+(sk*sk')/(sk'*yk);
    end
    k=k+1;     x0=x;
  end
  val=feval(fun,x0);
```

例 5.3 利用程序 5.3 求解无约束优化问题

$$\min\ f(x)=100(x_1^2-x_2)^2+(x_1-1)^2,\ \ x=(x_1,x_2)^T\in\mathbb{R}^2.$$

该问题有精确解 $x^*=(1,1)^T$, $f(x^*)=0$.

解 取终止准则值为 $\|\nabla f(x_k)\|\leq 10^{-5}$, 利用程序 5.3, 取不同的初始点, 数值结果如下表.

表 5.3 DFP 校正算法的数值结果.

初始点 (x_0)	迭代次数 (k)	目标函数值 $(f(x_k))$
$(0,0)^T$	23	9.4910×10^{-16}
$(0.5,0.5)^T$	19	1.5488×10^{-15}
$(2,2)^T$	22	4.0247×10^{-13}
$(-1,-1)^T$	35	2.2338×10^{-12}
$(1,10)^T$	1	0
$(10,10)^T$	77	8.6152×10^{-20}
$(-1.2,1)^T$	34	3.0415×10^{-14}

从上表可以看出, DFP 算法的计算效率似乎不如 BFGS 算法.

5.4　Broyden 族算法及其 Matlab 实现

前面我们讨论了 BFGS 和 DFP 校正, 它们都是由 y_k 和 $B_k s_k$(或 s_k 和 $H_k y_k$) 组成的秩 2 校正. 本节讨论由 BFGS 和 DFP 校正的凸组合产生

的一类校正族

$$
\begin{aligned}
B_{k+1}^{\theta} &= \theta_k B_{k+1}^{\mathrm{DFP}} + (1-\theta_k) B_{k+1}^{\mathrm{BFGS}} && (5.18) \\
&= B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k} + \theta_k u_k u_k^T, && (5.19)
\end{aligned}
$$

其中 θ_k 为实参数, u_k 由下式定义:

$$
u_k = \sqrt{s_k^T B_k s_k}\left(\frac{y_k}{y_k^T s_k} - \frac{B_k s_k}{s^T B_k s_k}\right). \tag{5.20}
$$

这类校正公式称为 Broyden 族. 不难发现, 在 (5.19) 中, 当 $\theta_k = 0$, 即得到 BFGS 公式; 当 $\theta_k = 1$, 即得到 DFP 公式.

对应地, 关于 H_k 的 Broyden 族校正公式为

$$
\begin{aligned}
H_{k+1}^{\phi} &= \phi_k H_{k+1}^{\mathrm{BFGS}} + (1-\phi_k) H_{k+1}^{\mathrm{DFP}} && (5.21) \\
&= H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{s_k^T y_k} + \phi_k v_k v_k^T, && (5.22)
\end{aligned}
$$

其中 ϕ_k 为实参数, v_k 由下式定义:

$$v_k = \sqrt{y_k^T H_k y_k}\left(\frac{s_k}{y_k^T s_k} - \frac{H_k y_k}{y_k^T H_k y_k}\right). \tag{5.23}$$

可以证明参数 θ_k 与 ϕ_k 之间的关系为

$$\theta_k = \frac{1-\phi_k}{1-\phi_k(1-\mu_k)}, \tag{5.24}$$

其中

$$\mu_k = \frac{(s_k^T B_k s_k)(y_k^T H_k y_k)}{(s_k^T y_k)^2}.$$

不难发现 $u_k^T s_k = 0$ 和 $v_k^T y_k = 0$, 因此 Broyden 族 (5.19) 和 (5.22) 给出的校正公式于任何参数 θ_k 和 ϕ_k 都满足拟牛顿方程 (5.1) 和 (5.2).

下面我们给出 Broyden 族校正公式的两个性质.

定理 5.1 用 Broyden 族算法求解极小化二次目标函数问题

$$\min f(x) = \frac{1}{2}x^T G x + b^T x + c. \tag{5.25}$$

若果初始矩阵 H_0 是正定的, 算法所产生的迭代点列是互异的. 则

(1) 当 $y_k^T s_k > 0$ 且 $\phi_k \geq 0$ 或 $\theta_k \geq 0$ 时, Broyden 族校正公式保持正定性.

(2) 算法所产生的搜索方向 $d_0, d_1, \cdots, d_k (k \leq n-1)$ 满足

(a) $d_i^T G d_j = 0,\ 0 \leq i < j \leq k$;　　(b) $H_k^{\phi} y_i = s_i,\ 0 \leq i \leq k-1$.

证 对 k 用归纳法. 注意到 $s_i = x_{i+1} - x_i = \alpha_i d_i$, $y_i = g_{i+1} - g_i = G(x_{i+1} - x_i) = G s_i$. 那么, 当 $k = 1$ 时, 由拟牛顿方程 (5.2) 可知

$$H_1^{\phi} y_0 = s_0$$

成立, 且有

$$\begin{aligned} d_0^T G d_1 &= (G d_0)^T d_1 = -\frac{1}{\alpha_0}(G s_0)^T H_1^{\phi} g_1 \\ &= -\frac{1}{\alpha_0} y_0^T H_1^{\phi} g_1 = -\frac{1}{\alpha_0} s_0^T g_1 \\ &= -d_0^T g_1 = 0, \end{aligned}$$

即 $k = 1$ 时结论成立.

设 $k=l$ 时结论成立, 要证 $k=l+1$ 时结论也成立. 由归纳法假设有

$$d_i^TGd_j=0,\ \ 0\le i<j\le l;\quad H_l^{\phi}y_i=s_i,\ \ 0\le i\le l-1. \tag{5.26}$$

当 $k=l+1$ 时, 对于 $0\le i\le l-1$, 我们有

$$\begin{aligned}
H_{l+1}^{\phi}y_i &= \left(H_l^{\phi}-\frac{H_l^{\phi}y_ly_l^TH_l^{\phi}}{y_l^TH_l^{\phi}y_l}+\frac{s_ls_l^T}{s_l^Ty_l}+\phi v_lv_l^T\right)y_i\\
&= H_l^{\phi}y_i-\frac{H_l^{\phi}y_ly_l^TH_l^{\phi}y_i}{y_l^TH_l^{\phi}y_l}+\frac{s_ls_l^Ty_i}{s_l^Ty_l}+\phi v_lv_l^Ty_i\\
&= s_i-\frac{H_l^{\phi}y_l(y_l^Ts_i)}{y_l^TH_l^{\phi}y_l}+\frac{s_ls_l^Ty_i}{s_l^Ty_l}+\phi v_lv_l^Ty_i\\
&= s_i-\frac{H_l^{\phi}y_l(s_l^TGs_i)}{y_l^TH_l^{\phi}y_l}+\frac{s_ls_l^TGs_i}{s_l^Ty_l}+\phi v_lv_l^Ty_i
\end{aligned}$$

由 $s_i = \alpha_i d_i$ 和 (5.26) 的第一式可知, 上式等式右边的第二、三两项为 0. 下面考虑第三项. 注意到

$$
\begin{aligned}
&\frac{\phi}{(y_l^T H_l^\phi y_l)} v_l v_l^T y_i = \left(\frac{s_l}{y_l^T s_l} - \frac{H_l^\phi y_l}{y_l^T H_l^\phi y_l}\right)\left(\frac{s_l}{y_l^T s_l} - \frac{H_l^\phi y_l}{y_l^T H_l^\phi y_l}\right)^T y_i \\
&= \frac{s_l s_l^T y_i}{(y_l^T s_l)^2} + \frac{H_l^\phi y_l y_l^T H_l^\phi y_i}{(y_l^T H_l^\phi y_l)^2} - \frac{s_l y_l^T H_l^\phi y_i}{(y_l^T s_l)(y_l^T H_l^\phi y_l)} - \frac{H_l^\phi y_l s_l^T y_i}{(y_l^T s_l)(y_l^T H_l^\phi y_l)} \\
&= \frac{s_l (s_l^T G s_i)}{(y_l^T s_l)^2} + \frac{H_l^\phi y_l y_l^T s_i}{(y_l^T H_l^\phi y_l)^2} - \frac{s_l y_l^T s_i}{(y_l^T s_l)(y_l^T H_l^\phi y_l)} - \frac{H_l^\phi y_l (s_l^T G s_i)}{(y_l^T s_l)(y_l^T H_l^\phi y_l)} \\
&= 0 + \frac{H_l^\phi y_l (s_l^T G s_i)}{(y_l^T H_l^\phi y_l)^2} - \frac{s_l (s_l^T G s_i)}{(y_l^T s_l)(y_l^T H_l^\phi y_l)} - 0 = 0.
\end{aligned}
$$

故有

$$
H_{l+1}^\phi y_i = s_i, \ \ 0 \le i \le l-1. \tag{5.27}
$$

又因为拟牛顿方程 $H_{l+1}^\phi y_l = s_l$ 满足, 所以 (5.27) 对于 $0 \le i \le l$ 成立.

下面证明

$$d_i^T G d_j = 0,\ \ 0 \le i < j \le l+1. \tag{5.28}$$

由归纳法假设 (5.26), 只需证明对于 $0 \le i \le l$ 成立

$$d_i G d_{l+1} = 0.$$

事实上,

$$\begin{aligned} d_i G d_{l+1} &= (Gd_i)^T d_{l+1} = \frac{1}{\alpha_i} y_i^T \big(- H_{l+1}^{\phi} g_{l+1} \big) \\ &= -\frac{1}{\alpha_i} s_i^T g_{l+1} = -d_i^T g_{l+1} \\ &= -d_i^T \Big(g_{i+1} + \sum_{j=i+1}^{l} (g_{j+1} - g_j) \Big) \\ &= -g_{i+1}^T d_i + \sum_{j=i+1}^{l} y_j^T d_i \\ &= -g_{i+1}^T d_i + \sum_{j=i+1}^{l} s_j^T G d_i = 0. \end{aligned}$$

因此, 对于 $k=l+1$, 结论成立. □

推论 5.1 在定理 5.1 的条件下, 有如下结论:

(1) Broyden 族校正公式至多迭代 n 次就可以达到极小点 x^*, 即存在 $k\,(0\le k\le n)$, 使得 $x_k=x^*$.

(2) 若 $x_k\ne x^*$, $0\le k\le n-1$, 则 $H_n=G^{-1}$.

证 由定理 5.1 可知, Broyden 族校正算法是一种共轭方向法, 故结论 (1) 成立.

若 $x_k\ne x^*$, $0\le k\le n-1$, 则 Broyden 族校正公式产生 n 个共轭方向 $d_0,d_1,\cdots,d_{n-1}$, 因而它们是线性无关的, 从而 $s_0,s_1,\cdots,s_{n-1}$ 也是线性无关的. 又由定理 5.1 得

$$H_nGs_i=H_ny_i=s_i,\ \ i=0,1,\cdots,n-1,$$

即

$$H_nG[s_0,s_1,\cdots,s_{n-1}]=[s_0,s_1,\cdots,s_{n-1}].$$

因矩阵 $[s_0, s_1, \cdots, s_{n-1}]$ 非奇异, 故有 $H_n G = I$, 即 $H_n = G^{-1}$. 证毕. □

下面给出基于 Armijo 搜索准则的 Broyden 族算法的详细步骤.

算法 5.4 (**Broyden 族算法**)

步 0 给定参数 $\delta \in (0,1)$, $\sigma \in (0, 0.5)$,$\phi \in [0,1]$. 初始点 $x_0 \in \mathbb{R}^n$, 终止误差 $0 \le \varepsilon \ll 1$. 初始对称正定阵 H_0 (通常取为 $G(x_0)^{-1}$ 或单位阵 I_n). 令 $k := 0$.

步 1 计算 $g_k = \nabla f(x_k)$. 若 $\|g_k\| \le \varepsilon$, 停算, 输出 x_k 作为近似极小点.

步 2 计算搜索方向:

$$d_k = -H_k g_k. \tag{5.29}$$

步 3 设 m_k 是满足下列不等式的最小非负整数 m:

$$f(x_k + \delta^m d_k) \le f(x_k) + \sigma \delta^m g_k^T d_k. \tag{5.30}$$

令 $\alpha_k = \delta^{m_k}$, $x_{k+1} = x_k + \alpha_k d_k$.

步 4 由下面的校正公式确定 H_{k+1}：

$$H_{k+1}=\begin{cases} H_k, & \text{若 } s_k^T y_k \le 0, \\ H_k - \dfrac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \dfrac{s_k s_k^T}{s_k^T y_k} + \phi v_k v_k^T, & \text{若 } y_k^T s_k > 0. \end{cases} \tag{5.31}$$

其中, v_k 由下式定义:

$$v_k = \sqrt{y_k^T H_k y_k}\left(\frac{s_k}{y_k^T s_k} - \frac{H_k y_k}{y_k^T H_k y_k}\right).$$

步 5 令 $k := k+1$, 转步 1.

下面给出基于 Armijo 搜索的 Broyden 族算法的 Matlab 程序.

程序 5.4 (**Broyden 族算法程序**)

```
function [x,val,k]=broyden(fun,gfun,x0)
%功能: 用Broyden族算法求解无约束问题:  min f(x)
%输入: x0是初始点, fun, gfun分别是目标函数及其梯度
```

```
%输出: x,val分别是近似最优点和最优值,  k是迭代次数.
maxk=1e5;   %给出最大迭代次数
rho=0.55;sigma=0.4; epsilon=1e-5;
phi=0.5; k=0;   n=length(x0);
Hk=inv(feval('Hess',x0));   %Hk=eye(n);
while(k<maxk)
  gk=feval(gfun,x0); %计算梯度
  if(norm(gk)<epsilon), break; end  %检验终止准则
  dk=-Hk*gk;  %解方程组, 计算搜索方向
  m=0; mk=0;
  while(m<20)   % 用Armijo搜索求步长
  if(feval(fun,x0+rho^m*dk)<feval(fun,x0)+sigma*rho^m*gk'*dk)
    mk=m; break;
  end
  m=m+1;
```

```
    end
    %Broyden族校正
    x=x0+rho^mk*dk;
    sk=x-x0;  yk=feval(gfun,x)-gk;
    Hy=Hk*yk; sy=sk'*yk;  yHy=yk'*Hk*yk;
    if(sy<0.2*yHy)
       theta=0.8*yHy/(yHy-sy);
       sk=theta*sk+(1-theta)*Hy;
       sy=0.2*yHy;
    end
    vk=sqrt(yHy)*(sk/sy - Hy/yHy);
    Hk=Hk-(Hy*Hy')/yHy+(sk*sk')/sy+phi*vk*vk';
    k=k+1;  x0=x;
  end
  val=feval(fun,x0);
```

例 5.4 利用程序 5.4 求解无约束优化问题

$$\min\ f(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2,\ \ x = (x_1, x_2)^T \in \mathbb{R}^2.$$

该问题有精确解 $x^* = (1,1)^T$, $f(x^*) = 0$.

解 取终止准则值为 $\|\nabla f(x_k)\| \le 10^{-5}$, 利用程序 5.4, 取不同的初始点, 数值结果如下表.

表 5.4 Broyden 族校正算法的数值结果.

初始点 (x_0)	迭代次数 (k)	目标函数值 $(f(x_k))$
$(0,0)^T$	23	9.4910×10^{-16}
$(0.5,0.5)^T$	19	1.5488×10^{-15}
$(2,2)^T$	22	4.0247×10^{-13}
$(-1,-1)^T$	35	2.2338×10^{-12}
$(1,10)^T$	1	0
$(10,10)^T$	77	8.6152×10^{-20}
$(-1.2,1)^T$	34	3.0415×10^{-14}

5.5 拟牛顿法的收敛性

本节讨论拟牛顿法的收敛性, 主要给出基于非精确线搜索的 BFGS 算法的全局收敛性和局部超线性收敛性定理. 为了方便, 我们将 BFGS 算法的迭代公式复述如下:

$$x_{k+1} = x_k - \alpha_k B_k^{-1} g_k, \tag{5.32}$$

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}, \tag{5.33}$$

其中, $s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$, α_k 由非精确线搜索方法得到. 在讨论收敛性之前, 我们先给出如下假设条件.

假设 5.1 (1) 函数 $f(x)$ 在 $\mathbb{R}^n$ 上二阶连续可微.

(2) 水平集

$$\mathcal{L}(x_0) = \{x \in \mathbb{R}^n \mid f(x) \le f(x_0)\}$$

是凸集, 且函数 f 在 $\mathcal{L}(x_0)$ 上一致凸, 即存在常数 $0 < m \le M$, 使得

$$m\|d\|^2 \le d^T G(x) d \le M\|d\|^2. \tag{5.34}$$

(3) 存在 x^* 的一个邻域 $N(x^*, \delta)$, 使得 $G(x) = \nabla^2 f(x)$ 在该邻域内 Lipschitz 连续, 即存在常数 $L > 0$, 使得

$$\|G(x) - G(x^*)\| \le L\|x - x^*\|, \quad \forall\, x \in N(x^*, \delta).$$

在后面的分析中我们需要用到一个求秩 2 校正矩阵行列式的公式, 先把它来写在下面以方便使用:

$$\det\big(I + w_1 w_2^T + w_3 w_4^T\big) = (1 + w_1^T w_2)(1 + w_3^T w_4) - (w_1^T w_4)(w_2^T w_3), \tag{5.35}$$

这里 $w_i\,(i = 1, \cdots, 4)$ 是任意的 n 维向量.

我们有下面的全局收敛性定理.

定理 5.2 设 $\{B_k\}$ 是由 BFGS 校正公式 (5.33) 产生的非奇异矩阵序列, α_k 为满足 Armijo 准则 (2.13) 的步长. 若 $f(x)$ 满足假设 5.1 (1) 和 (2). 那么由迭代公式 (5.32) 产生的序列 $\{x_k\}$ 全局收敛到 $f(x)$ 的极小点 x^*.

证 根据 Armijo 准则下的线搜索法的收敛定理 2.2, 我们只需验证搜索方向 d_k 与负梯度方向 $-g_k$ 的夹角 θ_k 满足条件 (2.15), 即 $0 < \theta_k \le \dfrac{\pi}{2} - \mu$, $\mu \in \left(0, \dfrac{\pi}{2}\right)$. 注意到

$$\cos\theta_k = \frac{-d_k^T g_k}{\|d_k\|\|g_k\|} = \frac{s_k^T(B_k s_k)}{\|s_k\|\|B_k s_k\|}.$$

以下只需证明由上式定义的 θ_k 满足 $\cos\theta_k \ge \delta > 0$ 即可.

由 y_k 的定义，可得

$$y_k = g_{k+1} - g_k = \int_0^1 G(x_k + \tau s_k) s_k \mathrm{d}\tau, \tag{5.36}$$

故有

$$y_k^T s_k = \int_0^1 s_k^T G(x_k + \tau s_k) s_k \mathrm{d}\tau.$$

利用假设 5.1 (2) 可得

$$y_k^T s_k \ge m\|s_k\|^2, \quad 即 \quad a_k \triangleq \frac{y_k^T s_k}{\|s_k\|^2} \ge m. \tag{5.37}$$

由 (5.36) 得

$$\|y_k\| \leq \int_0^1 \|G(x_k+\tau s_k)s_k\|\mathrm{d}\tau \leq \|s_k\| \int_0^1 \|G(x_k+\tau s_k)\|\mathrm{d}\tau. \tag{5.38}$$

不难发现, 由假设 5.1 (2) 有

$$\max_{x\in\mathcal{L}(x_0)} \|G(x)\| = \max_{x\in\mathcal{L}(x_0)} \sup_{d\neq 0} \frac{|d^T G(x) d|}{\|d\|^2} \leq M.$$

因此由 (5.38) 可推得 $\|y_k\| \leq M\|s_k\|$. 结合 (5.37), 我们有

$$b_k \triangleq \frac{\|y_k\|^2}{y_k^T s_k} \leq \frac{\|y_k\|^2}{m\|s_k\|^2} \leq \frac{M^2}{m} \triangleq \bar{b}.$$

对 BFGS 公式

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}$$

两边求迹得

$$\mathrm{tr}(B_{k+1}) = \mathrm{tr}(B_k) - \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + \frac{\|y_k\|^2}{y_k^T s_k}. \tag{5.39}$$

为了便于应用公式 (5.35), 我们将 BFGS 校正公式写成如下形式:

$$\begin{aligned} B_{k+1} &= B_k\left(I + \left[-\frac{1}{s_k^T B_k s_k} s_k\right](s_k^T B_k) + \left[\frac{1}{s_k^T y_k} B_k^{-1} y_k\right] y_k^T\right) \\ &\triangleq B_k(I + w_1 w_2^T + w_3 w_4^T). \end{aligned}$$

利用公式 (5.35) 得

$$\det\left(B_{k+1}\right) = -\det\left(B_k\right)\left(w_1^T w_4\right) = \det\left(B_k\right)\frac{s_k^T y_k}{s_k^T B_k s_k}. \tag{5.40}$$

记

$$c_k \triangleq \frac{s_k^T B_k s_k}{\|s_k\|^2},$$

则

$$\frac{\|B_k s_k\|^2}{s_k^T B_k s_k} = \left(\frac{\|s_k\|\|B_k s_k\|}{s_k^T B_k s_k}\right)^2 \frac{s_k^T B_k s_k}{\|s_k\|^2} = \frac{c_k}{\cos^2\theta_k}. \tag{5.41}$$

于是由 (5.40) 有

$$\det\big(B_{k+1}\big) = \det\big(B_k\big)\frac{s_k^T y_k}{\|s_k\|^2}\frac{\|s_k\|^2}{s_k^T B_k s_k} = \det\big(B_k\big)\frac{a_k}{c_k}. \tag{5.42}$$

关于对称正定矩阵 B, 定义函数

$$\phi(B) = \mathrm{tr}(B) - \ln(\det(B)),$$

则有 $\phi(B) > 0$. 事实上, 设 $0 < \lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$ 为 B 的特征值, 则

$$\begin{aligned}\phi(B) &= \mathrm{tr}(B) - \ln(\det(B)) \\ &= \sum_{i=1}^{n}\lambda_i - \ln(\lambda_1\lambda_2\cdots\lambda_n) \\ &= \sum_{i=1}^{n}(\lambda_i - \ln\lambda_i) > 0.\end{aligned}$$

由 (5.39), (5.41) 和 (5.42) 得

$$
\begin{aligned}
\phi(B_{k+1}) &= \operatorname{tr}(B_{k+1}) - \ln(\det(B_{k+1})) \\
&= \operatorname{tr}(B_k) - \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + \frac{\|y_k\|^2}{y_k^T s_k} - \ln\left(\det(B_k)\frac{a_k}{c_k}\right) \\
&= \phi(B_k) - \frac{c_k}{\cos^2\theta_k} + b_k - \ln a_k + \ln c_k \\
&= \phi(B_k) + (b_k - \ln a_k - 1) \\
&\quad + \left(1 - \frac{c_k}{\cos^2\theta_k} + \ln\frac{c_k}{\cos^2\theta_k}\right) + \ln\cos^2\theta_k \\
&\le \phi(B_k) + \sigma + \ln\cos^2\theta_k,
\end{aligned}
$$

上式的最后一个不等式利用了函数 $\psi(t) = 1 - t + \ln t$ 在区间 $(0,\infty)$ 上的非正性及 $a_k \ge m$ 和 $b_k \le \bar{b}$, 且正常数 $\sigma \ge \bar{b} - \ln m - 1 \ge b_k - \ln a_k - 1$. 于是有

$$
0 < \phi(B_{k+1}) \le \phi(B_0) + \sigma(k+1) + \sum_{i=0}^{k} \ln\cos^2\theta_i. \tag{5.43}
$$

下面证明数列 $\{\cos\theta_k\} \not\to 0\,(k\to\infty)$. 用反证法. 若结论不成立, 则对上述的常数 $l>0$, 存在 $k_0>0$ 使得对所有的 $k\geq k_0$, 有

$$\ln\cos^2\theta_k < -2\sigma.$$

由 (5.43) 得

$$\begin{aligned}
0 &< \phi(B_{k+1}) \\
&\leq \phi(B_0)+\sigma(k+1)+\sum_{i=0}^{k_0-1}\ln\cos^2\theta_i+\sum_{i=k_0}^{k}\ln\cos^2\theta_i \\
&\leq \phi(B_0)+\sigma(k+1)+\sum_{i=0}^{k_0-1}\ln\cos^2\theta_i+\sum_{i=k_0}^{k}(-2\sigma) \\
&= \phi(B_0)+\sum_{i=0}^{k_0-1}\ln\cos^2\theta_i+\sigma(k+1)-2\sigma(k-k_0+1) \\
&= \phi(B_0)+\sum_{i=0}^{k_0-1}\ln\cos^2\theta_i+2\sigma k_0-\sigma(k+1) \;\to\; -\infty,\ (k\to\infty),
\end{aligned}$$

矛盾. 这样便存在 $\{x_k\}$ 的无穷子列 $\{x_k\}_{k\in K}$ 和数 $\delta > 0$, 使得对所有的 $k \in K$, 有 $\cos\theta_k \geq \delta$. 于是类似于定理 2.2 的证明过程, 可以推出 $\{\|g_k\|\}_{k\in K} \to 0$. 因 $f(x)$ 在水平集上是严格凸的, 其稳定点与全局极小点是一致的也是唯一的, 这样便可推得整个序列 $\{x_k\}$ 收敛到 $f(x)$ 的全局极小点 x^*. 证毕. □

下面我们给出拟牛顿法超线性收敛的一个充分必要条件.

定理 5.3 设 $f(x)$ 满足假设 5.1, $\{B_k\}$ 是非奇异的矩阵序列. 若迭代公式

$$x_{k+1} = x_k - B_k^{-1} g_k, \quad x_0 \in \mathbb{R}^n \tag{5.44}$$

产生的无穷序列 $\{x_k\}$ 收敛于 $f(x)$ 的稳定点 x^*, 则 $\{x_k\}$ 超线性收敛到 x^* 的充分必要条件是

$$\lim_{k\to\infty} \frac{\|[B_k - G(x^*)]d_k\|}{\|d_k\|} = 0, \tag{5.45}$$

其中 $d_k = x_{k+1} - x_k$, $G(x^*) = \nabla^2 f(x^*)$.

证 设牛顿步为 $d_k^N = -G_k^{-1} g_k$. 首先证明 (5.45) 等价于

$$\|d_k - d_k^N\| = o(\|d_k\|). \tag{5.46}$$

事实上, 若 (5.45) 式成立, 则

$$\begin{aligned}
\|d_k - d_k^N\| &= \|G_k^{-1}[G_k d_k + g_k]\| \\
&= \|G_k^{-1}[G_k - B_k]d_k\| \le \|G_k^{-1}\| \cdot \|[G_k - B_k]d_k\| \\
&\le C\big(\|[G_k - G(x^*)]d_k\| + \|[G(x^*) - B_k]d_k\|\big) \\
&= o(\|d_k\|).
\end{aligned}$$

反之, 若 (5.46) 式成立, 注意到 $\|G_k\|$ 是有界的, 故有

$$\|G_k(d_k - d_k^N)\| = o(\|d_k\|).$$

由 $G_k d_k^N = -g_k = B_k d_k$, 我们有

$$\|(G_k - B_k)d_k\| = o(\|d_k\|).$$

由上式及 $G(x)$ 的连续性即可推得 (5.45) 式成立.

注意到牛顿法的二阶收敛性结果

$$\|x_k + d_k^N - x^*\| \le C\|x_k - x^*\|^2,$$

我们有

$$\begin{aligned}\|d_k\| - \|x_k - x^*\| &\le \|x_k + d_k - x^*\| \\ &\le \|x_k + d_k^N - x^*\| + \|d_k - d_k^N\| \\ &\le C\|x_k - x^*\|^2 + o(\|d_k\|).\end{aligned}$$

由此可得 $\|d_k\| = O(\|x_k - x^*\|)$, 再代入上式即得

$$\|x_k + d_k - x^*\| = o(\|x_k - x^*\|).$$

至此已经完成了定理的证明. □

最后, 我们不加证明地列出 BFGS 算法的局部超线性收敛定理, 其详细的证明过程可参阅文献 [15].

定理 5.4 设 $f(x)$ 满足假设 5.1, x_0 和 B_0 为任意给定的初始点和初始正定对称矩阵. $\{x_k\}$ 是由 BFGS 算法产生的迭代序列且收敛于假设

5.1 (3) 中的 x^*. 那么, 若

$$\sum_{k=1}^{\infty} \|x_k - x^*\| < \infty,$$

则 $\{x_k\}$ 局部超线性收敛于 x^*.