

# 牛顿类算法

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:本教案由陈乐恒、丁思哲协助准备

# 梯度法的困难

考虑无约束优化问题

$$\min_{x \in \mathbb{R}^n} f(x),$$

使用**梯度下降法**, 给出的迭代格式是

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

由于梯度下降的基本策略是沿一阶最速下降方向迭代。当 $\nabla^2 f(x)$ 的条件数较大时, 它的收敛速度比较缓慢(**只用到一阶信息**)。

如果 $f(x)$ 足够光滑, 我们可以利用 $f(x)$ 的二阶信息改进下降方向, 以加速算法的迭代。

- 1 经典牛顿法
- 2 收敛性分析
- 3 修正牛顿法
- 4 非精确牛顿法
- 5 应用举例: 逻辑回归模型

# 经典牛顿法

- 对于可微二次函数 $f(x)$ , 考虑目标函数 $f$ 在点 $x_k$ 的二阶泰勒近似

$$f(x^k + d^k) = f(x^k) + \nabla f(x^k)^T d^k + \frac{1}{2} (d^k)^T \nabla^2 f(x^k) d^k + o(\|d^k\|^2).$$

忽略高阶项 $o(\|d^k\|^2)$ , 并将等式右边视作 $d^k$ 的函数并极小化, 得

$$\nabla^2 f(x^k) d^k = -\nabla f(x^k). \quad (1)$$

方程(1)被称为牛顿方程,  $d^k$ 被称为牛顿方向.

- 若 $\nabla^2 f(x^k)$ 非奇异, 可构造迭代格式

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

当步长 $\alpha_k = 1$ 时迭代格式(2)被称为经典牛顿法.

- 1 经典牛顿法
- 2 收敛性分析**
- 3 修正牛顿法
- 4 非精确牛顿法
- 5 应用举例: 逻辑回归模型

# 经典牛顿法的收敛性

## 定理

**经典牛顿法的收敛性** 假设  $f$  二阶连续可微, 且存在  $x^*$  的一个邻域  $N_\delta(x^*)$  及常数  $L > 0$  使得

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq L\|x - y\|, \quad \forall x, y \in N_\delta(x^*)$$

如果  $f(x)$  满足  $\nabla f(x^*) = 0, \nabla^2 f(x^*) \succ 0$ , 则对于迭代格式(2)有:

- 如果初始点离  $x^*$  足够近, 则迭代点列  $\{x^k\}$  收敛到  $x^*$ ;
- $\{x^k\}$   $Q$ -二次收敛到  $x^*$ ;
- $\{\|\nabla f(x^k)\|\}$   $Q$ -二次收敛到 0.

## 定理证明

根据经典牛顿法定义以及  $\nabla f(x^*) = 0$ , 得

$$\begin{aligned}x^{k+1} - x^* &= x^k - \nabla^2 f(x^k)^{-1} \nabla f(x^k) - x^* \\ &= \nabla^2 f(x^k)^{-1} [\nabla^2 f(x^k) (x^k - x^*) - (\nabla f(x^k) - \nabla f(x^*))],\end{aligned}\quad (3)$$

注意到

$$\nabla f(x^k) - \nabla f(x^*) = \int_0^1 \nabla^2 f(x^k + t(x^* - x^k)) (x^k - x^*) dt,$$

由此

$$\begin{aligned}& \|\nabla^2 f(x^k) (x^k - x^*) - (\nabla f(x^k) - \nabla f(x^*))\| \\ &= \left\| \int_0^1 [\nabla^2 f(x^k + t(x^* - x^k)) - \nabla^2 f(x^k)] (x^k - x^*) dt \right\| \\ &\leq \int_0^1 \|\nabla^2 f(x^k + t(x^* - x^k)) - \nabla^2 f(x^k)\| \|x^k - x^*\| dt \\ &\leq \|x^k - x^*\|^2 \int_0^1 Lt dt \text{ (Lip. 连续性)} = \frac{L}{2} \|x^k - x^*\|^2.\end{aligned}\quad (4)$$

注意到,  $\exists r > 0$ , 当  $\|x - x^*\| \leq r$  时有  $\|\nabla^2 f(x)^{-1}\| \leq 2 \|\nabla^2 f(x^*)^{-1}\|$  成立 (请思考为何?), 故结合(3)及(4), 得到

$$\begin{aligned}
 & \|x^{k+1} - x^*\| \\
 & \leq \|\nabla^2 f(x^k)^{-1}\| \|\nabla^2 f(x^k)(x^k - x^*) - (\nabla f(x^k) - \nabla f(x^*))\| \\
 & \leq \|\nabla^2 f(x^k)^{-1}\| \cdot \frac{L}{2} \|x^k - x^*\|^2 \\
 & \leq L \|\nabla^2 f(x^*)^{-1}\| \|x^k - x^*\|^2.
 \end{aligned} \tag{5}$$

当初始点  $x^0$  满足  $\|x^0 - x^*\| \leq \min \left\{ \delta, r, \frac{1}{2L \|\nabla^2 f(x^*)^{-1}\|} \right\}$  时, 迭代点列一直处于邻域  $N_{\delta}(x^*)$  中, 故  $\{x^k\}$   $\mathbf{Q}$ -二次收敛到  $x^*$ .



另一方面,由牛顿方程(1)可知

$$\begin{aligned}\|\nabla f(x^{k+1})\| &= \|\nabla f(x^{k+1}) - \nabla f(x^k) - \nabla^2 f(x^k) d^k\| \\ &= \left\| \int_0^1 \nabla^2 f(x^k + td^k) d^k dt - \nabla^2 f(x^k) d^k \right\| \\ &\leq \int_0^1 \|\nabla^2 f(x^k + td^k) - \nabla^2 f(x^k)\| \|d^k\| dt \\ &\leq \frac{L}{2} \|d^k\|^2 \leq \frac{1}{2} L \left\| \nabla^2 f(x^k)^{-1} \right\|^2 \|\nabla f(x^k)\|^2 \\ &\leq 2L \left\| \nabla^2 f(x^*)^{-1} \right\|^2 \|\nabla f(x^k)\|^2.\end{aligned}$$

这证明梯度的范数Q-二次收敛到0.

# 收敛速度分析

牛顿法收敛速度非常快, 但实际使用中会存在若干限制因素.

- 初始点 $x^0$  需要距离最优解充分近(因为局部收敛性)  
应用时常以梯度类算法先求得较低精度的解, 后用牛顿法加速.
- $\nabla^2 f(x^*)$  需正定, 半正定条件下可能退化到Q-线性收敛.
- $\nabla^2 f$  的条件数较高时, 将对初值的选择作出较严苛的要求.

- 1 经典牛顿法
- 2 收敛性分析
- 3 修正牛顿法**
- 4 非精确牛顿法
- 5 应用举例: 逻辑回归模型

# 修正牛顿法

经典牛顿法的基本格式如下:

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

除了计算、存储代价昂贵之外,经典牛顿法还存在如下问题:

- 海瑟矩阵可能非正定,导致牛顿方向其实并非下降方向;
- 初始点离最优值点较远时候迭代不稳定(步长固定),因而算法可能不收敛.

为提高算法的稳定性,从以上两方面考虑,应该:

- 对 $\nabla^2 f(x)$ 进行修正,使其正定(所有特征值大于0);
- 用线搜索确定步长来增加算法的稳定性(Wolfe, Goldstein, Armijo).

综上考虑,我们提出下面带线搜索的牛顿方法,并称其为**修正牛顿方法**.

# 带线搜索的修正牛顿法

---

## Algorithm 1 带线搜索的修正牛顿法

---

- 1: 给定初始点 $x^0$ .
  - 2: **for**  $k = 0, 1, 2, \dots$  **do**
  - 3: 确定矩阵 $E^k$  使得矩阵 $B^k \stackrel{\text{def}}{=} \nabla^2 f(x^k) + E^k$  正定且条件数较小.
  - 4: 求解修正的牛顿方程 $B^k d^k = -\nabla f(x^k)$  得方向 $d^k$ .
  - 5: 使用任意一种线搜索准则确定步长 $\alpha_k$ .
  - 6: 更新 $x^{k+1} = x^k + \alpha_k d^k$ .
  - 7: **end for**
- 

在上述算法中,  $B^k$  (即 $E^k$ ) 的选取较为关键, 需要注意:

- $B^k$  应具有较低的条件数(原因见[收敛性定理](#));
- 对 $\nabla^2 f(x)$  的改动较小, 以保存二阶信息;
- $B^k$  本身的计算代价不应太高.

# 修正牛顿法的全局收敛性

## 定理

**修正牛顿法全局收敛性定理** 令  $f$  在开集  $D$  上二阶连续可微, 且初始点  $x^0$  满足  $\{x \in D : f(x) \leq f(x_0)\}$  为紧集. 若算法 1 中  $B^k$  的条件数上界存在, 即

$$\kappa(B_k) = \|B_k\| \|B_k^{-1}\| \leq C, \quad \exists C > 0, \forall k = 0, 1, 2, \dots \quad (6)$$

则

$$\lim_{k \rightarrow \infty} \nabla f(x_k) = 0,$$

即算法具有全局收敛性.

上述定理的证明冗长, 读者可具体参考: Newton's method, in Studies in Numerical Analysis, vol. 24 of MAA Studies in Mathematics, The Mathematical Association of America, 1984, pp. 29–82.

## 修正矩阵 $E^k$ 的显式选取

进一步地, 在修正步长后, 为还使  $B^k$  正定, 我们修正海瑟矩阵的特征值. 首先做特征分解

$$\nabla^2 f(x_k) = Q\Lambda Q^T,$$

其中,  $Q$  为正交矩阵,  $\Lambda$  为对角矩阵.

取  $E^k = \tau_k I$ , 即单位阵的常数倍, 代入上式, 则有

$$B^k = Q(\Lambda + \tau_k I)Q^T.$$

当正数  $\tau_k$  足够大的时候, 可保证  $B^k$  正定, 但  $d^k$  会接近负梯度方向, 退化为一阶方法.

一种简单的技巧是对  $\tau_k$  做截断, 即

$$\tau_k = \max\{0, \delta - \lambda_{\min}(\nabla^2 f(x^k))\}, \quad (\text{给定 } \delta > 0)$$

然而, 此时  $\lambda_{\min}(\nabla^2 f(x^k))$  通常难以计算. 我们通常使用 Cholesky 分解试探性取  $\tau_k$ , 从而避免这个问题.

下面利用Cholesky分解给出另一种算法 ( $a_{ii}$  表示  $\nabla^2 f(x^k)$  的对角元素)

---

### Algorithm 2 Cholesky分解(增加数量矩阵)

---

```
1: 选取  $\beta, \sigma$ . (如  $\beta = 10^{-3}, \sigma = 2$ )
2: if  $\min_i \{a_{ii}\} > 0$  then
3:    $\tau_0 = 0$ 
4: else
5:    $\tau_0 = -\min_i \{a_{ii}\} + \beta$ 
6: end if
7: for  $t = 0, 1, 2, \dots$  do
8:   尝试用Cholesky算法计算:  $LL^T = \nabla^2 f(x^k) + \tau_t I$ 
9:   if Cholesky算法成功运行 then
10:    终止循环并返回  $L$ 
11:   else
12:     $\tau_{t+1} = \max \{\sigma \tau_t, \beta\}$ 
13:   end if
14: end for
```

---

缺陷:可能需要多次的试验,而每一步都对  $\nabla^2 f(x^k) + \tau_t I$  做分解的计算代价大(可考虑较大的  $\sigma$ , 如  $\sigma = 10$ )



## 修正矩阵 $E^k$ 的隐式选取

直接对 $\nabla^2 f(x^k)$ 进行Cholesky分解可能会失败, 因此考虑修正分解算法.  
回顾Cholesky分解的定义:

$$A = LDL^T,$$

其中,  $A = (a_{ij})$  对称正定,  $L = (l_{ij})$  是对角线元素均为1的下三角矩阵,  
 $D = \text{Diag}(d_1, d_2, \dots, d_n)$  是对角矩阵且对角线元素均为正.

---

### Algorithm 3 标准Cholesky分解

---

- 1: 给定对称矩阵 $A = (a_{ij})$ .
- 2: **for**  $j = 1, 2, \dots, n$  **do**
- 3:      $c_{jj} = a_{jj} - \sum_{s=1}^{j-1} d_s l_{js}^2$
- 4:      $d_j = c_{jj}$
- 5:     **for**  $i = j + 1, j + 2, \dots, n$  **do**
- 6:          $c_{ij} = a_{ij} - \sum_{s=1}^{j-1} d_s l_{is} l_{js}$
- 7:          $l_{ij} = c_{ij} / d_j$ .
- 8:     **end for**
- 9: **end for**

## 修正Cholesky分解

修正的目标:使  $d_i$  为正数,且  $D$  和  $L$  中元素值不太大.

具体地, 选取正的参数  $\delta, \beta$ , 并要求在算法(3)中第  $j$  个外循环满足:

$$d_j \geq \delta, \quad |l_{ij}\sqrt{d_j}| \leq \beta, \quad i = j + 1, j + 2, \dots, n \quad (7)$$

考虑下面的更新方式:

$$d_j = \max \left\{ |c_{jj}|, \left( \frac{\theta_j}{\beta} \right)^2, \delta \right\}, \quad \theta_j = \max_{i>j} |c_{ij}|$$

注意到算法(3)中有  $c_{ij} = l_{ij}d_j$ , 根据下式可知条件(7)成立:

$$|l_{ij}\sqrt{d_j}| = \frac{|c_{ij}|}{\sqrt{d_j}} \leq \frac{|c_{ij}|\beta}{\theta_j} \leq \beta, \quad \forall i > j$$

- $\theta_j$  可以在之前  $d_j$  计算, 因为  $c_{ij}$  在算法(3)内循环的计算不包含  $d_j$ .
- 修正算法其实是在计算  $\nabla^2 f(x^k) + E^k$  的Cholesky分解 ( $E^k$  为对角元素非负的对角阵). 当  $\nabla^2 f(x^k)$  正定且条件数足够小时有  $E^k = 0$ .
- 可证明, 由修正算法得到的  $B_k$  的条件数存在上界, 即满足(6).

- 1 经典牛顿法
- 2 收敛性分析
- 3 修正牛顿法
- 4 非精确牛顿法**
- 5 应用举例: 逻辑回归模型

# 非精确牛顿法

当变量维数很大时，牛顿法可能有如下困难：

- 海瑟矩阵 $\nabla^2 f(x)$ 本身的计算、存储存在困难；
- 对 $\nabla^2 f(x)$ 求逆或者做Cholesky分解的代价很高。

## 非精确牛顿法

- 使用迭代法(如共轭梯度法)求解牛顿方程，在一定的精度下**提前停机**，以提高求解效率。
- 引入向量 $r^k$ 来表示残差，将上述方程记为

$$\nabla^2 f(x^k) d^k = -\nabla f(x^k) + r^k. \quad (8)$$

因此终止条件可设置为

$$\|r^k\| \leq \eta_k \|\nabla f(x^k)\|. \quad (9)$$

- 不同的 $\{\eta_k\}$ 将导致不同的精度要求，使算法有不同的收敛速度。

# 收敛性分析

我们叙述非精确牛顿法的局部收敛定理.

## 定理

**非精确牛顿法的收敛定理** 设函数 $f(x)$ 二阶连续可微, 且 $\nabla^2 f(x^*)$ 正定, 则在非精确牛顿法中,

(1) 若 $\exists t < 1$ , 使得 $\eta_k$  满足 $0 < \eta_k < t, k = 1, 2, \dots$ , 且起始点 $x_0$  充分靠近 $x^*$  并迭代最终收敛到 $x^*$ , 则梯度 $\nabla f(x^k)$  以 $Q$ -线性收敛速度收敛;

(2) 若 $\lim_{k \rightarrow \infty} \eta_k = 0$  成立, 则梯度 $\nabla f(x^k)$  以 $Q$ -超线性收敛速度收敛;

(3) 若(1)或(2)成立, 且 $\nabla^2 f$ 在 $x^*$ 附近 $Lip.$ 连续,  $\eta_k = O(\|\nabla f(x^k)\|)$ , 则梯度 $\nabla f(x^k)$  以 $Q$ -二次收敛速度收敛.

## 定理证明

我们只给出一个不严格的证明, 请读者主要体会证明思路.

注意到  $\nabla^2 f(x)$  在  $x^*$  处正定, 在  $x^*$  附近连续, 故存在正常数  $L$ , 使得

$$\left\| (\nabla^2 f(x^k))^{-1} \right\| \leq L, \quad (\forall x^k \text{ 同 } x^* \text{ 足够接近})$$

代入(8), 得(第二个不等式用到了  $\eta_k < 1$ )

$$\|d^k\| \leq L (\|\nabla f(x^k)\| + \|r^k\|) \leq 2L \|\nabla f(x^k)\|.$$

利用Taylor展式和  $\nabla^2 f$  的连续性, 得到

$$\begin{aligned} \nabla f(x^{k+1}) &= \nabla f(x^k) + \nabla^2 f(x^k)d^k + \int_0^1 [\nabla^2 f(x^k + td^k) - \nabla^2 f(x^k)] d^k dt \\ &= \nabla f(x^k) + \nabla^2 f(x^k)d^k + o(\|d^k\|) \\ &= \nabla f(x^k) - (\nabla f(x^k) - r^k) + o(\|\nabla f(x^k)\|) \\ &= r^k + o(\|\nabla f(x^k)\|). \end{aligned}$$

## 定理证明

上式中两边取范数, 结合精度控制式(9), 得到

$$\|\nabla f(x^{k+1})\| \leq \eta_k \|\nabla f(x^k)\| + o(\|\nabla f(x^k)\|) \leq (\eta_k + o(1)) \|\nabla f(x^k)\|.$$

当 $x^k$ 足够接近 $x^*$ 时, $o(1)$ 项可被 $(1-t)/2$ 控制, 则

$$\|\nabla f(x^{k+1})\| \leq (\eta_k + (1-t)/2) \|\nabla f(x^k)\| \leq \frac{1+t}{2} \|\nabla f(x^k)\|.$$

由于 $t < 1$ , 故梯度 $\nabla f(x^k)$ 以 $Q$ -线性收敛速度收敛.

## 推论

从以上证明过程可以看出如下的结果:

$$\frac{\|\nabla f(x^{k+1})\|}{\|\nabla f(x^k)\|} \leq \eta_k + o(1).$$

若  $\lim_{k \rightarrow \infty} \eta_k = 0$  成立, 可有 Q-超线性收敛的结论

$$\lim_{k \rightarrow \infty} \frac{\|\nabla f(x^{k+1})\|}{\|\nabla f(x^k)\|} = 0.$$

若  $\nabla^2 f$  在  $x^*$  附近 Lip. 连续, 令  $\eta_k = O(\|\nabla f(x^k)\|)$ , 可有二次收敛的结论

$$\|\nabla f(x^{k+1})\| = O\left(\|\nabla f(x^k)\|^2\right).$$

因此, 在实际应用时:

- 取  $\eta_k = \min\left(0.5, \sqrt{\|\nabla f(x^k)\|}\right)$ , 可成立局部的超线性收敛;
- 取  $\eta_k = \min\left(0.5, \|\nabla f(x^k)\|\right)$ , 可成立局部的二次收敛.



## 线搜索Newton-CG法

最后我们讨论求解牛顿方程的方法。

一种广泛应用的方法是线搜索Newton-CG法, 它利用共轭梯度迭代求解牛顿方程。

- 由于共轭梯度法要求被解方程的系数矩阵正定, 但 $\nabla^2 f$ 可能非正定, 因此在迭代时需要辨别 $\nabla^2 f$ 的正定性以保证算法有效。

具体算法在下一面给出。其中 $B_k$ 表示 $\nabla^2 f(x^k)$ ;  $\{z_j\}$ 表示CG法从零向量开始的迭代序列, 最终收敛到牛顿方向。

我们采用的迭代终止条件是 $\eta_k = \min\left(0.5, \sqrt{\|\nabla f(x^k)\|}\right)$ 。

CG法解牛顿方程的优势是只用到海瑟矩阵-向量积而无需求出 $\nabla^2 f$ , 故可以被设计成*Hessian-free*方法, 在大规模问题下解决Hessian矩阵计算或存储的困难。

# 线搜索Newton-CG法:算法

## Algorithm 4 线搜索Newton-CG法

- 1: 给定初始点 $x^0$ .
- 2: **for**  $k = 0, 1, 2, \dots$  **do**
- 3:   令 $\epsilon_k = \eta_k \|\nabla f(x^k)\|$ ,  $z_0 = 0, r_0 = \nabla f(x^k), p_0 = -r_0 = -\nabla f(x^k)$ .
- 4:   **for**  $j = 0, 1, 2, \dots$  **do**
- 5:     **if**  $p_j^T B_k p_j \leq 0$  (判断是否是负曲率方向) **then**
- 6:       终止循环. $j = 0$  则返回 $d^k = -\nabla f$ , 否则返回 $d^k = z_j$ .
- 7:     **end if**
- 8:      $\alpha_j = r_j^T r_j / p_j^T B_k p_j$ ,  $z_{j+1} = z_j + \alpha_j p_j$ ,  $r_{j+1} = r_j + \alpha_j B_k p_j$
- 9:     **if**  $r_j < \epsilon_k$  (判断是否达到收敛条件) **then**
- 10:       终止循环. 返回 $d^k = z_{j+1}$ .
- 11:     **end if**
- 12:      $\beta_{j+1} = r_{j+1}^T r_{j+1} / r_j^T r_j$ ,  $d_{j+1} = -r_{j+1} + \beta_{j+1} d_j$
- 13:   **end for**
- 14:   线搜索确定步长 $\alpha_k$ (符合条件则取 $\alpha_k = 1$ ), 更新 $x^{k+1} = x^k + \alpha_k d^k$ .
- 15: **end for**

- 1 经典牛顿法
- 2 收敛性分析
- 3 修正牛顿法
- 4 非精确牛顿法
- 5 应用举例: 逻辑回归模型**

# 逻辑回归模型

考虑二分类的逻辑回归模型

$$\min_x \ell(x) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^m \ln(1 + \exp(-b_i a_i^T x)) + \lambda \|x\|_2^2.$$

为使用牛顿法，需要计算目标函数的梯度与海瑟矩阵：

$$\begin{aligned} \nabla \ell(x) &= \frac{1}{m} \sum_{i=1}^m \frac{1}{1 + \exp(-b_i a_i^T x)} \cdot \exp(-b_i a_i^T x) \cdot (-b_i a_i) + 2\lambda x \\ &= -\frac{1}{m} \sum_{i=1}^m (1 - p_i(x)) b_i a_i + 2\lambda x, \end{aligned}$$

其中  $p_i(x) = \frac{1}{1 + \exp(-b_i a_i^T x)}$ .

# 逻辑回归模型

进一步对 $\nabla\ell(x)$ 求导, 成立

$$\begin{aligned}\nabla^2\ell(x) &= \frac{1}{m} \sum_{i=1}^m b_i \cdot \nabla p_i(x) a_i^T + 2\lambda I \\ &= \frac{1}{m} \sum_{i=1}^m b_i \frac{-1}{(1 + \exp(-b_i a_i^T x))^2} \cdot \exp(-b_i a_i^T x) \cdot (-b_i a_i a_i^T) + 2\lambda I \\ &= \frac{1}{m} \sum_{i=1}^m (1 - p_i(x)) p_i(x) a_i a_i^T + 2\lambda I \quad (b_i^2 = 1).\end{aligned}$$

## 逻辑回归模型

引入矩阵  $A = [a_1, a_2, \dots, a_m]^T \in \mathbb{R}^{m \times n}$ , 向量  $b = (b_1, b_2, \dots, b_m)^T$ , 以及

$$p(x) = (p_1(x), p_2(x), \dots, p_m(x))^T,$$

则可重写梯度和海瑟矩阵为

$$\nabla \ell(x) = -\frac{1}{m} A^T (b - b \odot p(x)) + 2\lambda x,$$

$$\nabla^2 \ell(x) = \frac{1}{m} A^T W(x) A + 2\lambda I,$$

其中  $W(x)$  为由  $\{p_i(x)(1-p_i(x))\}_{i=1}^m$  生成的对角矩阵.

则最终牛顿法迭代格式可以写作:

$$x^{k+1} = x^k + \left( \frac{1}{m} A^T W(x^k) A + 2\lambda I \right)^{-1} \left( \frac{1}{m} A^T (b - b \odot p(x^k)) - 2\lambda x^k \right).$$

## 逻辑回归模型

我们得到了牛顿法的迭代格式, 因此可以调用牛顿法直接求解逻辑回归问题. 正如我们对牛顿方程处理思路的不同, 若变量规模不大, 则可尝试利用正定矩阵的Cholesky分解求解牛顿方程; 若变量规模较大, 则可以使用共轭梯度法对方程进行不精确的求解.

我们使用LIBSVM网站的数据集(具体数据集见下表), 对不同的数据集均调用非精确CG-牛顿法求解, 设置精度条件为

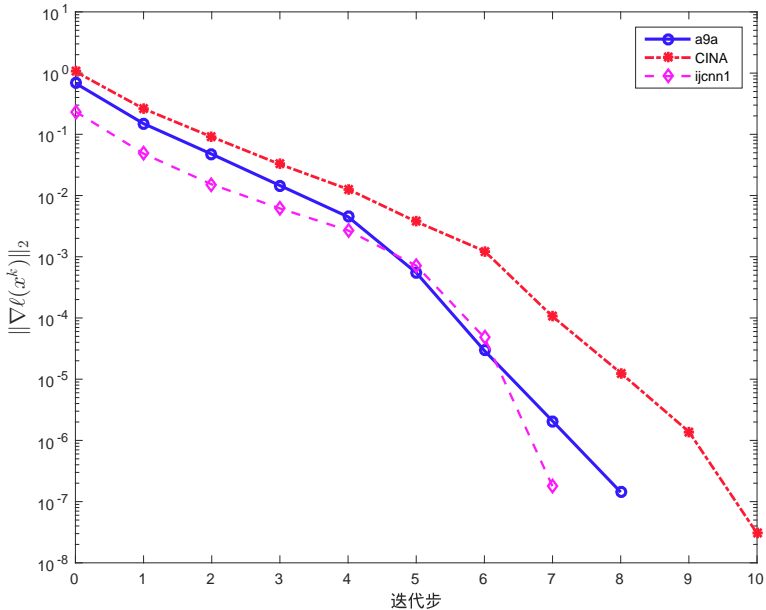
$$\|\nabla^2 \ell(x^k) d^k + \nabla \ell(x^k)\|_2 \leq \min \left\{ \|\nabla \ell(x^k)\|_2^2, 0.1 \|\nabla \ell(x^k)\|_2 \right\},$$

Table: LIBSVM数据集(部分)

名称	维数	
	$m$	$n$
<i>a9a</i>	16281	122
<i>ijcnn1</i>	91701	22
<i>CINA</i>	3206	132

在数据集集中进行算法测试, 数值结果如下图所示. 从图中可以看到, 精确解附近梯度范数具有Q-超线性收敛性.

# 数值结果





- [1] Nocedal J, Wright S. Numerical optimization[M]. Springer Science and Business Media, 2006.
- [2] Hu J, Milzarek A, Wen Z, et al. Adaptive quadratically regularized Newton method for Riemannian optimization[J]. SIAM Journal on Matrix Analysis and Applications, 2018, 39(3): 1181-1207.