最优化理论与方法第三次上机作业

郑灵超* 1601110040

2017 年 12 月 29 日

1 上机作业介绍

Minimax 问题是最优化一类经典的最优化问题，其描述如下：

$$\min_{x} \max_{1\le i\le m} f_i(x). \tag{1}$$

由于对象函数

$$F(x) = \max_{1\le i\le m} f_i(x) \tag{2}$$

并不一定是可微函数，传统的优化方法往往难以奏效，我们需要为其单独设计算法进行优化。

2 算法介绍

2.1 光滑函数方法

在 [1] 中，作者提出引入一个光滑函数来逼近目标函数 $F(x)$,

$$f(x,\mu) = \mu \ln \sum_{i=1}^{m} \exp\left(\frac{f_i(x)}{\mu}\right) \tag{3}$$

不难证明 $f(x,\mu)$ 关于 μ 单调递增，且

$$F(x) \le f(x,\mu) \le F(x) + \mu \ln m,$$

因此当 $\mu \to 0$ 时，可以用 $f(x,\mu)$ 给出目标函数一个较好的逼近。因此我们对 $\mu \to 0$ 时的 $f(x,\mu)$ 进行优化操作，就能实现对原目标函数的优化。

*Email: lczheng@pku.edu.cn

算法说明

1. 设置初始点 x_0, 初始参数 $\mu > 0, 0 < \beta < 1$.
2. 利用 Newton 方法计算此时的负梯度方向:

$$\nabla_x^2 f(x,\mu)d = -\nabla_x f(x,\mu). \tag{4}$$

3. 利用线搜索计算步长 α.
4. $x = x + \alpha d$.
5. $\mu = \mu\beta$，若满足终止条件，则终止；否则回到第二步。

2.2 Least-p 方法

在 [2] 中，作者对 Least-p 方法进行了改进，采用了另外一种函数逼近形式

$$U(x,u,p,\xi) = \begin{cases} M(x,\xi)\left(\sum_{i\in S(x,\xi)} u_i \left(\frac{f_i(x)-\xi}{M(x,\xi)}\right)^q\right)^{1/q}, & M(x,\xi) \neq 0 \\ 0 & M(x,\xi) = 0. \end{cases} \tag{5}$$

其中

$$u_i \geq 0, \quad \sum_{i\in I} u_i = 1, \quad q = p \times \operatorname{sgn}(M(x,\xi)), \tag{6}$$

而

$$M(x,\xi) = F(x) - \xi, \quad S(x,\xi) = \{i \in I | (f_i(x)-\xi)M(x,\xi) > 0\}. \tag{7}$$

算法说明

1. 设置初始 $p \geq 1, c \geq 1, u, \xi, x_0$
2. 在当前参数下极小化逼近函数。
3. 重新设置参数

$$u_i = \frac{v_i}{\sum_{i\in I} v_i}, \quad i \in I,$$

其中

$$v_i = \begin{cases} u_i \left(\dfrac{f_i(x) - \xi}{M(x,\xi)} \right)^{q-1}, & i \in S(x,\xi), \\ 0, & i \in I - S(x,\xi). \end{cases}$$

4. 若满足终止条件，则终止；否则 $p = cp$, 回到第二步.

2.3 SQP 方法

我采用了 [3] 中提出的用于求解 Minimax 问题的 SQP 算法，其算法如下：

算法介绍

1. 选取初始点 X_0, 初始正定对称矩阵 $B = I$, 取 $0 < \sigma < \frac{1}{2}, 0 < \delta \le 1, \varepsilon \ge 0$.

2. 求解二次规划问题

$$\begin{aligned} \min \quad & \frac{1}{2} d^T B d + \frac{\delta}{2} t^2 + t, \\ s.t. \quad & \nabla f_i(x)^T d - t \le F(x) - f_i(x), \quad 1 \le i \le m. \end{aligned} \tag{8}$$

其最优解设为 (d,t)，相应的 Lagrange 乘子为 λ, 置

$$d = \frac{d}{1 + \delta t}, \quad \lambda = \frac{\lambda}{1 + \delta t}.$$

3. 若 $\|d\| \le \varepsilon$，则终止，否则取 $0 < \beta < 1$

4. 计算最小的满足

$$F(x + \beta^j d) \le F(x) + \sigma \beta^j t \tag{9}$$

的自然数 j, 取 $x' = x + \beta^j d$.

5. 修正 B,

$$\begin{aligned} s &= x' - x, \\ y &= \sum_{i=1}^{m} \lambda_i \left[\nabla f_i(x') - \nabla f_i(x) \right], \\ y &= \theta y + (1 - \theta) B s, \end{aligned}$$

其中

$$\theta = \begin{cases} 1, & y's \geq 0.2s'Bs, \\ \dfrac{0.8s'Bs}{s'Bs - y's}, & otherwise. \end{cases}$$

令

$$B = B - \frac{Bss'B}{s'Bs} + \frac{yy'}{y's}. \tag{10}$$

转步 2.

3 数值结果

本次上机报告我们测试了三个案例：

3.1 例 1

第一个例子是 [1] 中的 exmaple 1, 三种方法的计算结果如下：

- 光滑函数方法: 在 21 步迭代后到达解 $(1.1390, 0.8996)^T$，函数调用次数为 84.
- Least-p 方法: 在 22 步迭代后到达解 $(1.1390, 0.8996)^T$，函数调用次数为 88.
- SQP 方法: 在 8 步迭代后到达解 $(1.1390, 0.8996)^T$，函数调用次数为 12.

3.2 例 3

第一个例子是 [1] 中的 exmaple 3，即 Rosen-Suzuki 问题, 三种方法的计算结果如下：

- 光滑函数方法: 在 24 步迭代后到达解 $(0, 1, 2, -1)^T$，函数调用次数为 110.
- Least-p 方法: 在 17 步迭代后到达解 $(0, 1, 2, -1)^T$，函数调用次数为 79.
- SQP 方法: 在 22 步迭代后到达解 $(0, 1, 2, -1)^T$，函数调用次数为 150.

3.3 滤波器问题

第一个例子是 [2] 中的 digital filter 问题，三种方法的计算结果如下：

- 光滑函数方法: 无法计算。
- Least-p 方法: 无法计算。
- SQP 方法: 在 39 步迭代后到达解 $(0.0001, 0.9801, 0.0000, -0.1657, 0.0001, -0.7351, 0.0001, -0.7672, 0.3679)^T$，函数调用次数为 357, 函数值 $f = 0.006189$.

4 上机报告总结

本次上机报告，我们实现了三种梯度型下降方法，在此作一个总结:

1. 光滑函数法和 Least-p 法都是利用一个带参数的光滑函数来逼近目标函数，再令参数趋于 0 或无穷，来实现对目标函数的逼近。可能是由于我们在求解其子问题时采用 Newton 迭代法，对较为复杂的滤波器问题，这两个方法的表现并不好。

2. SQP 方法是用于求解二次规划问题的经典方法，我们这里采用了 matlab 自带的 quadprog 函数求解它的二次规划子问题。该方法对测试的三个例子均有不错的表现。

参考文献

[1] Song Xu. Smoothing Method for Minimax Problems. Computational Optimization and Applications, 20, 267–279, 2001.

[2] Christakis CHARALAMBOUS. ACCELERATION OF THE LEAST pth ALGORITHM FOR MINIMAX OPTIMIZATION WITH ENGINEERING APPLICATIONS.Mathematical Programming 17 (1979) 270-297.

[3] 薛毅. 求解 Minimax 优化问题的 SQP 方法. 系统科学与数学, 22，355-364，2002.