

第16B讲 信赖域算法

- 信赖域算法的基本结构
- 信赖域算法的收敛性
- 信赖域子问题的求解

信赖域算法的基本思想

- 在当前迭代点 $\mathbf{x}^{(k)}$ 的附近用一个简单函数近似目标函数 f .
- 用该近似函数在 $\mathbf{x}^{(k)}$ 的某个领域内的极小值点作为下一个迭代点。
- 特点：在每次迭代同时确定搜索方向和步长

信赖域方法

- 信赖域方法基本思想
 - 1) 局部区域
 - 2) 逼近模型
 - 3) 调节模型和区域
- 孙悟空的信赖域



娱乐时尚 国际在线
WWW.CRI.CN
gb.chinabroadcast.cn 资料



· 信赖域算法需要考虑如下相关问题:

- 目标函数 f 的 (简单) 近似形式.
- 点 $x^{(k)}$ 的邻域 (称为信赖域) 大小的确定.
- 函数值序列的下降性检测.
- 近似问题 (称为信赖域子问题) 的求解.

信赖域算法的基本结构

$x^{(k+1)}$ 为下面问题的解

$$\begin{aligned} \min \quad & f(x^{(k)}) + \nabla f(x^{(k)})^T (x - x^{(k)}) + \frac{1}{2} (x - x^{(k)})^T B_k (x - x^{(k)}) \\ \text{s.t.} \quad & \|x - x^{(k)}\| \leq \Delta_k, \end{aligned}$$

其中, $B_k \in R^{n \times n}$ 是 f 在 $x^{(k)}$ 处的 Hessian 阵或其近似,

参数 $\Delta_k > 0$ 控制 $x^{(k)}$ 的邻域大小.

信赖域子问题

$$\begin{aligned} \min \quad & f(x^{(k)}) + \nabla f(x^{(k)})^T d + \frac{1}{2} d^T B_k d \triangleq q_k(d), \\ \text{s.t.} \quad & \|d\| \leq \Delta_k. \end{aligned}$$

$$d = x - x^{(k)}$$

$$\longrightarrow d^{(k)} \longrightarrow x^{(k+1)} = x^{(k)} + d^{(k)}$$

信赖域 $D = \{d \in R^n \mid \|d\| \leq \Delta_k\}$ 信赖域半径 Δ_k

信赖域半径的确定

信赖域半径确定的标准:

- 希望在信赖域 \mathbf{D} 中二次函数 q 与函数 f 近似程度好
- 希望信赖域尽可能的大

定义 Δf_k 为 f 在第 k 步的实际下降 $\Delta f_k = f(x^{(k)}) - f(x^{(k)} + d^{(k)})$,

其中 $d^{(k)}$ 是信赖域子问题 (6.2) 的解.

令 Δq_k 为对应的预测下降量 $\Delta q_k = f(x^{(k)}) - q_k(d^{(k)})$.

定义比值
$$r_k = \frac{\Delta f_k}{\Delta q_k}.$$

$$r_k = \frac{\Delta f_k}{\Delta q_k}.$$

若 r_k 接近于 1,

认为二次函数 $q_k(d^{(k)})$ 在信赖域 D 上与目标函数 $f(x^{(k)} + d^{(k)})$ 的近似程度很好.

若 r_k 离 1 较远,

认为 $q_k(d^{(k)})$ 在信赖域 D 上与目标函数 $f(x^{(k)} + d^{(k)})$ 的近似程度不好.

用 r_k 与 1 的近似程度作为对信赖域半径是否合适的准则.

给定常数 $\eta, \eta_1, \eta_2 \in (0, 1)$ 满足 $\eta < \eta_1 < \eta_2$ η 接近于或等于 0, η_2 接近于 1

若 $r_k \geq \eta_2$, 可将信赖域的半径扩大, 即令 $\Delta_{k+1} > \Delta$.

若 $\eta < r_k < \eta_2$, 可保持信赖域半径不变, 即令 $\Delta_{k+1} = \Delta_k$

若 $r_k \leq \eta$, 减少信赖域半径, 即令 $\Delta_{k+1} < \Delta_k$.

算法 6.1 (信赖域算法)

步 0 (初始化) 取初始点 $x^{(0)} \in R^n$, $\bar{\Delta} > 0$, $\Delta_0 \in (0, \bar{\Delta})$, $\eta \in [0, \frac{1}{4})$, 精度 $\epsilon > 0$. 令 $k := 0$.

步 1 (收敛性检测) 若 $\|\nabla f(x^{(k)})\| \leq \epsilon$, 则算法终止. 得问题的解 $x^{(k)}$. 否则, 转步 2.

步 2 (子问题求解) 解信赖域子问题 (6.2) 得解 $d^{(k)}$.

步 3 (信赖域修正) 由 (6.3), (6.4)) 和 (6.5) 计算 r_k .

若 $r_k > \frac{3}{4}$, 则令 $\Delta_{k+1} = \min\{2\Delta_k, \bar{\Delta}\}$.

若 $\eta < r_k < \frac{1}{4}$, 则令 $\Delta_{k+1} = \frac{1}{2}\Delta_k$;

若 $\frac{1}{4} \leq r_k \leq \frac{3}{4}$, 则令 $\Delta_{k+1} = \Delta_k$.

$$\Delta_{k+1} = \frac{1}{5}\Delta_k$$

步 4 (可接受检测) 若 $r_k \leq \eta$, 令 $x^{(k+1)} = x^{(k)}$, $k := k + 1$, 转步 2;

否则令 $x^{(k+1)} = x^{(k)} + d^{(k)}$, $k := k + 1$, 转步 1.

$$(6.24) \quad \begin{aligned} \min \quad & f(x) + \nabla f(x)^T d + \frac{1}{2} d^T B d \stackrel{\Delta}{=} q(d), \\ \text{s.t.} \quad & \|d\| \leq \Delta. \end{aligned}$$

算法 6.4 (截断共轭梯度法)

步 0 给定 $\epsilon > 0$. 设 $d_0 = 0$, $r_0 = \nabla f(x)$, $p_0 = -r_0$. 令 $j := 0$.

步 1 若 $\|r_j\| \leq \epsilon$, 取 $d = d_j$ 为问题 (6.24) 的解, 算法终止. 否则, 转步 2.

步 2 若 $p_j^T B p_j \leq 0$, 确定 τ 使得 $d = d_j + \tau p_j$ 满足 $\|d\| = \Delta$, d 作为子问题的近似解, 停止计算. 否则, 计算

$$\alpha_j = r_j^T r_j / (p_j^T B p_j), \quad d_{j+1} = d_j + \alpha_j p_j.$$

步 3 若 $\|d_{j+1}\| \geq \Delta$, 确定 $\tau \geq 0$ 使得 $d = d_j + \tau p_j$ 满足 $\|d\| = \Delta$, 取 d 为子问题的近似解, 停止计算. 否则设 $r_{j+1} = r_j + \alpha_j B p_j$.

步 4 若 $\|r_{j+1}\| < \epsilon \|r_0\|$, 设 $d = d_{j+1}$ 为子问题的近似解, 停止计算; 否则设

$$\beta_{j+1} = r_{j+1}^T r_{j+1} / r_j^T r_j, \quad p_{j+1} = r_{j+1} + \beta_{j+1} p_j.$$

步 5 设 $j := j + 1$, 转步 1.

与传统的共轭梯度方法比较, 截断共轭梯度法增加了两个出口, 其一是搜索方向 p_j 为零方向或沿 B 的负曲率方向时 (步 2); 其二是 d_{j+1} 破坏了信赖域约束时 (步 3). 两种情况下近似解均在约束的边界上达到.

由于子问题 (6.2) 的可行域有界, 因此, 算法 6.1 的步 2 中的 $d^{(k)}$ 存在. 下面的定理说明, 若 $x^{(k)}$ 不是问题 (6.1) 的稳定点, 则预估下降量 $\Delta q_k(d^{(k)}) > 0$. 因此, 算法是适定的.

定理 6.1.1 设 $d^{(k)}$ 是问题 (6.2) 的解. 若 $\nabla f(x^{(k)}) \neq 0$, 则

$$\Delta q_k(d^{(k)}) = f(x^{(k)}) - q_k(d^{(k)}) > 0.$$

推论 6.1.1 设 $\{x^{(k)}\}$ 由算法 6.1 产生. 则序列 $\{f(x^{(k)})\}$ 单调非增.

算法6.1的收敛性定理

定理 6.2.1 设函数 f 连续可微有下界, 且存在常数 $\beta > 0$, 使得 $\|B_k\| \leq \beta$. 若在算法 6.1 中取 $\eta = 0$, 则

$$\liminf_{k \rightarrow \infty} \|\nabla f(x^{(k)})\| = 0. \quad (6.11)$$

我们给出 $\eta > 0$ 时, 算法 6.1 的如下收敛性定理.

定理 6.2.2 设定理 6.2.1 的条件满足且 ∇f Lipschitz 连续. 若在算法 6.1 中取 $\eta > 0$, 则

$$\lim_{k \rightarrow \infty} \|\nabla f(x^{(k)})\| = 0.$$

下面的定理给出了 Newton 型信赖域算法 (即在算法 6.1 中取 $B_k = \nabla^2 f(x^{(k)})$) 的收敛速度估计.

定理 6.2.3 设 $f \in C^2$ 有下界, 且由 Newton 型信赖域算法产生的点列 $\{x^{(k)}\}$ 有界. 则存在 $\{x^{(k)}\}$ 的聚点 x^* 满足一阶必要条件和二阶必要条件. 若再假设在 x^* 处 f 的 Hessian 阵正定, 则

$$\lim_{k \rightarrow \infty} r_k = 1, \quad \lim_{k \rightarrow \infty} x^{(k)} = x^*, \quad \inf \Delta_k > 0,$$

而且, 当 k 充分大时, $\|d^{(k)}\| < \Delta_k$. 此外, $\{x^{(k)}\}$ 的超线性收敛到 x^* . 若进一步假设 $\nabla^2 f$ 在 x^* 处 Lipschitz 连续, 则 $\{x^{(k)}\}$ 的收敛速度是二阶的.

信赖域子问题的求解

信赖域子问题为:

$$\min_{p \in \mathbb{R}^n} m_k(p) = f_k + g_k^T p + \frac{1}{2} p^T B_k p \quad \text{s.t. } \|p\| \leq \Delta_k, \quad (4.3)$$

Algorithm 4.2 (Cauchy Point Calculation).

Find the vector p_k^s that solves a linear version of (4.3), that is,

$$p_k^s = \arg \min_{p \in \mathbb{R}^n} f_k + g_k^T p \quad \text{s.t. } \|p\| \leq \Delta_k; \quad (4.9)$$

Calculate the scalar $\tau_k > 0$ that minimizes $m_k(\tau p_k^s)$ subject to satisfying the trust-region bound, that is,

$$\tau_k = \arg \min_{\tau \geq 0} m_k(\tau p_k^s) \quad \text{s.t. } \|\tau p_k^s\| \leq \Delta_k; \quad (4.10)$$

Set $p_k^c = \tau_k p_k^s$.

$$p_k^s = -\frac{\Delta_k}{\|g_k\|} g_k, \quad \tau_k = \begin{cases} 1 & \text{if } g_k^T B_k g_k \leq 0; \\ \min(\|g_k\|^3 / (\Delta_k g_k^T B_k g_k), 1) & \text{otherwise.} \end{cases}$$

信赖域子问题的求解

当前迭代点 $x \in R^n$

信赖域子问题为:

$$\begin{aligned} \min \quad & f(x) + \nabla f(x)^T d + \frac{1}{2} d^T B d \triangleq q(d), \\ \text{s.t.} \quad & \|d\| \leq \Delta. \end{aligned} \tag{6.20}$$

定理 6.4.1 d^* 是子问题 (6.20) 的全局最优解当且仅当 d^* 可行, 且存在常数 $\lambda^* \geq 0$ 满足 $(B + \lambda^* I)$ 半正定且有

$$\begin{cases} (B + \lambda^* I) d^* = -\nabla f(x), \\ \lambda^* (\Delta - \|d^*\|) = 0. \end{cases} \tag{6.21}$$

练习：习题6

4. 用信赖域算法 6.1 和子问题的截断共轭梯度算法 6.4 编程计算如下最优化问题 (取 $n = 10$) 的解.

$$\min f(x) = \sum_{i=1}^n [(1 - x_{2i-1})^2 + 10(x_{2i} - x_{2i-1}^2)^2].$$