数值优化（Numerical Optimization）学习系列-拟牛顿方法（Quasi-Newton）

下一步　于 2015-12-27 18:48:32 发布

概述

> 拟牛顿方法类似于最速下降法，在每一步迭代过程中仅仅利用梯度信息，但是通过度量梯度之间的变化，能够产生超线性的收敛效果。本节主要学习一下知识点：
> 1. 拟牛顿方程推导
> 2. 几个常见的拟牛顿方法
> 3. 拟牛顿方法的收敛性

拟牛顿方程

拟牛顿方法既有线搜索的影子也有牛顿方法的思想，下面从两个角度分别介绍拟牛顿方程，即在拟牛顿方法中要遵循的一个原则。

线搜索角度

假设在第K步迭代过程中，对点x_k进行建模

$$m_k(p) = f_k + \nabla f_k^T + \frac{1}{2}p^T B_k p$$

，这是一个相对标准的建模过程，在点x_k处寻找下一个搜索方向。该模型满足 $m_k(0) = f_k$; $\nabla m_k(0) = \nabla f_k^T$。
此时如果B为正定矩阵，则最优解为 $p_k = -B_k^{-1}\nabla f_k$。则下一个迭代值 $x_{k+1} = x_k + \alpha_k p_k$.
问题来了如何构造有效的 B_k呢，如果选择Hessian矩阵该方法就为线搜索的牛顿方法。
高人就想出了通过当前点和上一步的搜索点构造该矩阵的方法， **需要满足模型m和目标函数f在x_k, x_{k+1}保持梯度一致。**

此时在x_{k+1}处的模型为

$$m_{k+1}(p) = f_{k+1} + \nabla f_{k+1}^T + \frac{1}{2}p^T B_{k+1} p$$

，需要满足 x_k，x_{k+1}梯度一致。则有

$$\nabla m_{k+1}(x_{k+1}) = \nabla f_{k+1}$$
$$\nabla m_{k+1}(x_k) = \nabla f_k$$

等价于

$$\nabla m_{k+1}(0) = \nabla f_{k+1}$$
$$\nabla m_{k+1}(-\alpha_k p_k) = \nabla f_k$$

从而有

$$\nabla m_{k+1}(-\alpha_k p_k) = \nabla f_{k+1} - \alpha_k B_{k+1} p_k = \nabla f_k$$

。根据 $x_{k+1} = x_k + \alpha_k p_k$有 $B_{k+1}(x_{k+1} - x_k) = \nabla f_{k+1} - \nabla f_k$。一般情况下记

$$s_k = x_{k+1} - x_k$$
$$y_k = \nabla f_{k+1} - \nabla f_k$$

可以推出拟牛顿方程也叫（Secant equation）：

$$B_{k+1} s_k = y_k$$

牛顿法角度

拟牛顿方法也可以认为是一种特殊的共轭梯度算法，其主要思想是利用目标函数梯度的差分构造目标函数Hessian矩阵的某种近似，然后基于牛顿方程产生搜索方向，最后通过线搜索完成迭代过程。
假设在点x_{k+1}处进行泰勒展开有

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

两端对x求梯度并且 $x = x_k$有

$$\nabla f_k = \nabla f_{k+1} + \nabla^2 f_{k+1}(x_k - x_{k+1})$$

，整理后得到

$$\nabla^2 f_{k+1}(x_{k+1} - x_k) = \nabla f_{k+1} - \nabla f_k$$

由于Hessian矩阵比较难求解，用其近似矩阵 B_{k+1}代替，同时借用上面的表达式有 $B_{k+1} s_k = y_k$。
如果记 $H_{k+1} = B_{k+1}^{-1}$，也有 $H_{k+1} s_k = y_k$。

以上两种形式都称为拟牛顿方程。

> 拟牛顿方程：$B_{k+1}s_k=y_k$或者$H_{k+1}s_k=y_k$

拟牛顿方程成立条件

> 由于B_k需要满足对称正定，因此需要满足$s_k^Ty_k\geq 0$，如果步长满足一定条件，例如Wolfe条件，则上式一定成立。

前半部分是由于$s_k^TB_{k+1}s_k=s_ky_k$，由于B是正定的，肯定必须满足两个向量相乘大于0

后半部分是由于，Wolfe条件的第二个约束是

$$\nabla f_{k+1}^Ts_k\geq c_2\nabla f_k^Ts_k <=>$$
$$\nabla f_{k+1}^Ts_k-\nabla f_k^Ts_k\geq c_2\nabla f_k^Ts_k-\nabla f_k^Ts_k <=>$$
$$y_k^Ts_k\geq (c_1-1)\alpha_k\nabla f_k^Tp_k$$

由于 $c_2<1$如果搜索方向是下降方向则一定是大于0的。

拟牛顿方法

根据拟牛顿方程可以找到很多满足约束的矩阵，为求解方便需要进行一些约束，主要是秩的约束，由此产生了下面一些方法。

DFP方法

为保证B求解的唯一性，寻找满足条件约束并且离B_k最近的一个矩阵，因此问题转变为：

$$min||B-B_k||\ s.tB=B^T\ Bs_k=y_k$$

。
如果上面范数选择Weighted Frobenius范数并且加权矩阵采用平均Hessian，则可以推导出一个唯一确定的解

$$B_{k+1}=(\mathbf{I}-\rho_ky_ks_k^T)B_k(\mathbf{I}-\rho_ky_ks_k^T)+\rho_ky_ky_k^T$$

其中

$$\rho_k=1/(y_k^Ts_k)$$

。由于实际应用时会用到 $H_{k+1}=B_{k+1}^{-1}$，根据一个求逆公式（Morrison-Woodbury）可以得到

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

该构造方法称之为DFP方法

BFGS方法

类似于DFP方法，如果利用第二个拟牛顿方程，问题转变为：

$$min||H - H_k|| \; s.t H = H^T \; Hs_k = y_k$$

。
如果上面范数选择Weighted Frobenius范数并且加权矩阵采用平均Hessian，则可以推导出一个唯一确定的解

$$H_{k+1} = (\mathbf{I} - \rho_k s_k y_k^T) H_k (\mathbf{I} - \rho_k s_k y_k^T) + \rho_k s_k s_k^T$$

其中

$$\rho_k = 1/(y_k^T s_k)$$

。根据一个求逆公式（Morrison-Woodbury）可以得到

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

该构造方法称之为BFGS方法，利用四个发明者名字进行命名。

DFP和BFGS关系

可以看到DFP和BFGS好像是将s_k, y_k以及B_k，H_k进行了位置替换。理论证明他们互为对偶。

> 1. BFGS和DFP一个比较好的性质是，如果H_k是正定的，则下一个H_{k+1}也是正定的，可以从公式中推导出来。
> 2. BFGS和DFP都有一定能力进行自我修正，如果某个位置选择的矩阵不好，在未来几步呢，可以自我修复。这个能力，BFGS比DFP效果要好，这也是BFGS比较常用的原因。
> 3. 不好的地方就是：需要存储这个对称矩阵。

BFGS算法如下图所示

Algorithm 6.1 (BFGS Method).

Given starting point x_0, convergence tolerance $\epsilon > 0$,
inverse Hessian approximation H_0;
$k \leftarrow 0$;
while $\|\nabla f_k\| > \epsilon$;
Compute search direction

$$\boxed{p_k = -H_k \nabla f_k;}$$

Set $x_{k+1} = x_k + \alpha_k p_k$ where α_k is computed from a line search procedure to satisfy the Wolfe conditions (3.6);
Define $s_k = x_{k+1} - x_k$ and $y_k = \nabla f_{k+1} - \nabla f_k$;
Compute H_{k+1} by means of (6.17);
$k \leftarrow k + 1$;
end (while)

> 实际实现中初始值H_0一般选择单位，初始化步长为1。 Wolfe条件中的参数$c_1 = 10^{-1}, c_2 = 0.9$

SR-1方法

上面推导DFP和BFGS方法是从最优化角度进行考虑，由于满足拟牛顿方程解个数不止一个，另外一个自然的想法就是通过对B_k进行修正从而得到B_{k+1}，即

$$B_{k+1} = B_k + \Delta B_k$$

。习惯上根据 ΔB_k的秩来称呼校正公司，例如秩-1校正公式和秩-2校正公式

秩-2校正公式

DFP秩-2校正公式

$$H_{k+1} = H_k + a s_k s_k^T + b H_k y_k y_k^T H_k$$

BFGS秩-2校正公式

$$B_{k+1} = B_k + a y_k y_k^T + b B_k s_k s_k^T B_k$$

根据拟牛顿方程可以推导出参数a和b的值，最终结果和上述最优化问题保持一致。

秩-1校正公式（SR-1）

SR-1校正公式为

$$H_{k+1} = H_k + v_k v_k^T$$

根据拟牛顿条件可以推导出

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

优缺点

相对与BFGS

> 1. SR1能够更好的拟合Hessian矩阵，因此在一些带约束的问题或者部分可导的函数，不总是能满足Wolfe条件或者$s_k^T y_k$是大于0的
> 2. SR1最大缺点是不能保证每一求解到的矩阵H_{k+1}是正定的

Broyden族校正公式

校正公式为：

$$H_{k+1} = H_k + a s_k s_k^T + b(H_k y_k s_k^T + s_k y_k^T H_k) + c H_k y_k y_k^T H_k$$

可以根据牛顿条件求解到参数值。

SR-1、DFP和BFGS都是该一族算法，共同的问题是
1） 不能利用目标函数的稀疏性质
2）需要存储中间矩阵H

收敛性

> 拟牛顿方法具有全局收敛性并且有超线性的收敛速度

总结

通过本节的学习能够了解

1. 拟牛顿方程以及由来
2. DFP、BFGS方法的迭代公式以及使用条件、场景和优缺点
3. 了解其收敛速度