

凸优化算法 I: 内点法(interior point method)求解线性规划问题

拉普拉斯算符
数学话题下的优秀答主

111 人赞同了该文章

第一节：导论

线性规划问题的一般形式为

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \preceq b \end{aligned}$$

这里，目标函数为线性函数 $c^T x: \mathbb{R}^n \rightarrow \mathbb{R}$ ，约束条件为 $A_{ij}x_j \leq b_i, i=1,2,\dots,m$. 矩阵 A 为 $m \times n$ 的满秩矩阵，其中 m 为约束条件的个数， n 为变量的个数。通常约束条件的个数大于变量的个数，所以有 $m > n$ 。

第二节：线性规划问题的等价(近似)表述

这个线性规划问题可以重新表述为计算 $\min f(x)$ ，其中

$$f(x) = c^T x + \sum_{i=1}^m I(A_{ij}x_j - b_i)$$

这里，我们使用了一个indicator函数，定义为

$$I(u) = \begin{cases} 0 & \text{if } u \leq 0 \\ \infty & \text{if } u > 0 \end{cases}$$

引入这个函数的意义在于可以将约束条件直接写入到目标函数里面，这样我们直接求新的函数的极小值就可以了，而不必借助于未知乘子。但是这里有一个问题，那就是indicator函数存在不可求导的点，因此在求函数极小值的时候我们没法通过普通的微分法来确定函数的极小值。为了规避这个问题，我们可以用一个光滑的函数来近似这个indicator函数。一个不错的选择是用

$I_t(u) = \frac{1}{t} \log(-u)$ 来代替indicator函数。 $I_t(u)$ 只有在 $u < 0$ 的时候有定义，我们规定当 $u > 0$ 的时候 $I_t(u) = \infty$ 。而且参数 $t > 0$ 越大，函数 $I_t(u)$ 就越接近于 $I(u)$ 。所以我们可以通过调节 t 的值来调节这个函数的近似程度。使用这个近似的indicator函数，我们新的的目标函数可以写作

$$f(x) = tc^T x - \sum_{i=1}^m \log(-A_{ij}x_j + b_i)。$$

因为线性函数是凸函数，并且 $I_t(u)$ 也是凸函数，所以 $f(x)$ 是凸函数，因此我们可以很容易用凸优化的经典方法得到该函数的极小值。

第三节：计算函数的梯度和Hessian矩阵

为了求函数的极小值，根据微积分的经典结果，我们只需令函数的梯度等于零，然后计算梯度为零时对应的解 x^* 。

函数的梯度为

$$\frac{\partial f}{\partial x_k} = tc_k + \sum_{i=1}^m \frac{-A_{ik}}{A_{ij}x_j - b_i}$$

Hessian 矩阵为

$$\frac{\partial^2 f}{\partial x_k \partial x_l} = \sum_{i=1}^m \frac{A_{ik}A_{il}}{(A_{ij}x_j - b_i)^2}$$

定义对角型矩阵为

$$D_{ij} = \delta_{ij} \frac{1}{(A_{ik}x_k - b_i)^2}$$

于是Hessian矩阵可以写作

$$H_f = A^T D A$$

因为 D 为正定矩阵，所以Hessian矩阵至少为半正定矩阵。所以函数 $f(x)$ 是一个凸函数。而且矩阵 D 为可逆矩阵，矩阵 A 满秩，所以Hessian矩阵为可逆矩阵。于是函数 $f(x)$ 为强凸函数。所以，要计算 $\nabla f = 0$ 的根，我们可以用高效的牛顿迭代法。

第四节：牛顿迭代法

我们现在的目标是计算 $\nabla f = 0$ 的根。因为Hessian矩阵可逆，所以我们可以用牛顿迭代法求解。牛顿迭代法为

$$x^{(n+1)} = x^{(n)} - H^{-1} \nabla f$$

在这个迭代过程中，参数 t 为固定的。每对应一个 t ，我们可以得到一个解 x_t^* 。如果我们扫描参数 t ，我们就可以得到一系列的解。其中最大的 t 的对应的解应该最精确。

一个更好的算法是选取一个比较小的初始参数 t_0 ，求出这个参数对应的解 $x_{t_0}^*$ 。然后增加 t ，用之前得到的解 $x_{t_0}^*$ 来初始化当前 t 所对应的牛顿迭代法的试解。这样算出来的解应该比直接计算 t 所对应的解更加精确。这样逐步迭代从小 t_0 到大 t 可以得到一系列的解，最后得到的解 x_t^* 称作是淬火解。这个解应该可以满足我们的精度需求。

第五节：一个例子

为了展示该算法的功效，我们举一个简单的例子。这个例子可以用简单的几何方法求解出来。我们要展示的是，我们可以用数值方法得到同样的结果（在一定的精度范围之内）。之所以要用数值方法求解这个简单的问题，是因为数值方法对更复杂的问题同样有效，而简单的几何方法对复杂问题却已经不适用了。

现在要研究的例子为

$$\begin{aligned} \min \quad & x + y \\ \text{s.t.} \quad & x + 2y \leq 1 \\ & 2x + y \leq 1 \\ & x \geq 0 \\ & y \geq 0 \end{aligned}$$

也就是

$$c = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}, b = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}。$$

梯度为

$$\nabla f(x) = tc - A^T \begin{pmatrix} \frac{1}{x+2y-1} \\ \frac{1}{2x+y-1} \\ \frac{1}{-x} \\ \frac{1}{-y} \end{pmatrix}$$

Hessian矩阵为

$$H_f(x) = A^T D A$$

其中，对角型矩阵 D 为

$$D = \begin{pmatrix} \frac{1}{(x+2y-1)^2} & 0 & 0 & 0 \\ 0 & \frac{1}{(2x+y-1)^2} & 0 & 0 \\ 0 & 0 & \frac{1}{x^2} & 0 \\ 0 & 0 & 0 & \frac{1}{y^2} \end{pmatrix}$$

对于一个固定的参数 t ，选择一个恰当的初始解 $x^{(0)}$ ，代入牛顿迭代公式

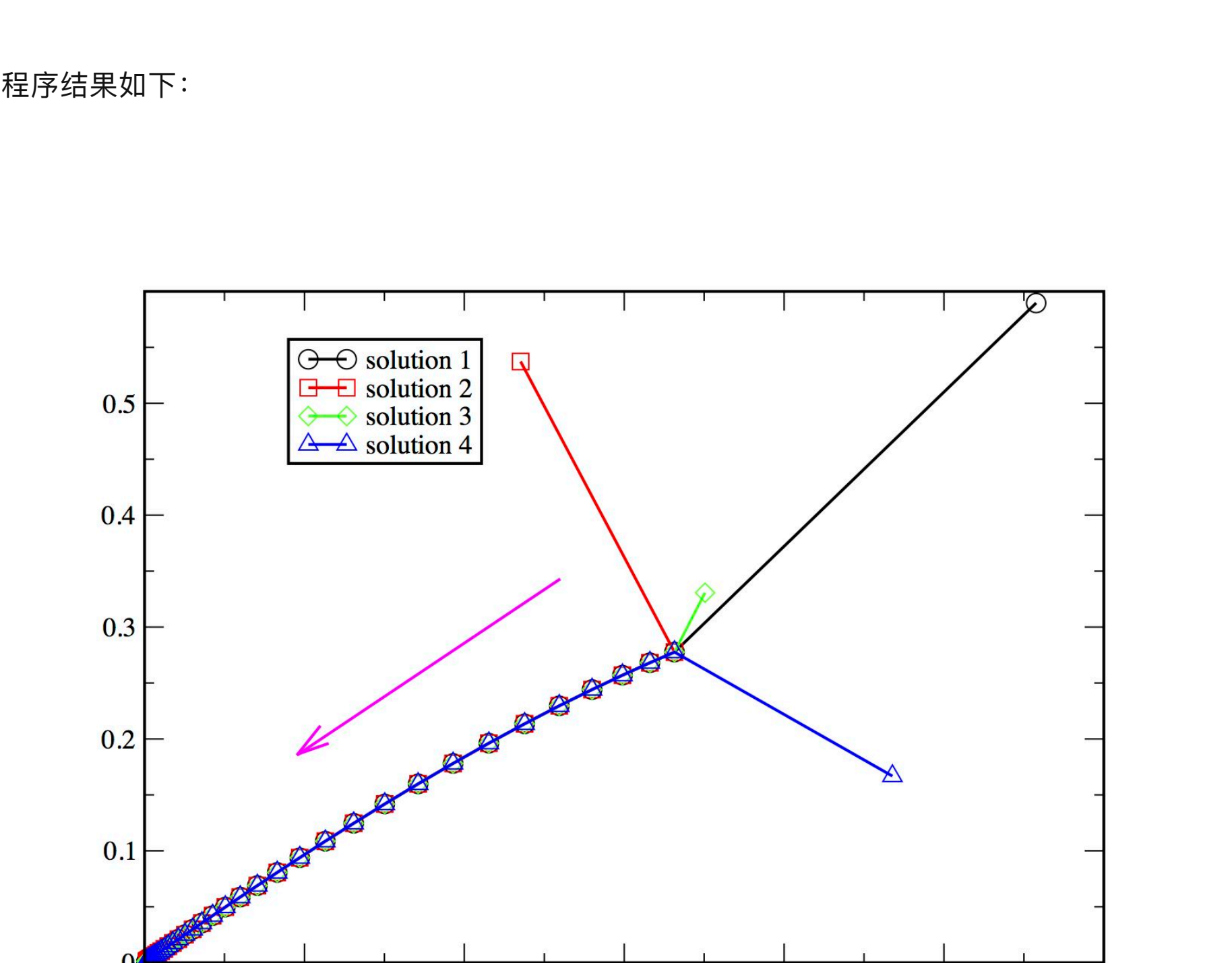
$$x^{(n+1)} = x^{(n)} - H_f(x^{(n)})^{-1} \nabla f(x^{(n)}), n \geq 0$$

可以得到一个依赖于参数 t 的解 x_t^* 。

用几何方法很容易求得这个例子的解为 $(x^*, y^*) = (0, 0)$ 。所以我们期待，当 $\lim_{t \rightarrow \infty} (x_t^*, y_t^*) = (0, 0)$ 。

我需要用程序求这个例子的数值解。我写了一个Python程序实现这个算法，程序地址为PrimerLi/linear-programming。

程序结果如下：



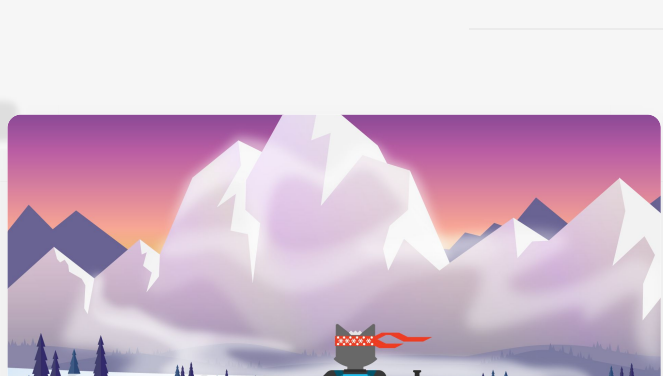
这里用了四个不同的初始点来初始化程序，最终得到解都收敛到了 $(0, 0)$ 。箭头方向为参数 t 增加时 (x_t^*, y_t^*) 移动的方向。这正是我们期待的结果。

编辑于 2019-08-26

凸优化

线性规划

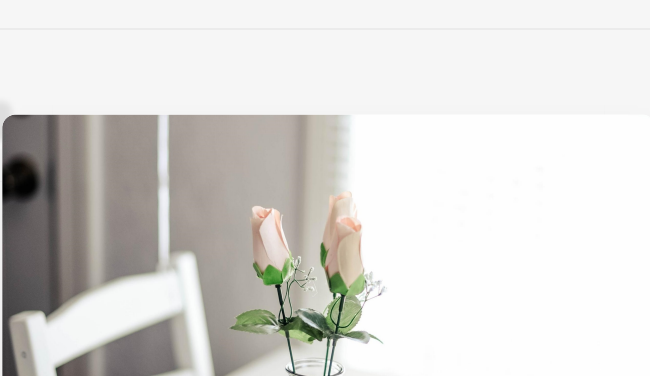
推荐阅读



01 什么是凸优化问题？主要目标和内容是什么？

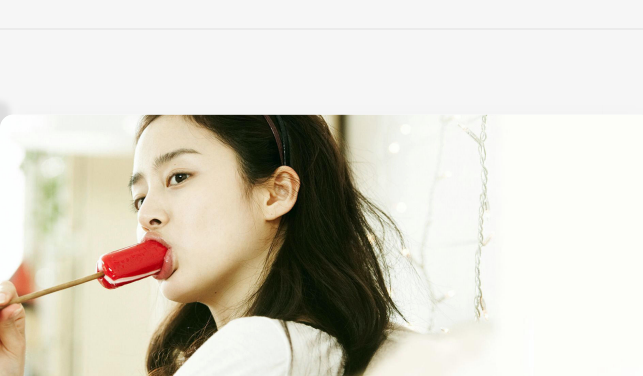
落日翻车

发表于凸优化笔记



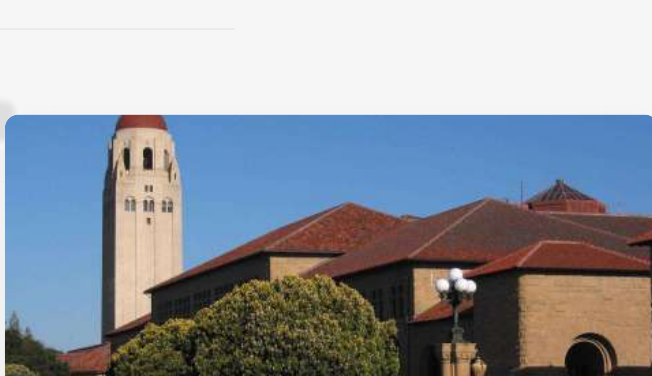
【凸优化笔记6】-拉格朗日对偶 (Lagrange duality)、K...

Lauer



凸优化笔记(6)无约束凸优化问题的求解算法

枕月



斯坦福大学 凸优化 对偶

Zihao Qiu

16 条评论

切换为时间排序

写下你的评论...

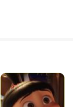


苏鲁支

2020-06-09

一直在学习如何用工程的方式求解带约束凸优化问题，看了作者的文章很有启发。期待作者再写一篇投影法求解线性划归的文章

1 个赞



大学渣

2018-04-07

不好意思打扰一下，我觉得那个第一节里边，约束条件那个地方是不是写的有问题，应该是sum j = {1:n} A_{(ij)} * x_j <= b_i?

1 个赞



1860390 回复 大学渣

2018-04-15

对，应该就是 $\sum A(i, j) * x_j \leq b_i$

1 个赞



拉普拉斯算符 (作者) 回复 大学渣

2018-08-26

我这里用了爱因斯坦求和约定，也就是重复的指标表示求和。j重复出现了两次，所以就相当于对求和了。这个求和约定可以极大地简化指标运算，并且在物理和数学中广为应用。

2 个赞



LGYKO

2020-09-12

牛顿迭代法中的H是指H_0吗

1 个赞



拉普拉斯算符 (作者) 回复 LGYKO

2020-09-12

是的

1 个赞



沧海小鱼儿

2019-08-26

indicator函数那个地方，I(u)自变量是u，所以后面应该是if u>0而不是x>0

1 个赞



拉普拉斯算符 (作者) 回复 沧海小鱼儿

2019-08-26

改了，谢谢

1 个赞



Miracle-Zym 回复 拉普拉斯算符 (作者)

2020-04-25

这个indicator函数出自哪本书。

1 个赞

展开其他 1 条回复



饭饭

2019-06-06

求问，如何确保凸优化问题一定有解？

1 个赞



沧海小鱼儿 回复 饭饭

2019-08-26

凸优化问题一定有解，而且求解复杂度是确定的，最难的是如何判断一个问题是凸的，以及如何建立一个凸优化模型

1 个赞



大学渣

2018-04-07

而且为啥梯度是4*1维的，Hessian是4*4的？

1 个赞

大头菠萝 回复 大学渣

2018-04-07

因为是对梯度向量再求导，向量对向量求导就是4*4了

2 个赞

好好计划

2018-03-08

“通常约束条件的个数大于变量的个数，所以有”，没有写反吗？

1 个赞

陳DJ 回复 好好计划

2018-11-04

你看看作者给的例子呗？

1 个赞