拟牛顿法、最小二乘问题

文再文

北京大学北京国际数学研究中心

教材《最优化：建模、算法与理论》配套电子教案

http://bicmr.pku.edu.cn/~wenzw/optbook.html

致谢：本教案由丁思哲、张轩熙协助准备

提纲

割线方程的推导

设$f(x)$是二阶连续可微函数. 对$\nabla f(x)$在点x^{k+1}处一阶泰勒近似, 得

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

令$x = x^k$, 且$s^k = x^{k+1} - x^k$为点差, $y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$为梯度差, 得

$$\nabla^2 f(x^{k+1}) s^k + \mathcal{O}(\|s^k\|^2) = y^k.$$

现忽略高阶项$\|s^k\|^2$, 只希望近似海瑟矩阵的矩阵B^{k+1}满足方程

$$B^{k+1} s^k = y^k,$$

或其逆矩阵H^{k+1}满足

$$H^{k+1} y^k = s^k.$$

上述两个方程即称为割线方程.

曲率条件

由于近似矩阵必须保证迭代收敛，正如牛顿法要求海瑟矩阵正定，B^k正定也是必须的，即有必要条件

$$\left(s^k\right)^{\mathrm{T}} B^{k+1} s^k > 0 \Longrightarrow \left(s^k\right)^{\mathrm{T}} y^k > 0,$$

定义

曲率条件 在迭代过程中满足$\left(s^k\right)^{\mathrm{T}} y^k > 0, \forall k \in \mathbb{N}^+$.

如果线搜索使用Wolfe准则：

$$\nabla f(x^k + \alpha d^k)^{\mathrm{T}} d^k \geqslant c_2 \nabla f(x^k)^{\mathrm{T}} d^k,$$

其中$c_2 \in (0,1)$. 上式即$\nabla f(x^{k+1})^{\mathrm{T}} s^k \geqslant c_2 \nabla f(x^k)^{\mathrm{T}} s^k$. 在不等式两边同时减去$\nabla f(x^k)^{\mathrm{T}} s^k$，由于$c_2 - 1 < 0$且$s^k$是下降方向，因此最终有

$$(y^k)^{\mathrm{T}} s^k \geqslant (c_2 - 1) \nabla f(x^k)^{\mathrm{T}} s^k > 0.$$

拟牛顿算法的基本框架

拟牛顿算法的基本框架为:

算法 1 拟牛顿算法框架

Require: 初始坐标$x^0 \in \mathbb{R}^n$, 初始矩阵$B^0 \in \mathbb{R}^{n\times n}$(或$H^0$), $k=0$.

Ensure: x^K, B^K(或H^K).

1: 检查初始元素.
2: **while** 未达到停机准则 **do**
3: 　计算方向$d^k = -(B^k)^{-1}\nabla f(x^k)$或$d^k = -H^k\nabla f(x^k)$.
4: 　通过线搜索(Wolfe)产生步长$\alpha_k > 0$, 令$x^{k+1} = x^k + \alpha_k d^k$.
5: 　更新海瑟矩阵的近似矩阵B^{k+1}或其逆矩阵H^{k+1}.
6: 　$k \leftarrow k+1$.
7: **end while**

秩一更新(SR1)

定义

秩一更新 对于拟牛顿矩阵$B^k \in \mathbb{R}^{n\times n}$, 设$0 \neq u \in \mathbb{R}^n$且$a \in \mathbb{R}$待定, 则$uu^{\mathrm{T}}$是秩一矩阵, 且有秩一更新

$$B^{k+1} = B^k + auu^{\mathrm{T}}.$$

根据割线方程$B^{k+1}s^k = y^k$, 代入秩一更新的结果, 得到

$$(B^k + auu^{\mathrm{T}})s^k = y^k,$$

整理得

$$auu^{\mathrm{T}}s^k = (a \cdot u^{\mathrm{T}}s^k)u = y^k - B^k s^k.$$

由于$a \cdot u^{\mathrm{T}}s^k$是标量, 因此上式表明$u$和$y^k - B^k s^k$同向. 简单考虑不妨就令$u$和$y^k - B^k s^k$相等, 即$u = y^k - B^k s^k$. 代入上式得

$$(a \cdot (y^k - B^k s^k)^{\mathrm{T}} s^k)(y^k - B^k s^k) = y^k - B^k s^k.$$

秩一更新公式

再令$(a \cdot (y^k - B^k s^k)^{\mathrm{T}} s^k) \neq 0$, 则可以确定$a$为

$$a = \frac{1}{(y^k - B^k s^k)^{\mathrm{T}} s^k}.$$

由同样的过程可以推出基于H^k的秩一更新公式.

定理

拟牛顿算法的秩一更新公式　拟牛顿矩阵B^k的秩一更新公式为

$$B^{k+1} = B^k + \frac{uu^{\mathrm{T}}}{u^{\mathrm{T}} s^k}, \quad u = y^k - B^k s^k,$$

拟牛顿矩阵H^k的秩一更新公式为

$$H^{k+1} = H^k + \frac{vv^{\mathrm{T}}}{v^{\mathrm{T}} y^k}, \quad v = s^k - H^k y^k.$$

B^k和H^k的公式在形式上互为对偶. 实际上$H^k = (B^k)^{-1}$, 利用秩一更新的SMW公式即可推出基于H^k的公式, 反之亦然.

秩一更新公式的缺陷

即使B^k正定，由秩一公式更新的B^{k+1}无法保证正定.

定理

秩一更新公式使B^{k+1}正定的充分条件 使用秩一更新公式从B^k更新B^{k+1}, B^{k+1}正定的充分条件可以是:

(1) B^k正定;

(2) $u^{\mathrm{T}}s^k > 0$.

证明: 设$0 \neq w \in \mathbb{R}^n$, 则

$$w^{\mathrm{T}}B^{k+1}w = w^{\mathrm{T}}B^k w + \frac{w^{\mathrm{T}}uu^{\mathrm{T}}w}{u^{\mathrm{T}}s^k} = w^{\mathrm{T}}B^k w + \frac{(u^{\mathrm{T}}w)^2}{u^{\mathrm{T}}s^k} > 0.$$

同样地, 将上述定理中B换成H, $u^{\mathrm{T}}s^k$换成$v^{\mathrm{T}}y^k$, 仍然成立. 因此, 由于无法保证$u^{\mathrm{T}}s^k$或$v^{\mathrm{T}}y^k$恒大于0, 上述的秩一更新公式一般不用.

BFGS公式

BFGS公式的核心思想是对B^k进行秩二更新.

定义

秩二更新　对于拟牛顿矩阵$B^k \in \mathbb{R}^{n\times n}$, 设$0 \neq u, v \in \mathbb{R}^n$且$a, b \in \mathbb{R}$待定, 则有秩二更新形式

$$B^{k+1} = B^k + auu^\mathrm{T} + bvv^\mathrm{T}.$$

根据割线方程, 将秩二更新的待定参量式代入, 得

$$B^{k+1}s^k = (B^k + auu^\mathrm{T} + bvv^\mathrm{T})s^k = y^k,$$

整理可得

$$(a \cdot u^\mathrm{T}s^k)u + (b \cdot v^\mathrm{T}s^k)v = y^k - B^k s^k.$$

简单的取法是令$(a \cdot u^\mathrm{T}s^k)u$对应$y^k$相等, $(b \cdot v^\mathrm{T}s^k)v$对应$-B^k s^k$相等, 即有

$$a \cdot u^\mathrm{T}s^k = 1, \quad u = y^k, \quad b \cdot v^\mathrm{T}s^k = -1, \quad v = B^k s^k.$$

BFGS公式

将上述参量代入割线方程，即得BFGS更新公式

$$B^{k+1} = B^k + \frac{uu^{\mathrm{T}}}{(s^k)^{\mathrm{T}}u} - \frac{vv^{\mathrm{T}}}{(s^k)^{\mathrm{T}}v}.$$

利用SMW公式以及$H^k = (B^k)^{-1}$，可以推出关于H^k的BFGS公式.

定义

BFGS公式 在拟牛顿类算法中，基于B^k的*BFGS*公式为

$$B^{k+1} = B^k + \frac{y^k\left(y^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}}y^k} - \frac{B^k s^k\left(B^k s^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}}B^k s^k},$$

基于H^k的*BFGS*公式为

$$H^{k+1} = \left(I - \frac{s^k\left(y^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}}y^k}\right)H^k\left(I - \frac{y^k\left(s^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}}y^k}\right) + \frac{s^k\left(s^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}}y^k}.$$

推导H^k的BFGS公式之提示

对于可逆矩阵$B\in\mathbb{R}^{n\times n}$与矩阵$U\in\mathbb{R}^{n\times m}, V\in\mathbb{R}^{n\times m}$，SMW公式为：

$$\left(B+UV^{\mathrm{T}}\right)^{-1}=B^{-1}-B^{-1}U\left(I+V^{\mathrm{T}}B^{-1}U\right)^{-1}V^{\mathrm{T}}B^{-1}.$$

在BFGS的推导中，关于B^k的更新公式为：

$$B_{k+1}=B_k+\frac{y_ky_k^{\mathrm{T}}}{s_k^{\mathrm{T}}y_k}-\frac{B_ks_k\left(B_ks_k\right)^{\mathrm{T}}}{s_k^{\mathrm{T}}B_ks_k}=B_k+\begin{pmatrix}-\frac{B_ks_k}{s_k^{\mathrm{T}}B_ks_k} & \frac{y_k}{s_k^{\mathrm{T}}y_k}\end{pmatrix}\begin{pmatrix}s_k^{\mathrm{T}}B_k\\ y_k^{\mathrm{T}}\end{pmatrix}.$$

对照SMW公式，令式中$B=B_k$，且

$$U_k=\begin{pmatrix}-\frac{B_ks_k}{s_k^{\mathrm{T}}B_ks_k} & \frac{y_k}{s_k^{\mathrm{T}}y_k}\end{pmatrix},\quad V_k=\begin{pmatrix}B_ks_k & y_k\end{pmatrix},$$

此时公式的左端就等于B_{k+1}^{-1}，且右端只需计算一个2阶矩阵的逆．假设$B_k^{-1}=H_k$，由SMW公式就得到

$$H_{k+1}=\left(B_k+U_kV_k^{\mathrm{T}}\right)^{-1}=\left(I-\frac{s_ky_k^{\mathrm{T}}}{s_k^{\mathrm{T}}y_k}\right)H_k\left(I-\frac{y_ks_k^{\mathrm{T}}}{s_k^{\mathrm{T}}y_k}\right)+\frac{s_ks_k^{\mathrm{T}}}{s_k^{\mathrm{T}}y_k}.$$

BFGS公式的有效性

BFGS公式产生的B^{k+1}或H^{k+1}是否正定呢?

定理

BFGS公式使拟牛顿矩阵正定的充分条件　使用秩二更新公式从B^k或H^k更新B^{k+1}或H^{k+1}, 拟牛顿矩阵正定的充分条件可以是:

(1) B^k或H^k正定;

(2) 满足曲率条件$\left(s^k\right)^{\mathrm{T}} y^k > 0, \forall k \in \mathbb{N}^+$.

证明上述定理, 只需要从基于H^k的BFGS公式分析即可, 从而得到H^{k+1}和其逆B^{k+1}均正定.

因为在确定步长时使用某一Wolfe准则线搜索即可满足曲率条件, 因此BFGS公式产生的拟牛顿矩阵有望保持正定, 是有效算法.

基于H^k的BFGS格式恰好是优化问题

$$\begin{aligned} \min_{H} \quad & \mathbf{OPT} = \left\|H - H^k\right\|_W, \\ s.t. \quad & H = H^{\mathrm{T}}, \\ & Hy^k = s^k. \end{aligned}$$

的解. 上式中$\|\cdot\|_W$是加权范数, 定义为

$$\|H\|_W = \left\|W^{1/2} H W^{1/2}\right\|_F,$$

且W满足割线方程, 即$Ws^k = y^k$.

注意$Hy^k = s^k$是割线方程, 因此优化问题的意义是在满足割线方程的对称矩阵中找到距离H^k最近的矩阵H作为H^{k+1}. 因此我们可以进一步认知, BFGS格式更新的拟牛顿矩阵是正定对称的, 且在满足割线方程的条件下采取的是最佳逼近策略.

DFP公式

DFP公式利用与BFGS公式类似的推导方法，不同的是其以割线方程$H^{k+1}y^k = s^k$为基础进行对H^k的秩二更新.

基于H^k满足的DFP公式，利用SMW公式以及$B^k = (H^k)^{-1}$，可以推出关于B^k的DFP公式. (关键的推导步骤仍然可以参考推导BFGS公式时给出的提示)

定义

DFP公式 基于H^k的*DFP*更新公式为

$$H^{k+1} = H^k - \frac{H^k y^k (H^k y^k)^{\mathrm{T}}}{(y^k)^{\mathrm{T}} H^k y^k} + \frac{s^k (s^k)^{\mathrm{T}}}{(y^k)^{\mathrm{T}} s^k},$$

基于B^k的*DFP*更新公式为

$$B^{k+1} = \left(I - \frac{y^k (s^k)^{\mathrm{T}}}{(s^k)^{\mathrm{T}} y^k}\right) B^k \left(I - \frac{s^k (y^k)^{\mathrm{T}}}{(s^k)^{\mathrm{T}} y^k}\right) + \frac{y^k (y^k)^{\mathrm{T}}}{(s^k)^{\mathrm{T}} y^k}.$$

从优化意义上理解DFP公式

有了BFGS公式的优化意义做铺垫，讨论DFP公式的优化意义显得十分简单. 利用对偶性质，基于B^k的DFP格式将是优化问题

$$\begin{aligned} \min_{B} \quad & \mathbf{OPT} = \left\| B - B^k \right\|_W , \\ \text{s.t.} \quad & B = B^{\mathrm{T}}, \\ & Bs^k = y^k . \end{aligned}$$

的解. 上式中$\|\cdot\|_W$是加权范数，定义为

$$\|B\|_W = \left\| W^{1/2} B W^{1/2} \right\|_F ,$$

且W满足另一割线方程，即$Wy^k = s^k$.
注意$Bs^k = y^k$是另一割线方程，因此优化问题的意义是在满足割线方程的对称矩阵中找到距离B^k最近的矩阵B作为B^{k+1}.

尽管DFP格式与BFGS对偶, 但从实际效果而言, DFP格式的求解效率整体上不如BFGS格式. M.J.D. Powell曾求解问题

$$\min_{x\in\mathbb{R}^2} \quad f(x) = \frac{1}{2}\|x\|_2^2 .$$

设置初始值

$$B^0 = \begin{pmatrix} 1 & 0 \\ 0 & \lambda \end{pmatrix}, \quad x_1 = \begin{pmatrix} \cos\psi \\ \sin\psi \end{pmatrix},$$

其中$\tan^2\psi = \lambda$. 当误差阈$\epsilon = 10^{-4}$时, 分别取λ为不同的值, 使用BFGS算法与DFP算法所产生的迭代步数分别如下表(见下页)所示. 由此看出, 在本问题中, BFGS算法的求解效率要远高于DFP算法.
(参考文献: Powell M J D. How bad are the BFGS and DFP methods when the objective function is quadratic?[J]. Mathematical Programming, 1986, 34(1): 34-47.)

DFP公式的缺陷

Table: BFGS方法的迭代次数

λ \ ϵ	0.1	0.01	10^{-4}	10^{-8}
10	5	6	8	10
100	7	8	10	12
10^4	12	13	15	17
10^6	17	18	20	22
10^9	24	25	27	29

Table: DFP方法的迭代次数

λ \ ϵ	0.1	0.01	10^{-4}	10^{-8}
10	10	13	16	19
30	25	32	37	40
100	80	99	107	111
300	237	290	307	313
10^3	787	958	1006	1014

提纲

BFGS全局收敛性

我们利用Zoutendijk条件得到基本收敛性. 需要复习的读者可参看纸质本Page 213的定理6.1.

根据对BFGS格式有效性的分析, 我们先确保初始矩阵B^0是对称正定的.

定理

BFGS全局收敛性　设初始矩阵B^0是对称正定矩阵, 目标函数$f(x)$是二阶连续可微函数, 下水平集

$$\mathcal{L} = \{x \in \mathbb{R}^n | f(x) \leqslant f(x^0)\}$$

凸, 且存在$m, M \in \mathbb{R}^+$使得对$\forall z \in \mathbb{R}^n, x \in \mathcal{L}$满足

$$m \|z\|^2 \leqslant z^{\mathrm{T}} \nabla^2 f(x) z \leqslant M \|z\|^2$$

(即$z^{\mathrm{T}} \nabla^2 f(x) z$被$\|z\|$控制), 那么BFGS格式结合Wolfe线搜索的拟牛顿算法全局收敛到$f(x)$的极小值点x^*.

BFGS全局收敛性的证明

通过Zoutendijk条件来证明收敛性. 因为BFGS拟牛顿矩阵在定理条件下对称正定，每个拟牛顿方向是下降方向. 因此, 只需要证明搜索方向与负梯度的夹角不太差。

基于B^k的BFGS格式为

$$B^{k+1} = B^k + \frac{y^k \left(y^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}} y^k} - \frac{B^k s^k \left(B^k s^k\right)^{\mathrm{T}}}{\left(s^k\right)^{\mathrm{T}} B^k s^k},$$

通过这一公式, 我们可以说明:

- (a) $\mathrm{Tr}\left(B^{k+1}\right) = \mathrm{Tr}\left(B^k\right) - \frac{\left\|B^k s^k\right\|^2}{\left(s^k\right)^{\mathrm{T}} B^k s^k} + \frac{\left\|y^k\right\|^2}{\left(s^k\right)^{\mathrm{T}} y^k}$,
- (b) $\det\left(B^{k+1}\right) = \det\left(B^k\right) \frac{\left(s^k\right)^{\mathrm{T}} y^k}{\left(s^k\right)^{\mathrm{T}} B^k s^k}$.

关于迹的公式是显然的, 因为迹的运算保和. 我们主要说明为何关于行列式的结论成立(习题6.11).

BFGS全局收敛性的证明

为说明(b)式成立, 先证明结论: 设$x, y, u, v \in \mathbb{R}^n$, 则

$$\det\left(I_{n\times n} + xy^{\mathrm{T}} + uv^{\mathrm{T}}\right) = \left(1 + y^{\mathrm{T}}x\right)\left(1 + v^{\mathrm{T}}u\right) - \left(x^{\mathrm{T}}v\right)\left(y^{\mathrm{T}}u\right).$$

Proof: 令$U = (x, u)$， $V = (y^{\mathrm{T}}; v^{\mathrm{T}})$。构造分块矩阵$A = (I_{n\times n}, U; V, I_{2\times 2})$. 利用Schur补的性质成立

$$\det\left(I_{n\times n} + xy^{\mathrm{T}} + uv^{\mathrm{T}}\right) = \det\left(I_{n\times n} + UV\right) = \det\left(I_{2\times 2} + U^{\mathrm{T}}V^{\mathrm{T}}\right).$$

利用上述定理, 将BFGS公式的左右两边乘以$(B^k)^{-1}$(B^k正定), 并设$x = \frac{-s^k}{\sqrt{(s^k)^T B^k s^k}}, y = \frac{B^k s^k}{\sqrt{(s^k)^T B^k s^k}}, u = \frac{(B^k)^{-1} y^k}{\sqrt{(y^k)^T s^k}}, v = \frac{y^k}{\sqrt{(y^k)^T s^k}}$, 代入即可证明结论成立.

定义$\cos\theta_k = \frac{\left(s^k\right)^{\mathrm{T}} B^k s^k}{\|s^k\| \|B^k s^k\|}$是欲求夹角的余弦. 这是因为

$$
\begin{aligned}
s^k &= x^{k+1} - x^k = -\alpha_k \left(B^k\right)^{-1} \nabla f\left(x^k\right), \\
B^k s^k &= -\alpha_k \nabla f\left(x^k\right).
\end{aligned}
$$

再设

$$
q_k = \frac{\left(s^k\right)^{\mathrm{T}} B^k s^k}{\left(s^k\right)^{\mathrm{T}} s^k}, \quad m_k = \frac{\left(y^k\right)^{\mathrm{T}} s^k}{\left(s^k\right)^{\mathrm{T}} s^k}, \quad M_k = \frac{\left(y^k\right)^{\mathrm{T}} y^k}{\left(y^k\right)^{\mathrm{T}} s^k},
$$

将上述定义代入(b)式以及余弦式, 得到

(c) $\det\left(B^{k+1}\right) = \det\left(B^k\right) \frac{m_k}{q_k}$.

(d) $\frac{\left\|B^k s^k\right\|^2}{\left(s^k\right)^{\mathrm{T}} B^k s^k} = \frac{q_k}{\cos^2\theta_k}$.

上述(a),(b),(c),(d)均是准备公式.

BFGS全局收敛性的证明

我们目标是构造一个不等式, 使得$\cos\theta_k > 0, k \to \infty$. 设

$$\Psi(B) = \mathrm{Tr}(B) - \ln(\det(B)),$$

注意上式成立$\Psi(B) > 0$.
在上式中代入$B = B^{k+1}$以及上述准备公式, 成立

$$\begin{aligned}\Psi\left(B^{k+1}\right) &= \mathrm{Tr}\left(B^{k+1}\right) - \ln\left(\det\left(B^{k+1}\right)\right) \\ &\leqslant \Psi\left(B^{k}\right) + (M_k - \ln(m_k) - 1) + 2\ln(\cos\theta_k).\end{aligned}$$

同时注意到$m_k \geqslant m, M_k \leqslant M$, 所以又有

$$\begin{aligned}\Psi\left(B^{k+1}\right) &\leqslant \Psi\left(B^{k}\right) + (M_k - \ln(m_k) - 1) + 2\ln(\cos\theta_k) \\ &\leqslant \Psi\left(B^{0}\right) + (k+1)(M - \ln(m) - 1) + 2\sum_{j=0}^{k}\ln(\cos\theta_j).\end{aligned}$$

如果我们假设迭代将不会停止, 则右式若无界, 将导出矛盾.

BFGS全局收敛性的证明

不妨设$cos\theta_k \to 0$，因此有$\ln(\cos^2\theta_k) \to -\infty$. 这意味着, 存在 $K \in \mathbb{N}^+$, 使得对$\forall j \geqslant K$, 均成立

$$\ln(\cos^2\theta_j) < -2(M - \ln(m) - 1) = -2C < 0,$$

联立上述两个最近的控制式, 注意$\Psi\left(B^{k+1}\right) > 0$, 则有$k \to \infty$时

$$0 < \Psi\left(B^0\right) + (k+1)\,C + 2\sum_{j=0}^{K} \ln\left(\cos\theta_j\right) + \sum_{j=K+1}^{k} (-2C) \to -\infty,$$

这就导出了矛盾. 因此$\cos\theta_k \to 0$是不成立的. 换句话说, 存在子列$\{j_k\}_{k=1,2,\ldots}$, 使得 $\cos\theta_{j_k} \geqslant \delta > 0$. 根据Zoutendijk条件, 又可以得到

$$\lim_{k\to\infty}\inf \left\|\nabla f\left(x^k\right)\right\| \to 0.$$

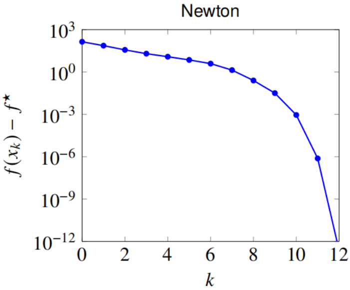
结合上述问题对$x \in \mathcal{L}$是强凸的, 所以导出$x^k \to x^*$.

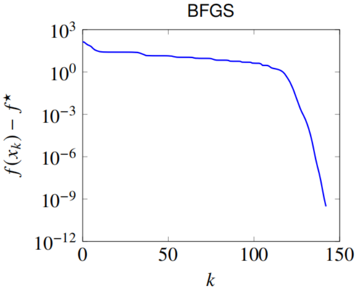
BFGS方法的收敛速度之例

例　考虑极小化问题

$$\min_{x\in\mathbb{R}^{100}} \quad c^{\mathrm{T}}x - \sum_{i=1}^{500} \ln\left(b_i - a_i^{\mathrm{T}}x\right),$$

下图展示了误差$f(x_k)-f^*$与迭代次数k之间的关系(k是迭代次数). 虽然BFGS方法的迭代次数显著得多, 但由于牛顿法每次迭代的计算代价为$\mathcal{O}(n^3)$加上计算海瑟矩阵的代价, 而BFGS方法的每步计算代价仅为$\mathcal{O}(n^2)$, 因此BFGS算法可能更快取得优势解.





提纲

有限内存方法的基本思路

基本思路　标准的拟牛顿近似矩阵的更新公式可以记为

$$B^{k+1}=g(B^k,s^k,y^k),\ s^k=x^{k+1}-x^k,y^k=\nabla f(x^{k+1})-\nabla f(x^k).$$

如果只保存最近的m组数据, 那么迭代公式可以写成

$$B^{k+1}=g(g(\cdots g(B^{k-m+1},s^{k-m+1},y^{k-m+1}))).$$

考虑BFGS方法：

$$d^k=-(B^k)^{-1}\nabla f(x^k)=-H^k\nabla f(x^k).$$

重写BFGS更新公式为

$$H^{k+1}=(V^k)^{\mathrm{T}}H^kV^k+\rho_k s^k(s^k)^{\mathrm{T}},$$

其中

$$\rho_k=\frac{1}{(y^k)^{\mathrm{T}}s^k},\quad V^k=I_{n\times n}-\rho_k y^k(s^k)^{\mathrm{T}}.$$

将上式递归地展开m次，即

$$
\begin{aligned}
H^k = & \left(\prod_{j=k-m}^{k-1} V^j\right)^{\mathrm{T}} H^{k-m} \left(\prod_{j=k-m}^{k-1} V^j\right) + \\
& \rho_{k-m} \left(\prod_{j=k-m+1}^{k-1} V^j\right)^{\mathrm{T}} s^{k-m} \left(s^{k-m}\right)^{\mathrm{T}} \left(\prod_{j=k-m}^{k-1} V^j\right) + \cdots + \\
& \rho_{k-1} s^{k-1} \left(s^{k-1}\right)^{\mathrm{T}}.
\end{aligned}
$$

为了节省内存，我们只展开m次，利用H^{k-m}进行计算，即可求出H^{k+1}.

下面介绍一种不计算H^k，只利用展开式计算$d^k = -H^k \nabla f(x^k)$的巧妙算法：双循环递归算法. 它利用迭代式的结构尽量节省计算d^k的开销.

将等式两边同右乘$\nabla f(x^k)$, 则等式左侧为$-d^k$. 观察等式右侧需要计算

$$V^{k-1}\nabla f(x^k), \cdots, V^{k-m}\cdots V^{k-1}\nabla f(x^k).$$

这些计算可以递归地进行. 同时在计算$V^{k-l}\cdots V^{k-1}\nabla f(x^k)$的过程中, 可以计算上一步的$\rho_{k-l}(s^{k-l})^{\mathrm{T}}[V^{k-l+1}\cdots V^{k-1}\nabla f(x^k)]$, 这是一个标量. 记

$$q = V^{k-m}\cdots V^{k-1}\nabla f(x^k),$$

$$\alpha_{k-l} = \rho_{k-l}(s^{k-l})^{\mathrm{T}}[V^{k-l+1}\cdots V^{k-1}\nabla f(x^k)],$$

因此递归公式可化为如下的形式:

$$H^k\nabla f(x^k) = \left(\prod_{j=k-m}^{k-1} V^j\right)^{\mathrm{T}} H^{k-m}q + \left(\prod_{j=k-m+1}^{k-1} V^j\right)^{\mathrm{T}} s^{k-m}\alpha_{k-m} + \cdots + s^{k-1}\alpha_{k-1}$$

在双循环递归算法中，除了上述第一个循环递归过程(自下而上)外，还有以下第二个循环递归过程. 我们需要在公式中自上而下合并每一项. 以前两项为例，它们有公共的因子$(V^{k-m+1}\cdots V^{k-1})^{\mathrm{T}}$，提取后可以将前两项写为(注意将$V^{k-m}$的定义回代)

$$
\begin{aligned}
&\left(V^{k-m+1}\cdots V^{k-1}\right)^{\mathrm{T}}\left[\left(V^{k-m}\right)^{\mathrm{T}} r+\alpha_{k-m} s^{k-m}\right] \\
=&\left(V^{k-m+1}\cdots V^{k-1}\right)^{\mathrm{T}}\left(r+\left(\alpha_{k-m}-\beta\right) s^{k-m}\right),
\end{aligned}
$$

这正是第二个循环的迭代格式. 注意合并后原递归式的结构仍不变，因此可以递归地计算下去. 最后，变量r就是我们期望的结果$H^k\nabla f(x^k)$.

L-BFGS双循环递归算法

拟牛顿算法的基本框架为:

算法 2 L-BFGS双循环递归

Require: 初始化$q \leftarrow \nabla f(x^k)$.

Ensure: r, 即$H^k \nabla f(x^k)$.

```
检查初始元素.
for i = k − 1, ··· , k − m do
    计算并保存 α_i ← ρ_i (s^i)^T q.
    更新 q ← q − α_i y^i.
end for
初始化 r ← Ĥ^{k−m} q, 其中 Ĥ^{k−m} 是 H^{k−m} 的近似矩阵.
for i = k − m, ··· , k − 1 do
    计算 β ← ρ_i (y^i)^T r.
    更新 r ← r + (α_i − β) s^i.
end for
```

算法分析

L-BFGS双循环递归算法约需要$4mn$次乘法运算, $2mn$次加法运算; 若近似矩阵$\hat{H}^{k-m}$是对角矩阵, 则额外需要n次乘法运算. 由于m不会很大, 因此算法的复杂度是$\mathcal{O}(mn)$. 算法需要的额外存储为临时变量α_i, 其大小是$\mathcal{O}(m)$.

$\hat{H}^{k-m}$的一种取法可以是取对角矩阵

$$\hat{H}^{k-m} = \gamma_k I_{n\times n} \triangleq \frac{\left(s^{k-1}\right)^{\mathrm{T}} y^{k-1}}{\left(y^{k-1}\right)^{\mathrm{T}} y^{k-1}} I_{n\times n}.$$

这恰好是BB方法的第一个步长.

提纲

最小二乘问题

$$\min_{x} \quad f(x) = \frac{1}{2}\sum_{j=1}^{m} r_j^2(x) \tag{1}$$

- 其中$r_j : \mathbb{R}^n \to \mathbb{R}$ 是光滑函数，并且假设$m \geq n$. 称r_j 为残差.
- 记$r : \mathbb{R}^n \to \mathbb{R}^m$

$$r(x) = (r_1(x), r_2(x), ..., r_m(x))^T.$$

问题可以表述为$\min \quad f(x) = \frac{1}{2}\|r(x)\|_2^2$

- 一般情况下不是凸问题
- 问题(1)是无约束优化问题，可以直接使用线搜索或信赖域法求解

最小二乘问题

- 记$J(x) \in \mathbb{R}^{m\times n}$是向量值函数$r(x)$ 在点x 处的雅可比矩阵：

$$J(x) = \begin{bmatrix} \nabla r_1(x)^T \\ \nabla r_2(x)^T \\ \vdots \\ \nabla r_m(x)^T \end{bmatrix}.$$

- $f(x)$的梯度和海瑟矩阵：

$$\nabla f(x) = \sum_{j=1}^{m} r_j(x)\nabla r_j(x) = J(x)^{\mathrm{T}} r(x), \tag{2a}$$

$$\nabla^2 f(x) = \sum_{j=1}^{m} \nabla r_j(x)\nabla r_j(x) + \sum_{i=1}^{m} r_i(x)\nabla^2 r_i(x) \tag{2b}$$

$$= J(x)^{\mathrm{T}} J(x) + \sum_{i=1}^{m} r_i(x)\nabla^2 r_i(x), \tag{2c}$$

高斯-牛顿方法

- 使用近似$\nabla^2 f_k \approx J_k^T J_k$. 省略 $\nabla^2 r_j$ 的计算，减少了计算量。其中J_k、r_k分别是$J(x_k)$、$r(x_k)$的简写。
- 高斯-牛顿法的迭代方向 d_k^{GN} 满足：

$$J_k^T J_k d_k^{GN} = -J_k^T r_k. \tag{3}$$

- 另一种理解: 在点x_k处，考虑近似$r(x_k + d) \approx r_k + J_k d$ 得到：

$$\min_d \quad f(x_k + d) = \frac{1}{2}\|r(x_k + d)\|^2 \approx \frac{1}{2}\|J_k d + r_k\|^2.$$

  求解该问题时也可对J_k做QR分解或SVD分解，无需计算出$J_k^T J_k$。
- 然后更新 $x_{k+1} = x_k + \alpha_k d_k$.

Levenberg-Marquardt (LM) 方法

- 当J_k不满秩时，(3)有很多个解，应该怎么更新?
- LM方法本质为信赖域方法，更新方向为如下问题的解

$$\min_{d} \quad \frac{1}{2}\|J^k d + r^k\|^2, \quad \text{s.t.} \quad \|d\| \le \Delta_k. \tag{4}$$

- LM 方法将如下近似当作信赖域方法中的m_k:

$$m_k(d) = \frac{1}{2}\|r^k\|^2 + d^{\mathrm{T}}(J^k)^{\mathrm{T}} r^k + \frac{1}{2} d^{\mathrm{T}}(J^k)^{\mathrm{T}} J^k d. \tag{5}$$

- 同样使用$(J^k)^{\mathrm{T}} J^k$来近似海瑟矩阵.

- 类似信赖域方法，引入如下定义来衡量$m_k(d)$近似程度的好坏：

$$\rho_k = \frac{f\left(x^k\right) - f\left(x^k + d^k\right)}{m_k\left(0\right) - m_k\left(d^k\right)} \tag{6}$$

为函数值实际下降量与预估下降量（即二阶近似模型下降量）的比值.

- 如果ρ_k接近1，说明$m_k(d)$来近似$f(x)$是比较成功的，则应该扩大Δ_k；如果ρ_k非常小甚至为负，就说明我们过分地相信了二阶模型$m_k(d)$，此时应该缩小Δ_k.
- 只有当ρ_k足够大，也就是对模型拟合较好时，才进行一步更新，否则不更新。

Levenberg-Marquardt 方法

Algorithm 3 Levenberg-Marquardt 方法

给定最大半径$\Delta_{\max}$，初始半径Δ_0，初始点x^0，$k \leftarrow 0$.
给定参数$0 \le \eta < \bar{\rho}_1 < \bar{\rho}_2 < 1$，$\gamma_1 < 1 < \gamma_2$.
while 未达到收敛准则 **do**
    计算子问题(4)得到迭代方向d^k.
    根据(6) 计算下降率ρ_k.
    更新信赖域半径：

$$\Delta_{k+1} = \begin{cases} \gamma_1 \Delta_k, & \rho_k < \bar{\rho}_1, \\ \min\{\gamma_2 \Delta_k, \Delta_{\max}\}, & \rho_k > \bar{\rho}_2 \text{ 以及 } \|d^k\| = \Delta_k, \\ \Delta_k, & \text{其他}. \end{cases}$$

    更新自变量：

$$x^{k+1} = \begin{cases} x^k + d^k, & \rho_k > \eta, \qquad /* \text{ 只有下降比例足够大才更新} */ \\ x^k, & \text{其他}. \end{cases}$$

    $k \leftarrow k + 1$.
end while

子问题求解

Corollary

向量d^*是信赖域子问题

$$\min_d \quad \frac{1}{2}\|Jd+r\|^2, \quad \text{s.t.} \quad \|d\| \le \Delta$$

的解当且仅当d^*是可行解并且存在数$\lambda \ge 0$使得

$$(J^{\mathrm{T}}J+\lambda I)d^* = -J^{\mathrm{T}}r, \tag{7}$$

$$\lambda(\Delta - \|d^*\|) = 0. \tag{8}$$

- 问题(7)等价于线性最小二乘问题，具体实现时可利用系数矩阵的结构

$$\min_d \quad \frac{1}{2}\left\|\begin{bmatrix} J \\ \sqrt{\lambda} I \end{bmatrix} p + \begin{bmatrix} r \\ 0 \end{bmatrix}\right\|^2.$$

子问题求解

$$\min_d \quad \frac{1}{2}\left\|\begin{bmatrix} J \\ \sqrt{\lambda} I \end{bmatrix} p + \begin{bmatrix} r \\ 0 \end{bmatrix}\right\|^2.$$

- 在试探λ的值时，J的块不变，设$J = QR$，则

$$\begin{bmatrix} J \\ \sqrt{\lambda} I \end{bmatrix} = \begin{bmatrix} QR \\ \sqrt{\lambda} I \end{bmatrix} = \begin{bmatrix} Q & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} R \\ \sqrt{\lambda} I \end{bmatrix}.$$

- 矩阵$\begin{bmatrix} R \\ \sqrt{\lambda} I \end{bmatrix}$有较多的零元素，可以使用Household变换或Givens变换完成QR分解。
- 如果矩阵J没有显式形式，只能提供矩阵乘法，则仍然可以用截断共轭梯度法。

Theorem

假设常数$\eta \in \left(0, \frac{1}{4}\right)$，下水平集$\mathcal{L}$是有界的且每个$r_i(x)$在下水平集$\mathcal{L}$的一个邻域$\mathcal{N}$内是利普希茨连续可微的．假设对于任意的$k$，子问题(4)的近似解$d_k$满足

$$m_k(0) - m_k(d_k) \geq c_1 \|J_k^{\mathrm{T}} r_k\| \min \left\{ \Delta_k, \frac{\|J_k^{\mathrm{T}} r_k\|}{\|J_k^{\mathrm{T}} J_k\|} \right\},$$

其中$c_1 > 0$且$\|d_k\| \leq \gamma \Delta_k, \gamma \geq 1$，则

$$\lim_{k\to\infty} \nabla f(x_k) = \lim_{k\to\infty} J_k^{\mathrm{T}} r_k = 0.$$

- 根据$r_j(x)$的连续性，存在$M > 0$，使得$\|J_k^T J_k\| \leq M$对任意的k成立。由于f有下界，可以直接套用信赖域算法全局收敛性的证明。

LMF

- 信赖域型LM 方法本质上是固定信赖域半径Δ, 通过迭代寻找满足条件的乘子λ, 每一步迭代需要求解线性方程组

$$\left(J^{\mathrm{T}}J + \lambda I\right) d = -J^{\mathrm{T}} r$$

- 调整λ的大小等价于调整信赖域半径的大小，Δ被λ隐式决定。
- LM的更新基于Δ，LMF的更新直接基于λ，每一步求解子问题：

$$\min_{d} \quad \|Jd + r\|_2^2 + \lambda \|d\|_2^2.$$

- 调整λ的原则可以参考信赖域半径的调整原则
- 考虑参数ρ_k

$$\rho_k = \frac{f\left(x_k\right) - f\left(x_k + d_k\right)}{m_k(0) - m_k\left(d_k\right)} \tag{9}$$

较大可以减小下一步的λ，较小可以增大下一步的λ。

Algorithm 4 LMF 方法

给定初始点x_0，初始乘子λ_0，$k \leftarrow 0$.
给定参数$0 \le \eta < \bar{\rho}_1 < \bar{\rho}_2 < 1$，$\gamma_1 < 1 < \gamma_2$.
while 未达到收敛准则 **do**
    求解LM 方程$((J_k)^{\mathrm{T}} J_k + \lambda I)d = -(J_k)^{\mathrm{T}} r_k$得到迭代方向$d_k$.
    根据(9)式计算下降率ρ_k.
    更新乘子：

$$\lambda_{k+1} = \begin{cases} \gamma_2 \lambda_k, & \rho_k < \bar{\rho}_1, \qquad /* \text{ 扩大乘子（缩小信赖域半径）} */ \\ \gamma_1 \lambda_k, & \rho_k > \bar{\rho}_2, \qquad /* \text{ 缩小乘子（扩大信赖域半径）} */ \\ \lambda_k, & \text{其他.} \qquad /* \text{ 乘子不变} */ \end{cases}$$

    更新自变量：

$$x_{k+1} = \begin{cases} x_k + d_k, & \rho_k > \eta, \qquad /* \text{ 只有下降比例足够大才更新} */ \\ x_k, & \text{其他.} \end{cases}$$

    $k \leftarrow k + 1$.
end while

- 大残量问题中，海瑟矩阵的第二部分不可忽视，此时高斯–牛顿法和LM 方法可能只有线性的收敛速度。
- 此时如果直接使用牛顿法，则开销太大；直接使用拟牛顿法，又似乎忽略了问题的特殊结构。
- 重新写出海瑟矩阵：

$$\nabla^2 f(x) = J(x)^{\mathrm{T}} J(x) + \sum_{i=1}^{m} r_i(x) \nabla^2 r_i(x)$$

第一项是容易求解，可以保留。第二项不易求解但不可忽略，用拟牛顿法进行近似。

大残量问题的拟牛顿算法

- 使用B_k来表示$\nabla^2 f(x_k)$的近似矩阵，T_k表示$\sum_{j=1}^m r_j(x_k)\nabla^2 r_j(x_k)$的近似，即

$$B_k = J_k^{\mathrm{T}} J_k + T_k,$$

- 目标为

$$T_{k+1} \approx \sum_{j=1}^{m} r_j\left(x_{k+1}\right)\nabla^2 r_j\left(x_{k+1}\right)$$

- 记$s_k = x_{k+1} - x_k$，$T_k + 1$ 应该尽量保留原海瑟矩阵的性质

$$\begin{aligned}
T_{k+1} s_k &\approx \sum_{j=1}^{m} r_j(x_{k+1})\nabla^2 r_j(x_{k+1}) s_k \\
&\approx \sum_{j=1}^{m} r_j(x_{k+1})(\nabla r_j(x_{k+1}) - \nabla r_j(x_k)) \\
&= J_{k+1}^{\mathrm{T}} r_{k+1} - J_k^{\mathrm{T}} r_{k+1}.
\end{aligned}$$

大残量问题的拟牛顿算法

- 拟牛顿条件为：

$$T_{k+1}s_k = J_{k+1}^{\mathrm{T}} r_{k+1} - J_k^{\mathrm{T}} r_{k+1}$$

- Dennis, Gay, 和Welsch给出的一种更新格式为：

$$T_{k+1} = T_k + \frac{\left(y^{\#} - T_k s_k\right) y^T + y\left(y^{\#} - T_k s_k\right)^T}{y^T s_k} - \frac{\left(y^{\#} - T_k s_k\right)^T s_k}{\left(y^T s\right)^2} y y^T$$

其中

$$\begin{aligned} s_k &= x_{k+1} - x_k \\ y &= J_{k+1}^T r_{k+1} - J_k^T r_k \\ y^{\#} &= J_{k+1}^T r_{k+1} - J_k^T r_{k+1} \end{aligned}$$

大残量问题的拟牛顿算法2

- 原始问题的拟牛顿公式：

$$(J_{k+1}^{\mathrm{T}}J_{k+1}+T_{k+1})s_k=J_{k+1}^{\mathrm{T}}r_{k+1}-J_k^T r_k.$$

- 改写上式得到

$$T_{k+1}s_k=\tilde{y}_k,$$

  其中

$$\tilde{y}_k=J_{k+1}^{\mathrm{T}}r_{k+1}-J_k^T r_k-J_{k+1}^{\mathrm{T}}J_{k+1}s_k.$$

- 再利用拟牛顿公式更新