次梯度算法

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：本教案由朱桢源协助准备

回顾：梯度下降算法

设 $f(x)$ 是可微凸函数且 $\mathbf{dom}\, f = \mathbb{R}^n$，考虑如下问题：

$$\min_x f(x).$$

- 梯度下降法：选择初始点 $x^0 \in \mathbb{R}^n$，然后重复：

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k), \quad k = 0, 1, 2, \ldots$$

其中 $\alpha_k > 0$为步长，可取为固定常数或者通过线搜索确定.

- 若 $\nabla f(x)$ 利普西茨连续，则梯度下降法的收敛速度是 $\mathcal{O}\left(\frac{1}{k}\right)$.

如果 $f(x)$不可微呢?

提纲

非光滑优化的例子

- 极小极大问题：

$$\min_{x\in X}\ \max_{1\le i\le m}\ f_i(x)$$

- 求解非线性方程组：

$$f_i(x)=0,\quad i=1,\cdots,m$$

可以把它化为一个极小化问题：

$$\min_{x\in X}\ \|\,(f_1(x),\cdots,f_m(x))\,\|$$

特别地，$\|\cdot\|=\|\cdot\|_1$ 对应 L_1 极小化问题，$\|\cdot\|=\|\cdot\|_\infty$对应切比雪夫近似问题.

- LASSO问题：

$$\min_{x}\ \|Ax-b\|^2+\mu\|x\|_1$$

梯度下降法失败的例子

考虑函数 $f:\mathbb{R}^2\to\mathbb{R}^1, x=(u,v)^{\mathrm{T}}$，

$$f(x)=\max\left[\frac{1}{2}u^2+(v-1)^2,\frac{1}{2}u^2+(v+1)^2\right].$$

- 假设迭代点 x^k 的形式为

$$x^k=\begin{pmatrix}2\left(1+|\epsilon_k|\right)\\ \epsilon_k\end{pmatrix},\quad \text{其中}\,\epsilon_k\neq 0.$$

- 可以计算迭代点 x^k 处的梯度：

$$\nabla f\left(x^k\right)=\begin{pmatrix}2\left(1+|\epsilon_k|\right)\\ 2\left(1+|\epsilon_k|\right)t_k\end{pmatrix}=2\left(1+|\epsilon_k|\right)\begin{pmatrix}1\\ t_k\end{pmatrix},$$

其中 $t_k=\operatorname{sign}\left(\epsilon_k\right)$.

梯度下降法失败的例子

下面我们考虑直接用梯度下降法进行迭代.

- 在负梯度方向$-\nabla f\left(x^k\right)$上做精确线搜索，可得

$$x^{k+1}=x^k+\alpha_k\left(-\nabla f\left(x^k\right)\right)=\begin{bmatrix}2\left(1+\left|\epsilon_k\right|/3\right)\\ -\epsilon_k/3\end{bmatrix}=\begin{bmatrix}2\left(1+\left|\epsilon_{k+1}\right|\right)\\ \epsilon_{k+1}\end{bmatrix}$$

  其中$\epsilon_{k+1}=-\epsilon_k/3\neq 0$. 所以显然有$\epsilon_k\to 0$.

- 给定一个初始点$x^0=(2+2|\delta|,\delta)^{\mathrm{T}}$，我们有$x^k\to(2,0)^{\mathrm{T}}$.
- 然而$(2,0)^{\mathrm{T}}$并不是稳定点.
- 这表明对非光滑问题直接使用梯度法可能会收敛到一个非稳定点.

提纲

问题设定

假设$f(x)$为凸函数，但不一定可微，考虑如下问题：

$$\min_x f(x)$$

- 一阶充要条件：

$$x^* \text{是一个全局极小点} \quad \Leftrightarrow \quad 0 \in \partial f(x^*)$$

- 因此可以通过计算凸函数的次梯度集合中包含0的点来求解其对应的全局极小点.

次梯度算法结构

为了极小化一个不可微的凸函数f，可类似梯度法构造如下次梯度算法的迭代格式：

$$x^{k+1} = x^k - \alpha_k g^k, \quad g^k \in \partial f(x^k),$$

其中$\alpha_k > 0$为步长. 它通常有如下四种选择：

1. 固定步长$\alpha_k = \alpha$；
2. 固定$\|x^{k+1} - x^k\|$，即$\alpha_k \|g^k\|$为常数；
3. 消失步长$\alpha_k \to 0$且$\sum_{k=0}^{\infty} \alpha_k = +\infty$；
4. 选取α_k使其满足某种线搜索准则.

下面我们讨论在不同步长取法下次梯度算法的收敛性质.

提纲

假设条件

(1) f为凸函数；

(2) f至少存在一个有限的极小值点x^*，且$f(x^*) > -\infty$；

(3) f为利普希茨连续的，即

$$|f(x) - f(y)| \leq G\|x - y\|, \quad \forall\, x, y \in \mathbb{R}^n,$$

其中$G > 0$为利普希茨常数.

我们下面证明这等价于$f(x)$的次梯度是有界的，即

$$\|g\| \leq G, \quad \forall\, g \in \partial f(x), x \in \mathbb{R}^n.$$

证明：

- 充分性：假设$\|g\|_2 \leq G, \forall\, g \in \partial f(x)$；取$g_y \in \partial f(y), g_x \in \partial f(x)$:

$$g_x^{\mathrm{T}}(x-y) \geq f(x) - f(y) \geq g_y^{\mathrm{T}}(x-y)$$

再由柯西不等式

$$G\|x-y\|_2 \geq f(x) - f(y) \geq -G\|x-y\|_2$$

- 必要性：反设存在x和$g \in \partial f(x)$，使得$\|g\|_2 > G$；取$y = x + \frac{g}{\|g\|_2}$

$$\begin{aligned} f(y) &\geq f(x) + g^{\mathrm{T}}(y-x) \\ &= f(x) + \|g\|_2 \\ &> f(x) + G \end{aligned}$$

这与$f(x)$是G-利普希茨连续的矛盾.

收敛性分析

- 次梯度方法不是一个下降方法，即无法保证$f(x^{k+1}) < f(x^k)$；
- 收敛性分析的关键是分析$f(x)$历史迭代的最优点所满足的性质.
- 设x^*是$f(x)$的一个全局极小值点，$f^* = f(x^*)$，根据迭代格式，

$$\begin{aligned}\left\|x^{i+1} - x^*\right\|^2 &= \left\|x^i - \alpha_i g^i - x^*\right\|^2 \\ &= \left\|x^i - x^*\right\|^2 - 2\alpha_i \left\langle g^i, x^i - x^*\right\rangle + \alpha_i^2 \left\|g^i\right\|^2 \\ &\leqslant \left\|x^i - x^*\right\|^2 - 2\alpha_i \left(f\left(x^i\right) - f^*\right) + \alpha_i^2 G^2\end{aligned}$$

- 结合$i = 0, \cdots, k$ 时相应的不等式，并定义$\hat{f}^k = \min_{0 \leqslant i \leqslant k} f\left(x^i\right)$：

$$\begin{aligned}2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\hat{f}^k - f^*\right) &\leqslant \left\|x^0 - x^*\right\|^2 - \left\|x^{k+1} - x^*\right\|^2 + G^2 \sum_{i=0}^{k} \alpha_i^2 \\ &\leqslant \left\|x^0 - x^*\right\|^2 + G^2 \sum_{i=0}^{k} \alpha_i^2\end{aligned}$$

不同步长下的收敛性

(1) 取$\alpha_i = t$为固定步长，则

$$\hat{f}^k - f^* \leq \frac{\|x^0 - x^*\|^2}{2kt} + \frac{G^2 t}{2};$$

- $\hat{f}^k$ 无法保证收敛性
- 当k 足够大时,$\hat{f}^k$ 近似为$G^2t/2$-次优的

(2) 取α_i使得$\|x^{i+1} - x^i\|$固定，即$\alpha_i\|g^i\| = s$为常数，则

$$\hat{f}^k - f^* \leq \frac{G\|x^0 - x^*\|^2}{2ks} + \frac{Gs}{2};$$

- $\hat{f}^k$ 无法保证收敛性
- 当k 足够大时,$\hat{f}^k$ 近似为$Gs/2$-次优的

不同步长下的收敛性

(3) 取α_i为消失步长，即$\alpha_i \to 0$且$\sum_{i=0}^{\infty} \alpha_i = +\infty$，则

$$\hat{f}^k - f^* \leq \frac{\|x^0 - x^*\|^2 + G^2 \sum_{i=0}^{k} \alpha_i^2}{2 \sum_{i=0}^{k} \alpha_i};$$

进一步可得$\hat{f}^k$收敛到f^*.

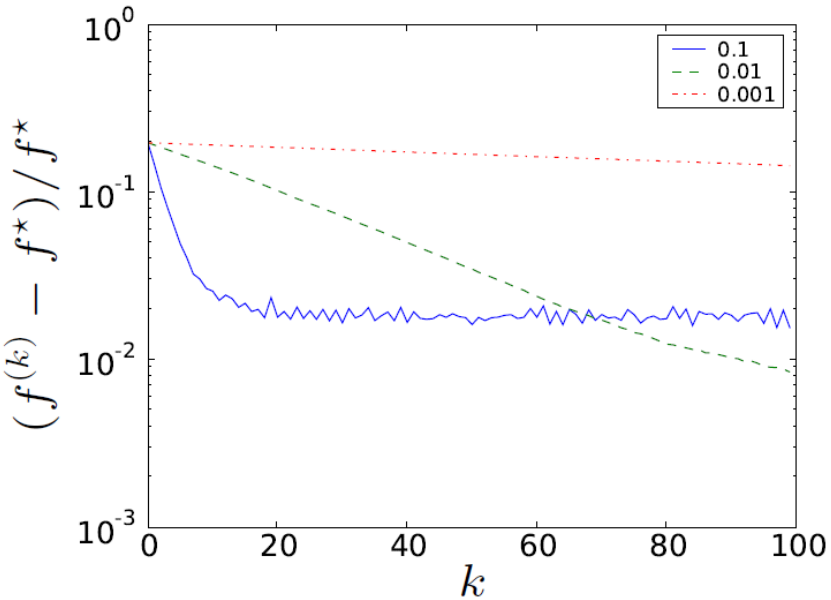
- 和梯度法不同，只有当α_k取消失步长时$\hat{f}^k$才具有收敛性.
- 一个常用的步长取法是$\alpha_k = \frac{1}{k}$.

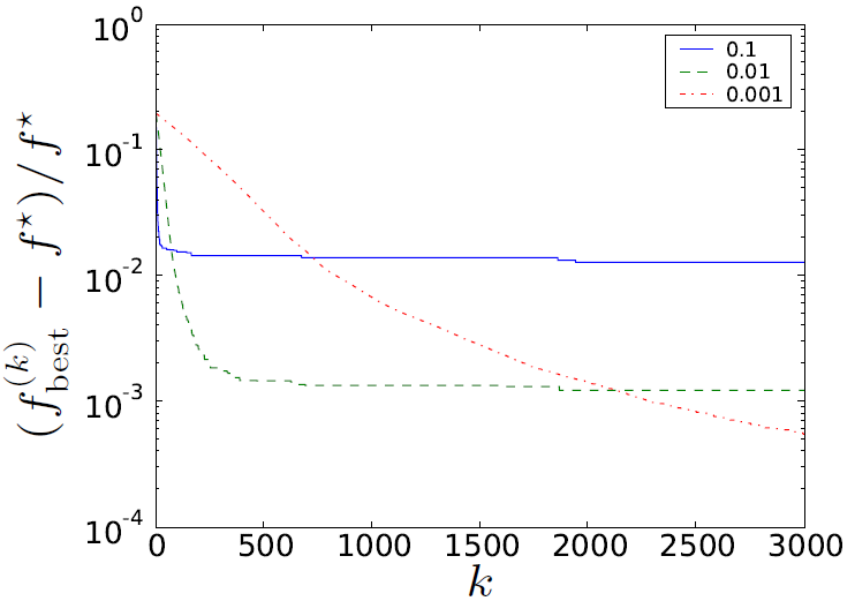
例：ℓ_1-范数极小化问题

$$\min \quad \|Ax-b\|_1 \quad (A \in \mathbb{R}^{500\times 100},\ b \in \mathbb{R}^{500})$$

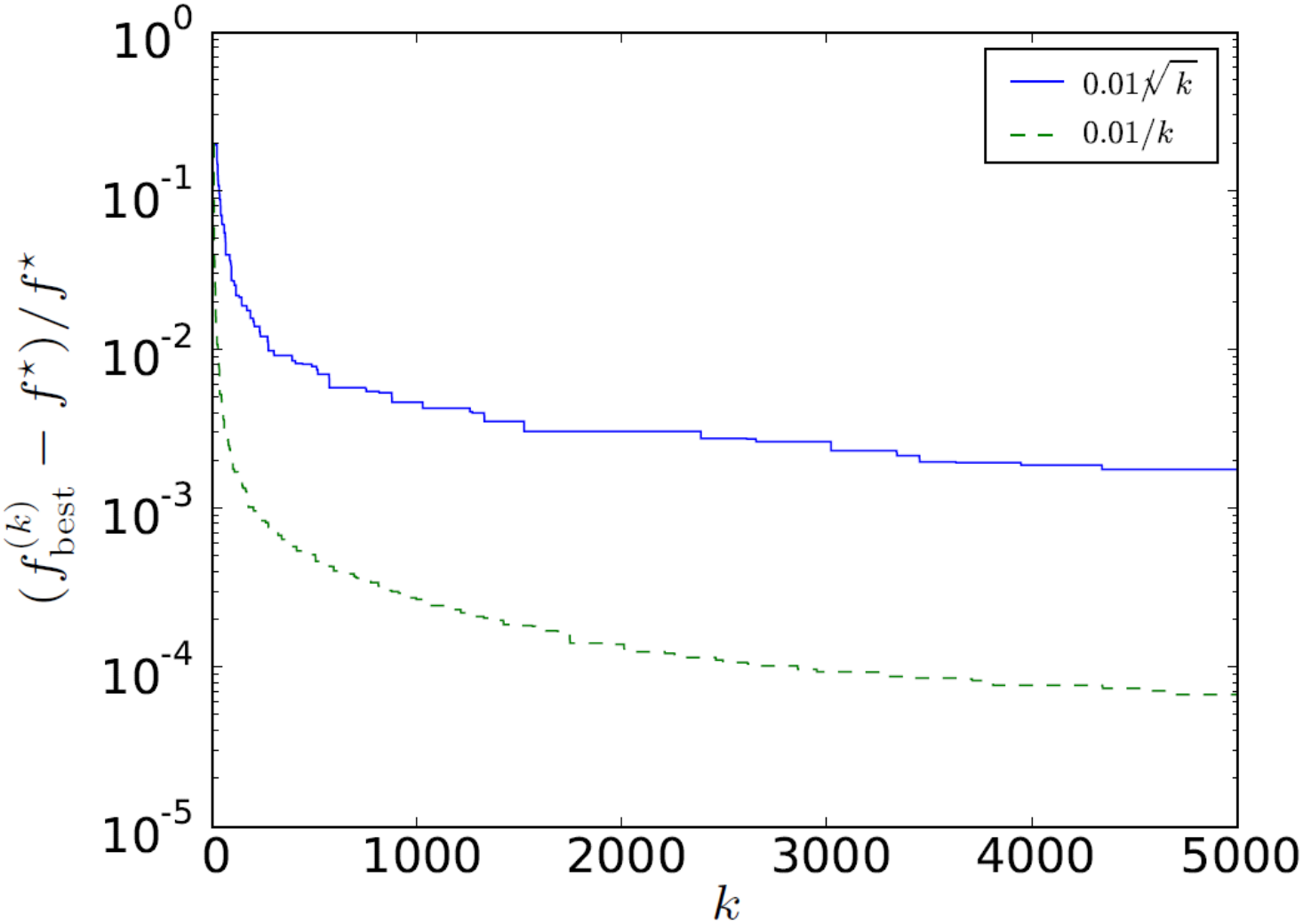
次梯度取为$A^{\mathrm{T}}\mathbf{sign}(Ax-b)$

- 第二类步长策略：$t_k = s/\|g^{(k-1)}\|_2$，$s = 0.1, 0.01, 0.001$





- 第三类步长策略：$t_k = 0.01/\sqrt{k}, t_k = 0.01/k$



固定迭代步数下的最优步长

- 假设$\left\|x^0 - x^*\right\| \leqslant R$，并且总迭代步数$k$是给定的，在固定步长下，

$$\hat{f}^k - f^* \leq \frac{\|x^0 - x^*\|^2}{2kt} + \frac{G^2 t}{2} \leq \frac{R^2}{2kt} + \frac{G^2 t}{2}.$$

- 由平均值不等式知当t满足$\frac{R^2}{2kt} = \frac{G^2 t}{2}$, 即 $t = \frac{R}{G\sqrt{k}}$时，右端达到最小.
- k 步后得到的上界是

$$\hat{f}^k - f^* \leqslant \frac{GR}{\sqrt{k}}$$

- 这表明在$k = O(1/\epsilon^2)$ 步迭代后可以得到$\hat{f}^k - f^* \leq \epsilon$ 的精度
- 类似地可证明第二类步长选取策略下，取$s = \frac{R}{\sqrt{k}}$，可得到估计

$$\hat{f}^k - f^* \leq \frac{GR}{\sqrt{k}}.$$

f^* 已知时的最优步长

- 第13页第一个不等式右端在

$$\alpha_i = \frac{f(x^i) - f^*}{\|g^i\|^2}$$

时取到极小.

- 这等价于

$$\frac{(f(x^i) - f^*)^2}{\|g^i\|^2} \leq \|x^i - x^*\|^2 - \|x^{i+1} - x^*\|^2.$$

- 递归地利用上式并结合$\|x^0 - x^*\| \leq R$和$\|g^i\| \leq G$，可以得到

$$\hat{f}^k - f^* \leq \frac{GR}{\sqrt{k}}.$$

练习：计算交集中的一点

为了寻找一个落在m个闭凸集$C_1, ..., C_m$的交集中的点：

$$\min \quad f(x) = \max\{d_1(x), \cdots, d_m(x)\}$$

其中$d_j(x) = \inf_{y \in C_j} \|x - y\|_2$表示点$x$到集合$C_j$的欧几里得距离

- 如果交集非空，$f^* = 0$
- 如果$g \in \partial d_j(\hat{x})$并且$C_j$是距离$\hat{x}$最远的集合，那么$g \in \partial f(\hat{x})$
- 次梯度$g \in \partial d_j(\hat{x})$：

$$g = 0 \text{ (如果 } \hat{x} \in C_j), \quad g = \frac{1}{d(\hat{x}, C_j)}(\hat{x} - P_j(\hat{x})) \text{ (如果 } \hat{x} \notin C_j)$$

其中$P_j(\hat{x})$是到C_j的投影；注意当$\hat{x} \notin C_j$时，$\|g\|_2 = 1$

最优步长选取下的次梯度法：

- 当$f^* = 0$ 且$\|g^{i-1}\|_2 = 1$ 时，最优步长是$\alpha_i = f(x^{i-1})$.
- 在第k 步迭代, 找到距离最远的集合C_j (此时$f(x^{k-1}) = d_j(x^{k-1})$)；取

$$\begin{aligned} x^k &= x^{k-1} - \frac{f(x^{k-1})}{d_j(x^{k-1})}(x^{k-1} - P_j(x^{k-1})) \\ &= P_j(x^{k-1}) \end{aligned}$$

- 是一种交替投影算法
- 每步迭代都将当前迭代点投影到最远的集合上
- 当$m = 2$ 时, 交替投影到两个集合上

提纲

例：LASSO 问题求解

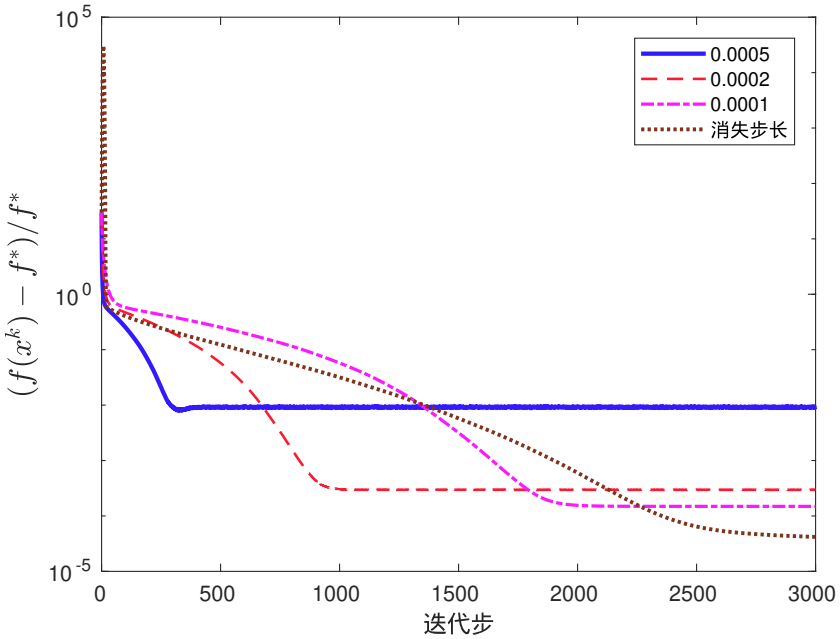
考虑LASSO 问题

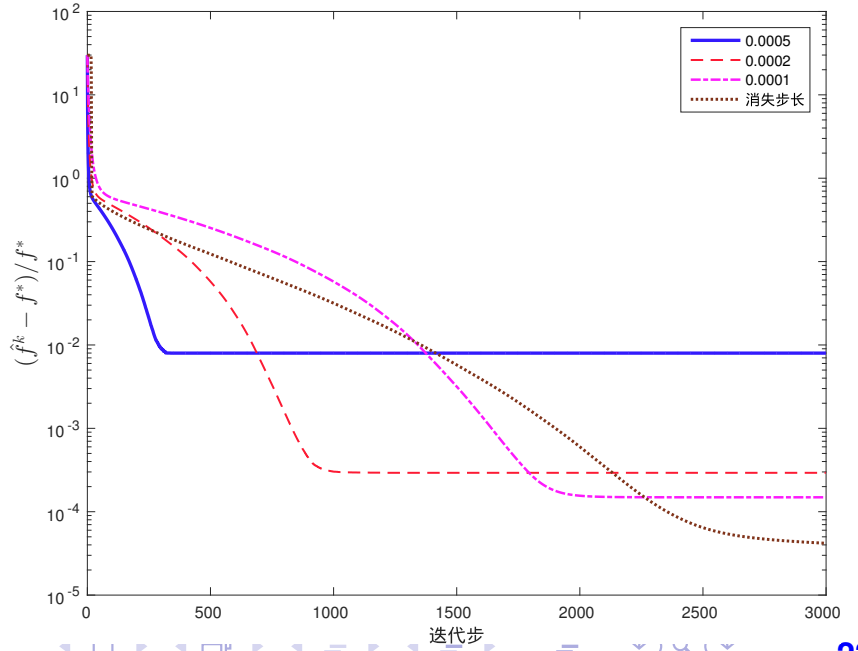
$$\min \quad f(x) = \frac{1}{2}\|Ax - b\|^2 + \mu\|x\|_1,$$

$f(x)$的一个次梯度为$g = A^{\mathrm{T}}(Ax - b) + \mu\mathrm{sign}(x)$, 其中$\mathrm{sign}(x)$是关于$x$逐分量的符号函数. 因此的次梯度算法为

$$x^{k+1} = x^k - \alpha_k(A^{\mathrm{T}}(Ax^k - b) + \mu\mathrm{sign}(x^k)),$$

步长α_k可选为固定步长或消失步长.





例：LASSO 问题求解

对于$\mu = 10^{-2}, 10^{-3}$，采用连续化次梯度算法进行求解. 若$\mu_t > \mu$，则取固定步长$\frac{1}{\lambda_{\max}(A^{\mathrm{T}}A)}$；若$\mu_t = \mu$，则取步长

$$\frac{1}{\lambda_{\max}(A^{\mathrm{T}}A) \cdot (\max\{k, 100\} - 99)},$$

其中k为迭代步数

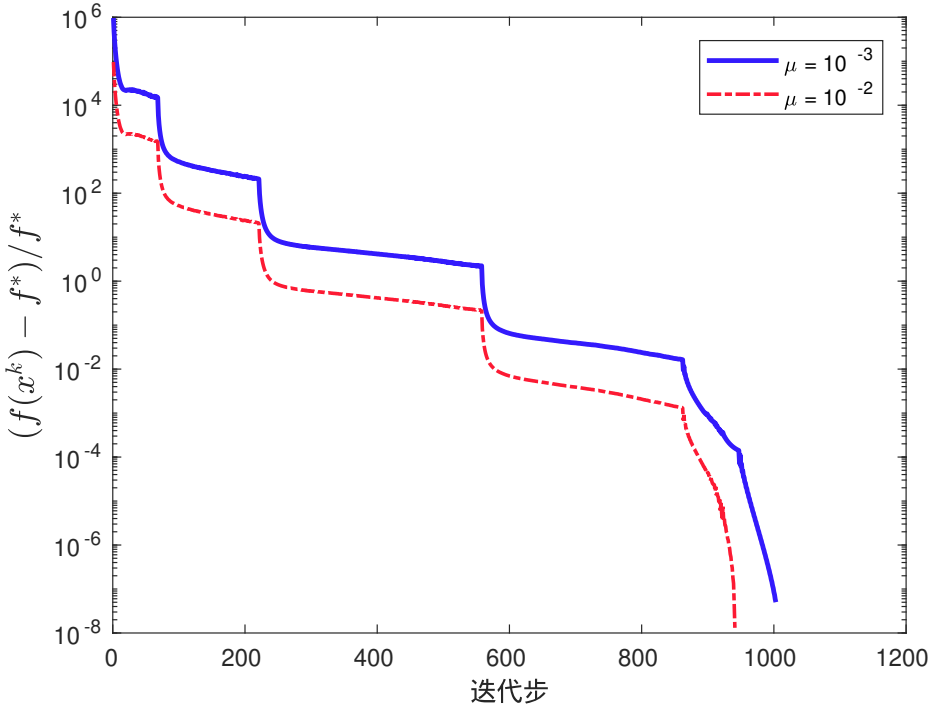


Figure: LASSO 问题在不同正则化参数下的求解结果

例：正定矩阵补全问题

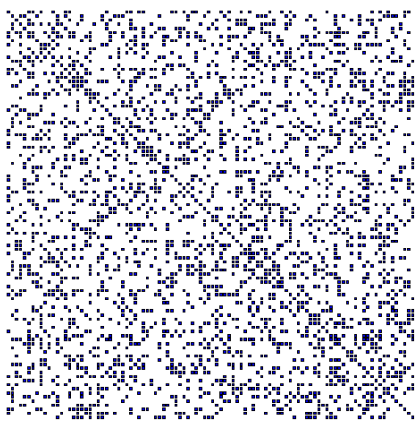
保持$X \in \mathbf{S}^n$ 中特定分量固定; 寻找其他分量的值使得$X \succeq 0$

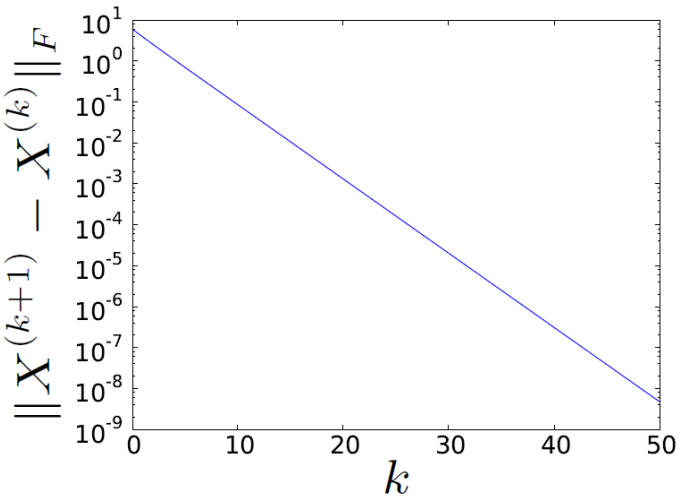
- $C_1 = \mathbf{S}^n_+$, C_2 是$\mathbf{S}^n$ 中特定分量固定的（仿射）集合
- 通过特征值分解与截断，将X投影到C_1

$$P_1(X) = \sum_{i=1}^{n} \max\{0, \lambda_i\} q_i q_i^{\mathrm{T}} \quad \text{if } X = \sum_{i=1}^{n} \lambda_i q_i q_i^{\mathrm{T}}$$

- 通过将特定分量的值重设为固定值，将X投影到C_2

100×100
matrix missing
71% entries





提纲

投影次梯度法

考虑在凸集 C 上极小化一个不可微凸函数 f：

$$\min_x f(x) \quad \text{s.t. } x \in C$$

- 投影次梯度法和次梯度法的区别在于每步迭代都需要将迭代点投影到集合C 上：

$$x^{k+1} = P_C\left(x^k - \alpha_k g^k\right), \quad k = 0, 1, 2, \ldots$$

- 假设投影算子 P_C可以计算，那么投影次梯度法可以得到和次梯度法相同的收敛性保证.

投影次梯度法

容易计算投影的集合举例：

- 仿射变换的像：$\{Ax+b : x \in \mathbb{R}^n\}$
- 线性系统的解集：$\{x : Ax = b\}$
- 非负象限：$\mathbb{R}^n_+ = \{x : x \geq 0\}$
- 一些范数球：$\{x : \|x\|_p \leq 1\}, p = 1, 2, \infty$
- 一些简单的多面体和简单的锥

注意：存在很多简单的集合C，它们的投影算子 P_C 很难计算
例：任意一个多面体$C = \{x : Ax \leq b\}$

提纲

次梯度法的最优性

第19页得出的上界$\hat{f}^k - f^* \leqslant \frac{GR}{\sqrt{k}}$ 可以进一步改进吗?

问题设定:

- f 是凸函数, 有一个极小点x^*
- 已知初始点x^0 满足$\|x^0 - x^*\|_2 \leq R$
- 已知f 在$\{x|\|x - x^0\|_2 \leq R\}$ 上的利普希茨常数G
- f 的定义方式: 给定x, 返回$f(x)$ 及一个次梯度

算法设定: 每步迭代点x^i 都可用任一方法在如下集合中进行选择, 迭代k 步

$$x^0 + \text{span}\{g^0, g^1, \cdots, g^{i-1}\}$$

测试问题

$$f(x) = \max_{i=1,\cdots,k} x_i + \frac{1}{2}\|x\|_2^2, \quad x^0 = 0$$

- 测试问题的解：$x^* = -\frac{1}{k}(\underbrace{1,\cdots,1}_{k},\underbrace{0,\cdots,0}_{n-k})$ 以及 $f^* = -\frac{1}{2k}$
- $R = \|x^0 - x^*\|_2 = \frac{1}{\sqrt{k}}$ 且 $G = 1 + \frac{1}{\sqrt{k}}$
- 返回的次梯度是 $e_{\hat{j}} + x$ 其中 $\hat{j} = \min\{j | x_j = \max_{i=1,\ldots,k} x_i\}$

迭代： 当 $i = 0,\cdots,k-1$, 分量 $x_{i+1}^i,\cdots,x_k^i$ 等于零

$$\hat{f}^k - f^* = \min_{i<k} f(x^i) - f^* \geq -f^* = \frac{GR}{2(1+\sqrt{k})}$$

结论： 上界 $O(1/\sqrt{k})$ 无法再改进

总结：次梯度法

- 能够处理一般的不可微凸函数
- 常能推导出非常简单的算法
- 收敛速度可能非常缓慢
- 没有很好的停机准则
- 理论复杂度：迭代$O(1/\epsilon^2)$步，得到ϵ-次优的点
- 一种“最优”的一阶方法：$O(1/\epsilon^2)$无法再改进

提纲

割平面法

- 假设$f(x)$为凸函数，但不一定可微，由次梯度的性质，我们有

$$f(x) = \sup_{y} \sup_{g \in \partial f(y)} \left[f(y) + g^{\mathrm{T}}(x - y) \right]$$

- $f(x)$的极小化问题等价于

$$\begin{aligned} \min_{v,x} \quad & v \\ \text{s.t.} \quad & v \geq f(y) + g^{\mathrm{T}}(x - y), \quad \forall y \in \mathbb{R}^n, g \in \partial f(y). \end{aligned} \tag{1}$$

- 割平面法通过求解一系列近似线性规划来逼近(1)的解：假设已有迭代点序列x_i $(i = 1, \cdots, k)$，$g_i \in \partial f(x_i)$，在第$k+1$步迭代中计算如下子问题，得到v_{k+1}, x_{k+1}

$$\begin{aligned} v_{k+1}, x_{k+1} = \ & \operatorname*{argmin}_{v,x} \quad v \\ \text{s.t.} \quad & v \geq f(x_i) + g_i^{\mathrm{T}}(x - x_i), \quad i = 1, \cdots, k. \end{aligned} \tag{2}$$

割平面法

- 子问题(2)是原问题(1)的一个近似
- 每步迭代后，子问题都添加一个割平面约束
- 这意味着每次迭代后，一个不包含解的区域被割平面排除
- 子问题的多面体可行域越来越小，最终收敛到原问题(1)的解
- 假设f有下界，$\{v_k\}$和$\{x_k\}$是算法产生的迭代序列，可以证明：
  - $v_2 \leq v_3 \leq \cdots \leq v_k \to f^*$
  - $\{x_k\}$的任一聚点都是$f(x)$的最小值点
- 割平面法的缺点：
  - 假设f可微，当k足够大时，g_k会趋于零，约束会变得很病态
  - 约束个数只增不减，所以当k很大时，求解子问题的代价会很昂贵

捆集方法

假设$\{x_i\}$为算法产生的迭代序列. 第k次迭代的搜索方向由下式给出：

$$d_k = -\sum_{i=1}^{k} \lambda_i^{(k)} g_i, \quad g_i \in \partial f(x_i),$$

其中$\lambda_i^{(k)}$是通过解如下子问题得到的：

$$\begin{aligned} \min \quad & \left\| \sum_{i=1}^{k} \lambda_i g_i \right\| \\ \text{s.t.} \quad & \sum_{i=1}^{k} \lambda_i = 1, \lambda_i \geq 0, \\ & \sum_{i+1}^{k} \lambda_i t_i^{(k)} \leq \bar{\epsilon}, \end{aligned} \tag{3}$$

其中$t_i^{(k)} \geq 0$为加权因子，$\bar{\epsilon} > 0$为一给定常数.

捆集方法

(1) 给出初值$x_0 \in \mathbb{R}^n$，计算$g_0 \in \partial f(x_0)$，选取$0 < m_2 < m_1 < \frac{1}{2}, 0 < m_3 < 1, \epsilon > 0, \eta > 0, k := 1, t_1^{(1)} = 1.$

(2) 求解(3)得到$\lambda_i^{(k)}$并由此计算d_k；如果$\|d_k\| \leq \eta$则停；

(3) 计算$y_k = x_k + \alpha_k d_k$使得

$$f(y_k) \leq f(x_k) - m_2 \alpha_k \|d_k\|_2^2 \tag{4}$$

或者

$$f(y_k) - \alpha_k g_{k+1}^{\mathrm{T}} d_k \geq f(x_k) - \epsilon,$$

成立，其中$g_{k+1} \in \partial f(y_k)$，如果(4)不成立，则转步5；

(4) $x_{k+1} := y_k, t_{k+1}^{(k+1)} = 1$

$t_j^{(k+1)} = t_j^{(k)} + f(x_{k+1}) - f(x_k) - \alpha_k g_j^{\mathrm{T}} d_k, j = 1, \cdots, k$

令$k := k + 1$；转步2.

(5) $x_{k+1} := x_k, t_j^{(k+1)} = t_j^{(k)} (j = 1, \cdots, k)$

$t_{k+1}^{(k+1)} = f(x_k) - f(y_k) + \alpha_k g_{k+1}^T d_k$

令$k := k + 1$；转步2.

捆集方法

- 捆集方法是一类下降算法，要求每次迭代都有$f(x_{k+1}) \leq f(x_k)$
- 当$f(x)$为凸的二次函数且$t_i^{(k)} = 0$时，在精确线搜索下，捆集方法的搜索方向和共轭梯度法相同
- 收敛性保证：设f是凸的，$\|\partial f(x)\|$在包含集合$\{x \mid f(x) \leq f(x_1)\}$的某一开集上是有界的，设算法产生的点列使得$f(x_k)$下方有界，则算法必经过有限次迭代后终止