

求解一类双线性规划问题的数值算法

答辩人: 胡雨宽

导 师: 殷俊锋 教授

同济大学数学科学学院

2019.6.4



提纲

- ① 引言
- ② 最优性条件
- ③ 算法设计
- ④ 收敛性分析
- ⑤ 数值实验
- ⑥ 总结

提纲

- 1 引言
- 2 最优性条件
- 3 算法设计
- 4 收敛性分析
- 5 数值实验
- 6 总结

问题背景

最优运输问题 (*Villani '08*):

- 建立有效比较概率分布的几何工具;
- 极小化将一概率分布“运输”到另一概率分布的“花费”.

$$\begin{aligned} \min_{a_{ij}} \quad & \sum_{i,j} c_{ij} a_{ij} \\ \text{s.t.} \quad & \sum_j a_{ij} = f_i, \quad i = 1, \dots, m, \\ & \sum_i a_{ij} = g_j, \quad j = 1, \dots, n, \\ & a_{ij} \geq 0, \quad i = 1, \dots, m, j = 1, \dots, n, \end{aligned} \quad (\text{LP})$$

这里 $c_{ij}, f_i, g_j \geq 0, \forall i, j$.

问题陈述

$$\begin{aligned}
 \min_{X, Y} \quad & \sum_{i \neq j} \frac{x_{ij}}{|r_i - r_j|} + \sum_{i \neq k} \frac{y_{ik}}{|r_i - r_k|} + \sum_{i, j, k: j \neq k} \frac{x_{ij} y_{ik}}{|r_j - r_k|} \\
 \text{s.t.} \quad & \sum_j x_{ij} = \rho_i, \quad i = 1, \dots, n, \\
 & \sum_i x_{ij} = \rho_j, \quad j = 1, \dots, n, \\
 & \sum_k y_{ik} = \rho_i, \quad i = 1, \dots, n, \\
 & \sum_i y_{ik} = \rho_k, \quad i = 1, \dots, n, \\
 & x_{ij}, y_{ik} \geq 0, \quad i, j, k = 1, \dots, n, \\
 & x_{ii}, y_{ii} = 0, \quad i = 1, \dots, n,
 \end{aligned} \tag{1}$$

其中 $X, Y \in \mathbb{R}^{n \times n}$, $r = (r_1, \dots, r_n)^T$, $\rho = (\rho_1, \dots, \rho_n)^T \in \mathbb{R}_+^n$. 将 $\{|r_i - r_j|\}$ 储存于 $R = (r_{ij}) \in \mathbb{R}^{n \times n}$, 其中

$$r_{ij} = \begin{cases} 1/|r_i - r_j|, & i \neq j, \\ 0, & i = j. \end{cases}$$

研究现状

角度 1 可分离约束的双线性规划. 广泛的应用 (*Konno '71*等).

已有方法:

- 割平面法 (*Ritter '66*等);
- 分支定界法 (*Falk '73*等).

角度 2 非凸二次规划.

已有方法 (*Nocedal/Wright '06*):

- 逐步二次规划 (SQP);
- 积极集法 (Active-set Methods);
- 内点法 (Interior-point Methods);
-

以上方法均以列向量为求解对象 \Rightarrow 计算量问题.

预备知识

标准矩阵内积

对 $\forall A, B \in \mathbb{R}^{n \times n}$, 定义

$$\langle A, B \rangle := \text{tr}(A^T B) = \sum_{i,j} a_{ij} b_{ij}.$$

问题(1)即可化为矩阵形式:

$$\begin{aligned} \min_{X, Y} \quad & \langle R, X \rangle + \langle R, Y \rangle + \langle Y, XR \rangle \\ \text{s.t.} \quad & X\mathbf{1} = \rho, X^T\mathbf{1} = \rho, \text{tr}(X) = 0, X \geq 0, \\ & Y\mathbf{1} = \rho, Y^T\mathbf{1} = \rho, \text{tr}(Y) = 0, Y \geq 0, \end{aligned} \tag{2}$$

其中 $\mathbf{1}$ 为全 1 向量.

预备知识 (续)

假设

问题(2)的所有稳定点 (X, Y) 均满足 $X = Y$.

\Rightarrow 简化问题:

$$\begin{aligned} \min_X \quad & 2\langle X, R \rangle + \langle X, XR \rangle \\ \text{s.t.} \quad & X\mathbf{1} = \rho, X^T\mathbf{1} = \rho, \text{tr}(X) = 0, X \geq 0. \end{aligned} \quad (3)$$

引入分裂变量 $Z \in \mathbb{R}^{n \times n}$, 进一步得到等价的

$$\begin{aligned} \min_{X, Z} \quad & f(X, Z) \triangleq 2\langle X, R \rangle + \langle Z, XR \rangle \\ \text{s.t.} \quad & X\mathbf{1} = \rho, \text{tr}(X) = 0, \\ & Z^T\mathbf{1} = \rho, Z \geq 0, \\ & X = Z. \end{aligned} \quad (4)$$

注

问题(4)的约束非常特殊, 但在最优运输问题中十分普遍.

提纲

- 1 引言
- 2 最优性条件
- 3 算法设计
- 4 收敛性分析
- 5 数值实验
- 6 总结

最优性条件

问题(4)的 KKT 条件

若 (X^*, Z^*) 为问题(4)的解, 则存在拉格朗日乘子 $\mu^* \in \mathbb{R}, \lambda_1^*, \lambda_2^* \in \mathbb{R}^n, \Phi^*, 0 \leq \Omega^* \in \mathbb{R}^{n \times n}$, 使得

$$\left\{ \begin{array}{l} 2R + Z^*R - \lambda_1^* \mathbf{1}^T - \Phi^* - \mu^* I = 0, \\ X^*R - \mathbf{1} (\lambda_2^*)^T + \Phi^* - \Omega^* = 0, \\ X^* \mathbf{1} = \rho, \text{tr}(X^*) = 0, \\ (Z^*)^T \mathbf{1} = \rho, Z^* \geq 0, \\ \Omega^* \geq 0, \\ \Omega^* \circ Z^* = 0, \end{array} \right\} \quad \begin{array}{l} \text{稳定性条件} \\ \text{或对偶可行性条件,} \\ \text{原始可行性条件,} \\ \text{互补松弛条件,} \end{array} \quad (5)$$

这里 “ \circ ” 表示两矩阵的 Hadamard 积.

提纲

- 1 引言
- 2 最优性条件
- 3 算法设计
- 4 收敛性分析
- 5 数值实验
- 6 总结

ADMM 算法简介

ADMM 算法 (介绍可见 *Boyd, et al. '10*) 求解问题一般形式:

$$\begin{aligned} \min_{x \in \mathbb{R}^d} \quad & \theta(x) := \sum_{i=1}^n \theta_i(x_i) + \ell(x_1, \dots, x_n) \\ \text{s.t.} \quad & \sum_{i=1}^n A_i x_i = b. \end{aligned} \tag{6}$$

现有工作简介:

- 凸可分问题 (*Boyd, et al. '10, He/Yuan '12*等): $n \geq 2, \ell = 0, \theta_1, \theta_2$ 是凸函数. 可能发散 (*Chen, et al. '16*) \Rightarrow 无假设的困境;
- 凸不可分问题 (*Hong, et al. '14, Chen, et al. '19*), 非凸问题. 理论缺乏, 应用广泛.

特点: 交替求解子问题, 子问题需要易于求解; 适合于分布式计算.
 \Rightarrow 引入 Z 的理由之一.

ADMM 算法简介 (续)

$$\mathcal{L}_A(X, Z, \Phi) = f(X, Z) - \langle \Phi, X - Z \rangle + \frac{\beta}{2} \|X - Z\|_F^2, \quad (7)$$

其中 $\beta > 0$ 为惩罚因子.

框架 1 求解问题 (4) 的 ADMM 算法框架

输入: $X^0, Z^0, \Phi^0, \beta^0, k := 0$.

输出: X^k, Z^k, Φ^k .

- 1: **while** 收敛性测试未通过 **do**
 - 2: $X^{k+1} = \arg \min_{X: X\mathbf{1}=\rho, \text{tr}(X)=0} \mathcal{L}_A(X, Z^k, \Phi^k);$ %X 子问题
 - 3: $Z^{k+1} = \arg \min_{Z: Z^T\mathbf{1}=\rho, Z \geq 0} \mathcal{L}_A(X^{k+1}, Z, \Phi^k);$ %Z 子问题
 - 4: 更新 Φ^k 得到 $\Phi^{k+1};$ % 拉格朗日乘子更新
 - 5: 如有需要, 更新 β^k 得到 $\beta^{k+1};$
 - 6: $k := k + 1;$
 - 7: **end while**
-

子问题的求解 - X 子问题

省去上标 k , 改用 '+' 标记更新值.

$$\begin{aligned} \min_X \quad & 2\langle X, R \rangle + \langle Z, XR \rangle - \langle \Phi, X - Z \rangle + \frac{\beta}{2} \|X - Z\|_F^2 \\ \text{s.t.} \quad & X\mathbf{1} = \rho, \quad \text{tr}(X) = 0. \end{aligned} \quad (8)$$

$$\begin{aligned} M_1 &= 2R\mathbf{1} + ZR\mathbf{1} - \Phi\mathbf{1} - \beta Z\mathbf{1} + \beta\rho, \\ m_1 &= 2\text{tr}(R) + \text{tr}(ZR) - \text{tr}(\Phi) - \beta\text{tr}(Z). \end{aligned}$$

$$\mu = \frac{1}{n-1} \left(-\frac{1}{n} \mathbf{1}^T M_1 + m_1 \right), \quad \lambda_1 = \frac{1}{n} (M_1 - \mathbf{1}\mu). \quad (9)$$

$$X^+ = -\frac{1}{\beta} (2R + ZR - \lambda_1 \mathbf{1}^T - \mu I - \Phi - \beta Z). \quad (10)$$

(9)和(10)给出 X 子问题(8)的解 X^+ .

子问题的求解 - Z 子问题

$$\begin{aligned} \min_Z \quad & \langle Z, X^+ R \rangle - \langle \Phi, X^+ - Z \rangle + \frac{\beta}{2} \|X^+ - Z\|_F^2 \\ \text{s.t.} \quad & Z^T \mathbf{1} = \rho, \quad Z \geq 0. \end{aligned} \quad (11)$$

忽略非负约束, 求超平面 $Z^T \mathbf{1} = \rho$ 上的一点 \tilde{Z} :

$$\lambda_2 = \frac{1}{n} \left[R(X^+)^T \mathbf{1} + \Phi^T \mathbf{1} - \beta (X^+)^T \mathbf{1} + \beta \rho \right],$$

$$\tilde{Z} = \frac{1}{\beta} (X^+ R + \Phi - \beta X^+ - \mathbf{1} \lambda_2^T).$$

子问题的求解 - Z 子问题 (续)

问题(11)的等价形式:

$$\begin{aligned} \min_Z \quad & \|Z - \tilde{Z}\|_F^2 \\ \text{s.t.} \quad & Z^T \mathbf{1} = \rho, \quad Z \geq 0. \end{aligned} \quad (12)$$

分块

$$\begin{aligned} Z &= [z_1, \dots, z_n], \quad \tilde{Z} = [\tilde{z}_1, \dots, \tilde{z}_n]. \\ \Leftrightarrow \min_{z_1, \dots, z_n} \quad & \sum_{j=1}^n \|z_j - \tilde{z}_j\|^2 \\ \text{s.t.} \quad & \mathbf{1}^T z_j = \rho_j, \quad z_j \geq 0, \quad j = 1, \dots, n. \end{aligned} \quad (13)$$

$\text{quadprog}() \Rightarrow Z^+$.

注 (引入 Z 的理由之二)

若未恰当引入分裂变量 Z , X 子问题的求解难度非常之大.

拉格朗日乘子的更新

受 ALM 算法启发, 更新策略可选为

$$\Phi^+ = \Phi - \beta(X^+ - Z^+); \quad (14)$$

若带松弛因子 $\alpha > 0$, 则

$$\Phi^+ = \Phi - \alpha\beta(X^+ - Z^+). \quad (15)$$

停机准则与 KKT 违反度

对 X 子问题:

$$X^{k+1} = \arg \min_{X: X\mathbf{1}=\rho, \text{tr}(X)=0} \mathcal{L}_A(X, Z^k, \Phi^k)$$

- 使用更新策略(14):

$$(Z^{k+1} - Z^k)(\beta I - R);$$

- 使用更新策略(15):

$$(Z^{k+1} - Z^k)(\beta I - R), \quad (1 - \alpha)\beta(X^{k+1} - Z^{k+1}).$$

停机准则与 KKT 违反度 (续)

对 Z 子问题:

$$Z^{k+1} = \arg \min_{Z: Z^T \mathbf{1} = \rho, Z \geq 0} \mathcal{L}_A(X^{k+1}, Z, \Phi^k)$$

- 使用更新策略(14):

恰好此部分 KKT 违反度为 0;

- 使用更新策略(15):

$$(1 - \alpha)\beta(X^{k+1} - Z^{k+1}).$$

停机准则与 KKT 违反度 (续)

KKT 违反度:

$$\begin{aligned} t^{k+1} &\triangleq \|X^{k+1} - Z^{k+1}\|_{\infty} && \text{原始残差,} \\ s^{k+1} &\triangleq \|(Z^{k+1} - Z^k)(\beta I - R)\|_{\infty} && \text{对偶残差.} \end{aligned} \quad (16)$$

停机准则:

情形 1 t^{k+1}, s^{k+1} 都足够小;

情形 2 对某个 $p^{k+1} \in (0, 1)$, $p^{k+1}s^{k+1} + (1 - p^{k+1})t^{k+1}$ 足够小.

我们使用

$$E^{k+1} = (1 - p^{k+1})t^{k+1} + p^{k+1}s^{k+1} \quad (17)$$

作为 **KKT 违反度**.

完整算法

算法 2 求解问题 (4) 的 ADMM 算法

输入: $X^0, Z^0, \Phi^0, \beta^0, \epsilon, k := 0, s^0 := 1, t^0 := 1, \alpha > 0$ (默认值为1),
 $p^0 \in (0, 1)$.

输出: X^k, Z^k, Φ^k

- 1: $E^k = (1 - p^k)t^k + p^k s^k$;
- 2: **while** $E^k > \epsilon$ **do**
- 3: 由公式(9),(10)计算 X^{k+1} ;
- 4: 使用 MATLAB 内置函数 `quadprog()` 求解列子问题(13)得到 Z^{k+1} ;
- 5: $\Phi^{k+1} = \Phi^k - \alpha\beta^k(X^{k+1} - Z^{k+1})$;
- 6: 由公式(16)计算 t^{k+1}, s^{k+1} ;
- 7: 更新 β^k 得到 β^{k+1} ;
- 8: 更新 p^k 得到 p^{k+1} ;
- 9: 由公式(17)得到 E^{k+1} ;
- 10: $k := k + 1$;
- 11: **end while**

提纲

- 1 引言
- 2 最优性条件
- 3 算法设计
- 4 收敛性分析**
- 5 数值实验
- 6 总结

收敛性分析

定理 (充分性定理)

假设在算法2的每一步, Z 子问题均精确求解, 且产生的迭代序列 $\{X^k\}, \{Z^k\}, \{\Phi^k\}$ 分别收敛到 X^*, Z^*, Φ^* , 满足 $X^* = Z^*$. 则 (X^*, Z^*, Φ^*) 为问题(4)的稳定点.

关键点:

- ① $X^* \in \arg \min_{X \mathbf{1} = \rho, \text{tr}(X) = 0} f(X, Z^*) - \langle \Phi^*, X - Z^* \rangle;$
- ② $Z^* \in \arg \min_{Z^T \mathbf{1} = \rho, Z \geq 0} f(X^*, Z) - \langle \Phi^*, X^* - Z \rangle.$

提纲

- 1 引言
- 2 最优性条件
- 3 算法设计
- 4 收敛性分析
- 5 数值实验**
- 6 总结

数值实验

使用 MATLAB 内置函数 `randn()` 和 `abs()` 生成随机问题. 小型问题以 $n = 3, 4, 5$ 各一问题为代表, 大型问题以 $n = 20, 30, 40$ 各一问题为代表.

目的:

- 揭示算法的**性质**;
- 说明算法的**有效性**;
- 说明算法的**优越性**.

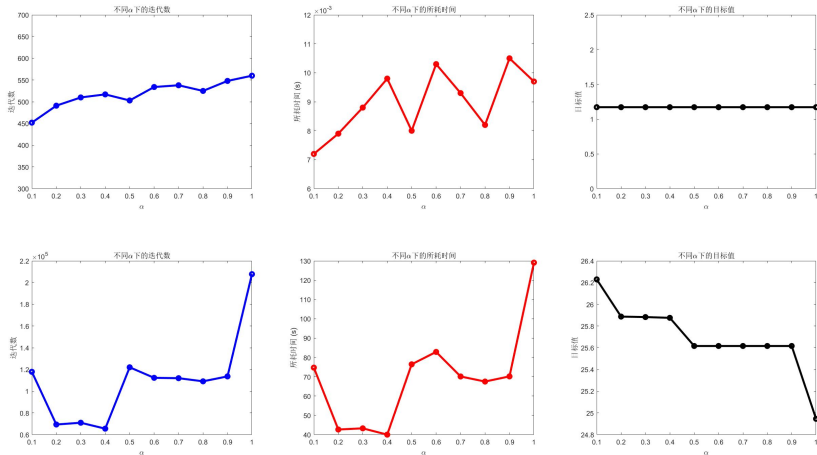
数值实验 - 松弛因子 α $\alpha = 0.1, 0.2, \dots, 1.0$.

图 1: 上 $n=3$, 下 $n=30$, 不同的松弛因子
左: 迭代数; 中: 所耗时间; 右: 目标值.

数值实验 - 惩罚因子 β

表 1: 不同的惩罚因子

β	$n = 3$			
	迭代数	所耗时间 (s)	KKT 违反度	目标值
100	224	0.0043	5.46×10^{-9}	1.1722
1000	560	0.0095	8.02×10^{-9}	1.1722
10000	3998	0.0647	4.92×10^{-9}	1.1722
100000	38377	0.5818	7.68×10^{-9}	1.1722
β	$n = 5$			
	迭代数	所耗时间 (s)	KKT 违反度	目标值
100	969	0.0465	5.00×10^{-9}	2.7741
1000	860	0.0384	9.54×10^{-9}	2.7741
10000	3857	0.1698	4.64×10^{-9}	2.7741
100000	33470	1.4344	9.47×10^{-9}	2.7741
β	$n = 30$			
	迭代数	所耗时间 (s)	KKT 违反度	目标值
10000	207788	133.6154	6.00×10^{-7}	24.9461
100000	2053406	1327.5987	7.08×10^{-7}	24.9542

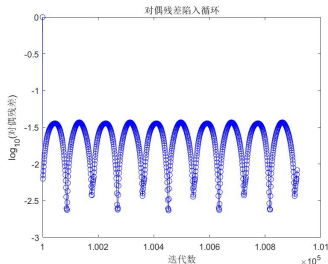
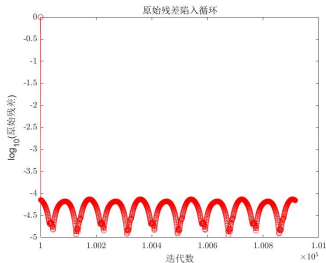
数值实验 - 惩罚因子 β 

图 2: $n = 30, \beta = 10^3$ 残差陷入循环
左: 原始残差; 右: 对偶残差.

数值实验 - 测试问题

给定一数对 $(p, q) : p \neq q, p, q \in \{1, 2, \dots, n\}$, 定义 R 为

$$r_{ij} = \begin{cases} 1, & i = p, j = q \text{ 或 } i = q, j = p, \\ 0, & \text{其它}, \end{cases} \quad \rho := \mathbf{1}. \quad (18)$$

$$\begin{aligned} \min_X \quad & 2x_{pq} + 2 \sum_i x_{ip} x_{iq} \\ \text{s.t.} \quad & X\mathbf{1} = \rho, X^T\mathbf{1} = \rho, X \geq 0, \text{tr}(X) = 0. \end{aligned} \quad (19)$$

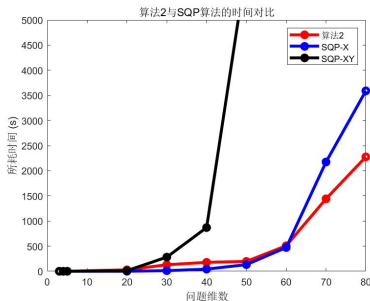
最优值为 0.

表 2: 测试问题, $n = 5, 10, 15, 20$

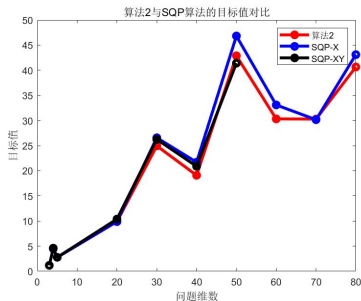
n	迭代数	所耗时间 (s)	KKT 违反度	目标值
5	3187	0.1536	3.00×10^{-9}	-1.44×10^{-11}
10	1447	0.1766	1.65×10^{-9}	-4.94×10^{-15}
15	2243	0.3284	2.30×10^{-9}	6.54×10^{-13}
20	2030	0.3299	4.53×10^{-9}	-8.07×10^{-13}

数值实验 - 与求解非凸二次规划的算法比较

MATLAB 内置函数 `fmincon()`, 调用算法 'sqp' 求解问题(3)和(2), 设置停机准则 'ConstraintTolerance' 为与算法2相同的水平.



(a) 所耗时间



(b) 目标值

图 3: 算法2和 SQP 运行时间对比

结论: 对于大型问题, 算法2更具优势.

提纲

- 1 引言
- 2 最优性条件
- 3 算法设计
- 4 收敛性分析
- 5 数值实验
- 6 总结**

总结

- 引入 ADMM 框架求解问题(4);
- 证明了一定条件下算法2收敛到问题(4)的稳定点;
- 在随机生成的问题上进行数值实验.

特别地:

- 引入变量分裂约束和目标函数, 使问题构造方便算法设计;
- 使用 ADMM 算法求解带特殊约束的问题;
- 简洁地求解了子问题, 简化了 KKT 违反度的计算;
- 给出了算法收敛到稳定点的充分性定理.

感谢聆听!

`huyukuan2015@tongji.edu.cn`