ADMM 算法

王璐

Alternating direction method of multipliers (ADMM) 算法建立在在一些凸优化算法的基础上，如 dual ascent, augmented Lagrangian method 等，它在统计和机器学习问题中有广泛应用，比如 lasso, group lasso, 稀疏协方差矩阵的估计等 (Boyd et al., 2011).

1 Dual Ascent

考虑如下带等式限制条件的凸优化问题：

$$\begin{aligned} &\min_{\boldsymbol{x}} \ f(\boldsymbol{x}) \\ &\text{s.t. } A\boldsymbol{x} = \boldsymbol{b} \end{aligned} \tag{1}$$

其中 $\boldsymbol{x} \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$, $f : \mathbb{R}^n \to \mathbb{R}$ 是一个凸函数。

(1)的 Lagrangian 为

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^\top (A\boldsymbol{x} - \boldsymbol{b}). \tag{2}$$

(1)的 dual objective 为

$$\Theta_D(\boldsymbol{\lambda}) = \min_{\boldsymbol{x}} \ L(\boldsymbol{x}, \boldsymbol{\lambda}). \tag{3}$$

当 strong duality 成立时，primal optimization 和 dual optimization 的函数最优值相等：

$$\min_{\boldsymbol{x}} \left[\max_{\boldsymbol{\lambda}} \ L(\boldsymbol{x}, \boldsymbol{\lambda}) \right] = \max_{\boldsymbol{\lambda}} \ \Theta_D(\boldsymbol{\lambda}). \tag{4}$$

Dual ascent 方法是使用梯度上升法求解 dual optimization (4). 令

$$\boldsymbol{x}^\star = \operatorname*{argmin}_{\boldsymbol{x}} \ L(\boldsymbol{x}, \boldsymbol{\lambda})$$

则

$$\nabla \Theta_D(\boldsymbol{\lambda}) = A\boldsymbol{x}^\star - \boldsymbol{b} \tag{5}$$

即 dual objective (3)的梯度为等式限制条件的残差 (residual). 因此 dual ascent 方法可总结为按如下迭代不断更新 $\boldsymbol{x}$ 和 $\boldsymbol{\lambda}$:

$$\begin{aligned}\boldsymbol{x}^{(t+1)} &= \operatorname*{argmin}_{\boldsymbol{x}} L(\boldsymbol{x}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{\lambda}^{(t+1)} &= \boldsymbol{\lambda}^{(t)} + \alpha_t (A\boldsymbol{x}^{(t+1)} - \boldsymbol{b})\end{aligned} \tag{6}$$

其中 $\alpha_t > 0$ 是第 t 步在梯度方向上移动的步长。如果每步选择合适的步长 α_t, dual objective (3)随迭代进行会不断增大，即 $\Theta_D(\boldsymbol{\lambda}^{(t+1)}) > \Theta_D(\boldsymbol{\lambda}^{(t)})$. 当算法(6)收敛时，$(A\boldsymbol{x}^{(t+1)} - \boldsymbol{b})$ 会收敛到 0，保证得到的解 $\boldsymbol{x}^\star$ 是 primal feasible. 在一些假设条件成立的情况下，比如 f 是有界的严格凸函数，dual ascent 算法(6)最终会收敛到 $(\boldsymbol{x}, \boldsymbol{\lambda})$ 的最优解，但是在很多应用中，这些假设条件并不满足，导致 dual ascent 失效。

2 Augmented Lagrangian and the Method of Multipliers

Augmented Lagrangian 方法可以增强 dual ascent 的稳定性 (robustness), 它可以放松 dual ascent 的一些假设条件，比如严格凸或有界。

优化问题(1)的 augmented Lagrangian 定义为:

$$L_\rho(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^\top (A\boldsymbol{x} - \boldsymbol{b}) + \frac{\rho}{2} \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2 \tag{7}$$

其中 $\rho > 0$ 是惩罚系数。当 $\rho = 0$ 时，(7)退化为标准的 Lagrangian (2).

Augmented Lagrangian (7)可以看作以下优化问题的 Lagrangian 函数:

$$\begin{aligned}&\min_{\boldsymbol{x}} \ f(\boldsymbol{x}) + \frac{\rho}{2} \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2 \\ &\text{s.t. } A\boldsymbol{x} = \boldsymbol{b}\end{aligned} \tag{8}$$

注意到该优化问题(8)与(1)是等价的，因为在满足等式限制条件的 $\boldsymbol{x}$ 中，(8)中的 $\rho/2 \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2$ 对目标函数的贡献为 0. 将(8)的 dual objective 记为:

$$\Theta_{D,\rho}(\boldsymbol{\lambda}) = \min_{\boldsymbol{x}} \ L_\rho(\boldsymbol{x}, \boldsymbol{\lambda}). \tag{9}$$

加入 $\rho/2 \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2$ 的目的是使(8)中的目标函数变为严格凸 (strictly convex) 函数, 以避免在(6)中更新 $\boldsymbol{x}$ 时出现 $\boldsymbol{x}$ 的某些分量为 $\pm\infty$ 的情况（此时可能导致无法更新 $\boldsymbol{\lambda}$)。

令

$$\boldsymbol{x}^\star = \operatorname*{argmin}_{\boldsymbol{x}} L_\rho(\boldsymbol{x}, \boldsymbol{\lambda})$$

则 augmented dual objective (9)的梯度也为

$$\nabla \Theta_{D,\rho}(\boldsymbol{\lambda}) = A\boldsymbol{x}^\star - \boldsymbol{b}.$$

所以对优化问题(8)使用 dual ascent 算法可总结为以下迭代：

$$\begin{aligned} \boldsymbol{x}^{(t+1)} &= \operatorname*{argmin}_{\boldsymbol{x}} L_\rho(\boldsymbol{x}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{\lambda}^{(t+1)} &= \boldsymbol{\lambda}^{(t)} + \rho(A\boldsymbol{x}^{(t+1)} - \boldsymbol{b}) \end{aligned} \tag{10}$$

这里我们将每步步长 α_t 取为 ρ, 原因如下：当 $\boldsymbol{x}^{(t+1)}$ 最小化 $L_\rho(\boldsymbol{x}, \boldsymbol{\lambda}^{(t)})$ 时,

$$\begin{aligned} 0 &= \nabla_{\boldsymbol{x}} L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{\lambda}^{(t)}) \\ &= \nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(t+1)}) + A^\top \left[\boldsymbol{\lambda}^{(t)} + \rho(A\boldsymbol{x}^{(t+1)} - \boldsymbol{b})\right] \\ &= \nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(t+1)}) + A^\top \boldsymbol{\lambda}^{(t+1)}. \end{aligned}$$

因此在算法(10)中使用步长 ρ 可以保证每步更新的 $(\boldsymbol{x}^{(t+1)}, \boldsymbol{\lambda}^{(t+1)})$ 满足优化问题(1)的 Lagrangian stationary 条件：

$$\nabla_{\boldsymbol{x}} L(\boldsymbol{x}^{(t+1)}, \boldsymbol{\lambda}^{(t+1)}) = \nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(t+1)}) + A^\top \boldsymbol{\lambda}^{(t+1)} = 0. \tag{11}$$

算法(10)被称为 method of multipliers. 虽然该方法使 dual ascent 算法更稳定，它也有一个缺点：当目标函数 f separable 时，即

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} f_i(x_i) \tag{12}$$

对应的 augmented Lagrangian L_ρ (7) 并不 separable. 因此在(10)中不能单独更新 $\boldsymbol{x}$ 的各个分量 x_i 进行求解。ADMM 算法通过轮流更新 $\boldsymbol{x}$ 的各个分量将直接求解高维优化问题转化为求解一系列较简单的低维优化问题，提高每步迭代的计算效率。

3 ADMM 算法

考虑具有如下形式的优化问题：

$$\begin{aligned} &\min_{\boldsymbol{x}, \boldsymbol{z}} \ f(\boldsymbol{x}) + g(\boldsymbol{z}) \\ &\text{s.t. } A\boldsymbol{x} + B\boldsymbol{z} = \boldsymbol{c} \end{aligned} \tag{13}$$

其中 f 和 g 都是凸函数，$\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{z} \in \mathbb{R}^m$, $A \in \mathbb{R}^{p\times n}$, $B \in \mathbb{R}^{p\times m}$, $\boldsymbol{c} \in \mathbb{R}^p$. 注意此时(13)中的目标函数关于 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ separable.

写出(13)的 augmented Lagriangian:

$$L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \boldsymbol{\lambda}^\top (A\boldsymbol{x} + B\boldsymbol{z} - \boldsymbol{c}) + \frac{\rho}{2} \|A\boldsymbol{x} + B\boldsymbol{z} - \boldsymbol{c}\|_2^2. \tag{14}$$

ADMM 算法求解(13)的过程可总结为以下迭代：

$$\begin{aligned}\boldsymbol{x}^{(t+1)} &= \underset{\boldsymbol{x}}{\operatorname{argmin}}\ L_\rho(\boldsymbol{x}, \boldsymbol{z}^{(t)}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{z}^{(t+1)} &= \underset{\boldsymbol{z}}{\operatorname{argmin}}\ L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{\lambda}^{(t+1)} &= \boldsymbol{\lambda}^{(t)} + \rho(A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c})\end{aligned} \tag{15}$$

其中 $\rho > 0$. ADMM 算法(15)与 method of multipliers (10)十分相似，不同之处是 ADMM 每步轮流更新变量 $\boldsymbol{x}$ 和 $\boldsymbol{z}$, 而算法(10)联合更新 $(\boldsymbol{x}, \boldsymbol{z})$:

$$\begin{aligned}(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}^{(t+1)}) &= \underset{\boldsymbol{x},\boldsymbol{z}}{\operatorname{argmin}}\ L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{\lambda}^{(t+1)} &= \boldsymbol{\lambda}^{(t)} + \rho(A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c}).\end{aligned}$$

这也正是 ADMM 中 alternating direction 得名的原因。

3.1 Scaled Form

如果对(14)中的 $\boldsymbol{\lambda}$ 做一些放缩，ADMM 算法(15)可写为更容易求解的形式。

将残差记为 $\boldsymbol{r} = A\boldsymbol{x} + B\boldsymbol{z} - \boldsymbol{c}$，则有

$$\begin{aligned}\boldsymbol{\lambda}^\top \boldsymbol{r} + \frac{\rho}{2}\|\boldsymbol{r}\|_2^2 &= \frac{\rho}{2}\left(\boldsymbol{r} + \frac{1}{\rho}\boldsymbol{\lambda}\right)^\top \left(\boldsymbol{r} + \frac{1}{\rho}\boldsymbol{\lambda}\right) - \frac{1}{2\rho}\boldsymbol{\lambda}^\top\boldsymbol{\lambda} \\ &= \frac{\rho}{2}\|\boldsymbol{r} + \boldsymbol{u}\|_2^2 - \frac{\rho}{2}\|\boldsymbol{u}\|_2^2\end{aligned} \tag{16}$$

其中新变量 $\boldsymbol{u} \triangleq \boldsymbol{\lambda}/\rho$, 被称为 scaled dual variables.

将(16)代入 augmented Lagrangian (14), ADMM 算法(15)可写为如下更方便计算的 scaled form:

$$\begin{aligned}\boldsymbol{x}^{(t+1)} &= \underset{\boldsymbol{x}}{\operatorname{argmin}}\ f(\boldsymbol{x}) + \frac{\rho}{2}\left\|A\boldsymbol{x} + B\boldsymbol{z}^{(t)} - \boldsymbol{c} + \boldsymbol{u}^{(t)}\right\|_2^2 \\ \boldsymbol{z}^{(t+1)} &= \underset{\boldsymbol{z}}{\operatorname{argmin}}\ g(\boldsymbol{z}) + \frac{\rho}{2}\left\|A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z} - \boldsymbol{c} + \boldsymbol{u}^{(t)}\right\|_2^2 \\ \boldsymbol{u}^{(t+1)} &= \boldsymbol{u}^{(t)} + A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c}.\end{aligned} \tag{17}$$

将每步的残差记为 $\boldsymbol{r}^{(t)} = A\boldsymbol{x}^{(t)} + B\boldsymbol{z}^{(t)} - \boldsymbol{c}$, 由(17)可得

$$\boldsymbol{u}^{(T)} = \boldsymbol{u}^{(0)} + \sum_{t=1}^{T} \boldsymbol{r}^{(t)}$$

即 $\boldsymbol{u}^{(T)}$ 是前 T 步残差的累加。

3.2 ADMM 收敛性

Boyd et al. (2011) 证明了如下有关 ADMM 收敛性的定理。

Theorem 1. 当优化问题(13)满足以下假设条件时：

- **假设 1**. 函数 $f:\mathbb{R}^n\to\mathbb{R}$ 和 $g:\mathbb{R}^m\to\mathbb{R}$ 是闭凸函数 *(closed convex functions)*.
- **假设 2**. (13)的 *unaugmented Lagrangian* L_0 至少有一个驻点。

ADMM 算法(15)可以保证：

- 残差收敛：$t\to\infty$ 时, $\boldsymbol{r}^{(t)}\to\mathbf{0}$. 即迭代可以保证 $(\boldsymbol{x}^{(t)},\boldsymbol{z}^{(t)})$ 趋于 *primal feasible*.
- 目标函数收敛：$t\to\infty$ 时, $f(\boldsymbol{x}^{(t)})+g(\boldsymbol{z}^{(t)})\to p^\star$, 其中 $p^\star=\inf\{f(\boldsymbol{x})+g(\boldsymbol{z}):A\boldsymbol{x}+B\boldsymbol{z}=\boldsymbol{c}\}$.
- *Dual variable convergence:* $t\to\infty$ 时, $\boldsymbol{\lambda}^{(t)}\to\boldsymbol{\lambda}^\star$, 其中 $\boldsymbol{\lambda}^\star$ 是(13)的一个 *dual optimal point*, 即 $\boldsymbol{\lambda}^\star\in\underset{\boldsymbol{\lambda}}{argmax}\,\Theta_D(\boldsymbol{\lambda})$.

Remarks.

1. 称函数 $f:\mathbb{R}^n\to\mathbb{R}$ 为**闭凸函数**当且仅当集合

$$\{(\boldsymbol{x},t)\in\mathbb{R}^n\times\mathbb{R}:f(\boldsymbol{x})\le t\}$$

是一个非空的闭凸集 (closed convex set)。

2. 假设 1 保证了 ADMM 每步迭代(15)中 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ 都是可解的。

3. 假设 1 并不要求函数 f 或 g 有界，比如 f 可以是如下的 indicator function:

$$f=\begin{cases}0, & \boldsymbol{x}\in\mathcal{C}\\ +\infty, & \boldsymbol{x}\notin\mathcal{C}\end{cases}$$

其中 $\mathcal{C}$ 是一个非空的闭凸集。此时(17)中对 $\boldsymbol{x}$ 的更新即是寻找一个二次函数在 $\mathcal{C}$ 中的最优解。

4. 假设 2 表明存在 $(\boldsymbol{x}^\star,\boldsymbol{z}^\star,\boldsymbol{\lambda}^\star)$ 使得

$$\nabla_{\boldsymbol{x}}L_0(\boldsymbol{x}^\star,\boldsymbol{z}^\star,\boldsymbol{\lambda}^\star)=\nabla f(\boldsymbol{x}^\star)+A^\top\boldsymbol{\lambda}^\star=\mathbf{0} \tag{18}$$

$$\nabla_{\boldsymbol{z}}L_0(\boldsymbol{x}^\star,\boldsymbol{z}^\star,\boldsymbol{\lambda}^\star)=\nabla g(\boldsymbol{z}^\star)+B^\top\boldsymbol{\lambda}^\star=\mathbf{0} \tag{19}$$

$$\nabla_{\boldsymbol{\lambda}}L_0(\boldsymbol{x}^\star,\boldsymbol{z}^\star,\boldsymbol{\lambda}^\star)=A\boldsymbol{x}^\star+B\boldsymbol{z}^\star-\boldsymbol{c}=\mathbf{0}. \tag{20}$$

(20)表明此时 $(\boldsymbol{x}^\star,\boldsymbol{z}^\star)$ 是 primal feasible. (18) - (19)表明 $(\boldsymbol{x}^\star,\boldsymbol{z}^\star,\boldsymbol{\lambda}^\star)$ 满足 Lagrangian stationary 条件，由 KKT 条件可得 $(\boldsymbol{x}^\star,\boldsymbol{z}^\star)$ 是(13)的 primal optimal point, $\boldsymbol{\lambda}^\star$ 是(13)的 dual optimal point.

3.3 ADMM 算法的终止条件 (Stopping Criterion)

根据 KKT 条件，$(\boldsymbol{x}^\star, \boldsymbol{z}^\star, \boldsymbol{\lambda}^\star)$ 是优化问题(13)最优解的充分条件是(18) - (20). 接下来我们检查 ADMM 算法每步更新的 $(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}^{(t+1)}, \boldsymbol{\lambda}^{(t+1)})$ 是否满足这些条件。

在 ADMM 算法(15)中，由于 $\boldsymbol{z}^{(t+1)}$ 最小化 $L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(t)})$，所以

$$\begin{aligned} 0 &= \nabla_{\boldsymbol{z}} L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}^{(t+1)}, \boldsymbol{\lambda}^{(t)}) \\ &= \nabla g(\boldsymbol{z}^{(t+1)}) + B^\top \boldsymbol{\lambda}^{(t)} + \rho B^\top (A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c}) \\ &= \nabla g(\boldsymbol{z}^{(t+1)}) + B^\top (\boldsymbol{\lambda}^{(t)} + \rho \boldsymbol{r}^{(t+1)}) \\ &= \nabla g(\boldsymbol{z}^{(t+1)}) + B^\top \boldsymbol{\lambda}^{(t+1)}. \end{aligned} \tag{21}$$

(21)表明 $\boldsymbol{z}^{(t+1)}$ 和 $\boldsymbol{\lambda}^{(t+1)}$ 总满足条件(19). 这与 method of multipliers 的解总是满足 Lagrangian stationary 条件(11)的证明类似。

在(15)中，$\boldsymbol{x}^{(t+1)}$ 最小化 $L_\rho(\boldsymbol{x}, \boldsymbol{z}^{(t)}, \boldsymbol{\lambda}^{(t)})$, 则有

$$\begin{aligned} 0 &= \nabla_{\boldsymbol{x}} L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}^{(t)}, \boldsymbol{\lambda}^{(t)}) \\ &= \nabla f(\boldsymbol{x}^{(t+1)}) + A^\top \boldsymbol{\lambda}^{(t)} + \rho A^\top (A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t)} - \boldsymbol{c}) \\ &= \nabla f(\boldsymbol{x}^{(t+1)}) + A^\top \left[\boldsymbol{\lambda}^{(t)} + \rho \boldsymbol{r}^{(t+1)} + \rho B(\boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t+1)})\right] \\ &= \nabla f(\boldsymbol{x}^{(t+1)}) + A^\top \boldsymbol{\lambda}^{(t+1)} + \rho A^\top B(\boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t+1)}) \end{aligned}$$

即

$$\rho A^\top B(\boldsymbol{z}^{(t+1)} - \boldsymbol{z}^{(t)}) = \nabla f(\boldsymbol{x}^{(t+1)}) + A^\top \boldsymbol{\lambda}^{(t+1)}. \tag{22}$$

将(22)中等式左边的项记为

$$\boldsymbol{s}^{(t+1)} = \rho A^\top B(\boldsymbol{z}^{(t+1)} - \boldsymbol{z}^{(t)}). \tag{23}$$

称 $\boldsymbol{s}^{(t+1)}$ 为 *dual residual*, 称此时的 $\boldsymbol{r}^{(t+1)} = A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c}$ 为 *primal residual*.

总结一下，ADMM 的解是优化问题(13)的最优解的充分条件是(18) - (20)，第二个条件(19)对每步得到的 $(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}^{(t+1)}, \boldsymbol{\lambda}^{(t+1)})$ 总成立，其它两个条件与 dual residual $\boldsymbol{s}^{(t+1)}$ 和 primal residual $\boldsymbol{r}^{(t+1)}$ 有关。定理1表明随着 ADMM 迭代的进行，残差项 $\boldsymbol{s}^{(t+1)}$ 和 $\boldsymbol{r}^{(t+1)}$ 都会收敛到 $\mathbf{0}$ (Boyd et al. (2011) 在 Appendix A 中证明了 $t \to \infty$ 时，$B(\boldsymbol{z}^{(t+1)} - \boldsymbol{z}^{(t)}) \to \mathbf{0}$). 因此 ADMM 算法一个合理的终止条件是保证 primal and dual residuals 都很小，即

$$\left\|\boldsymbol{r}^{(t+1)}\right\|_2 < \epsilon_p \text{ 且 } \left\|\boldsymbol{s}^{(t+1)}\right\|_2 < \epsilon_d$$

其中 $\epsilon_p > 0$ 和 $\epsilon_d > 0$ 是一些很小的临界值。

3.4 ADMM 应用举例

在很多机器学习的问题中，最小化的目标函数通常由一个可导的损失函数和一个惩罚项组成。ADMM 算法的优势在于它可以将原目标函数拆分为两个独立的函数 f 和 g 分别优化，这为处理不可导的惩罚项，比如 L_1 惩罚项，提供了一种新思路。

3.4.1 加入 L_1 惩罚的统计模型的估计

令 $l(\boldsymbol{\beta})$ 表示一个统计模型的对数似然函数，比如 logistic regression, Poisson regression 或任意广义线性模型，它是一个可导的凹函数。为保证模型的稀疏性，我们在参数估计中加入对系数向量 $\boldsymbol{\beta} \in \mathbb{R}^p$ 的 L_1 范数的惩罚，则 $\boldsymbol{\beta}$ 的最优估计为以下凸优化问题的解：

$$\min_{\boldsymbol{\beta}} \ -l(\boldsymbol{\beta}) + \lambda \left\|\boldsymbol{\beta}\right\|_1 . \tag{24}$$

(24)可以写为如下等价形式，然后使用 ADMM 求解：

$$\begin{aligned} &\min_{\boldsymbol{\beta},\boldsymbol{z}} \ -l(\boldsymbol{\beta}) + g(\boldsymbol{z}) \\ &\text{s.t. } \boldsymbol{\beta} = \boldsymbol{z} \end{aligned} \tag{25}$$

其中 $g(\boldsymbol{z}) = \lambda \left\|\boldsymbol{z}\right\|_1$.

ADMM 求解(25)的迭代格式 (scaled form) 为：

$$\begin{aligned} \boldsymbol{\beta}^{(t+1)} &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \ -l(\boldsymbol{\beta}) + \frac{\rho}{2}\left\|\boldsymbol{\beta} - \boldsymbol{z}^{(t)} + \boldsymbol{u}^{(t)}\right\|_2^2 \\ \boldsymbol{z}^{(t+1)} &= \underset{\boldsymbol{z}}{\operatorname{argmin}} \ \lambda \left\|\boldsymbol{z}\right\|_1 + \frac{\rho}{2}\left\|\boldsymbol{\beta}^{(t+1)} - \boldsymbol{z} + \boldsymbol{u}^{(t)}\right\|_2^2 \\ \boldsymbol{u}^{(t+1)} &= \boldsymbol{u}^{(t)} + \boldsymbol{\beta}^{(t+1)} - \boldsymbol{z}^{(t+1)}. \end{aligned} \tag{26}$$

其中 $\boldsymbol{z}^{(t+1)}$ 的每个分量都有解析解：

$$z_j^{(t+1)} = \operatorname{sign}\left(\beta_j^{(t+1)} + u_j^{(t)}\right)\left(\left|\beta_j^{(t+1)} + u_j^{(t)}\right| - \frac{\lambda}{\rho}\right)_+, \ j = 1, \ldots, p.$$

在(26)中，对 $\boldsymbol{\beta}$ 的更新涉及最小化一个可导的凸函数，因此总可以用一些经典方法求解，比如 Newton 迭代法。

3.4.2 逆协方差矩阵的稀疏估计

假设样本 $\boldsymbol{x}_i \in \mathbb{R}^p$ 服从期望为 $\mathbf{0}$ 的多元正态分布：

$$\boldsymbol{x}_i \overset{iid}{\sim} N_p(\mathbf{0}, \Sigma), \ i = 1, \ldots, n. \tag{27}$$

在逆协方差矩阵 (precision matrix) Σ^{-1} 中，第 (j,k) 元素 $\left(\Sigma^{-1}\right)_{jk}=0$ 表明：给定随机向量 $\boldsymbol{x}$ 其它分量的取值，$\boldsymbol{x}$ 的第 j 和第 k 个分量是条件独立的。很多 graphical models 假设 Σ^{-1} 是一个稀疏矩阵 (Friedman et al., 2008) 以便对条件相关的变量做一个筛选。筛选后，把随机向量 $\boldsymbol{x}$ 的 p 个分量看作 p 个节点，将条件相关的分量用边连接就得到了一个无向图，这是 graphical models 得名的原因。此时 Σ^{-1} 越稀疏得到的无向图就越稀疏。当样本 n 较小时 $(n<p)$，对 Σ^{-1} 做稀疏性假设可以减少参数个数，使估计结果更稳定。

对模型(27)重新参数化，令

$$\Theta=\Sigma^{-1}. \tag{28}$$

则 n 个样本的似然函数为

$$\begin{aligned} l(\Theta) &\propto [\det(\Theta)]^{n/2}\exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\boldsymbol{x}_i^\top\Theta\boldsymbol{x}_i\right\} \\ &\propto [\det(\Theta)]^{n/2}\exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\operatorname{tr}(\boldsymbol{x}_i^\top\Theta\boldsymbol{x}_i)\right\} \\ &\propto [\det(\Theta)]^{n/2}\exp\left\{-\frac{1}{2}\operatorname{tr}\left(\sum_{i=1}^{n}\boldsymbol{x}_i\boldsymbol{x}_i^\top\Theta\right)\right\}. \end{aligned} \tag{29}$$

将样本协方差矩阵记为

$$S=\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i\boldsymbol{x}_i^\top. \tag{30}$$

为得到 Θ 的稀疏估计，考虑惩罚矩阵 Θ 中非对角线 (off-diagonal) 元素的绝对值，则 Θ 的估计值即为以下优化问题的解：

$$\begin{aligned} &\min_{\Theta\in\mathcal{S}_+} \; -\frac{2}{n}\log l(\Theta)+\lambda\left\|\Theta\right\|_1 \\ =&\min_{\Theta\in\mathcal{S}_+} \; -\log\left[\det(\Theta)\right]+\operatorname{tr}(S\Theta)+\lambda\left\|\Theta\right\|_1 \end{aligned} \tag{31}$$

其中 $\mathcal{S}_+$ 表示所有 $p\times p$ 对称正定矩阵的集合，$\left\|\Theta\right\|_1$ 表示 Θ 非对角线元素绝对值的和。(31)是一个凸优化问题，因为函数 $f(\Theta)=-\log\left[\det(\Theta)\right]+\operatorname{tr}(S\Theta)$ 是凸函数。文献中有很多算法可以求解(31)，比如 neighborhood selection with lasso (Meinshausen et al., 2006), graphical lasso (Friedman et al., 2008), interior point algorithm (Yuan and Lin, 2007)，projected subgradient method (Duchi et al., 2012), smoothing method (Lu, 2009) 等。Scheinberg et al. (2010) 使用 ADMM 算法求解(31)，并展示它的效率超越后两种算法。

将优化问题(31)写为以下等价形式：

$$\begin{aligned} &\min_{\Theta\in\mathcal{S}_+} \; -\log\left[\det(\Theta)\right]+\operatorname{tr}(S\Theta)+\lambda\left\|Z\right\|_1 \\ &\text{s.t. } \Theta=Z. \end{aligned} \tag{32}$$

使用 ADMM 求解(32)的迭代格式 (scaled form) 为:

$$\begin{aligned}\Theta^{(t+1)} &= \underset{\Theta\in\mathcal{S}_+}{\operatorname{argmin}} \ -\log\left[\det(\Theta)\right] + \operatorname{tr}(S\Theta) + \frac{\rho}{2}\left\|\Theta - Z^{(t)} + U^{(t)}\right\|_F^2 \\ Z^{(t+1)} &= \underset{Z}{\operatorname{argmin}}\ \lambda\|Z\|_1 + \frac{\rho}{2}\left\|\Theta^{(t+1)} - Z + U^{(t)}\right\|_F^2 \\ U^{(t+1)} &= U^{(t)} + \Theta^{(t+1)} - Z^{(t+1)}\end{aligned} \tag{33}$$

其中 $\|\cdot\|_F$ 为矩阵的 Frobenius norm，即矩阵中所有元素的平方和再开根号。

(33)中对矩阵 Z 每个元素的更新存在解析解:

$$Z_{jk}^{(t+1)} = \begin{cases} \operatorname{sign}\left(\Theta_{jk}^{(t+1)} + U_{jk}^{(t)}\right)\left(\left|\Theta_{jk}^{(t+1)} + U_{jk}^{(t)}\right| - \dfrac{\lambda}{\rho}\right)_+, & j \neq k \\ \Theta_{jk}^{(t+1)} + U_{jk}^{(t)}, & j = k. \end{cases} \tag{34}$$

(33)中对矩阵 Θ 的更新也存在解析解。令 Θ 的一阶导数为零得

$$-\Theta^{-1} + S + \rho(\Theta - Z^{(t)} + U^{(t)}) = \mathbf{0}. \tag{35}$$

由于 Θ 是对称正定矩阵，我们需要找到满足

$$\rho\Theta - \Theta^{-1} = \rho(Z^{(t)} - U^{(t)}) - S \tag{36}$$

的对称正定矩阵 Θ. 由于(36)等式两端都是对称矩阵，所以存在特征值分解，令

$$\rho(Z^{(t)} - U^{(t)}) - S = Q\Lambda Q^\top$$

其中 Q 是单位正交矩阵, $Q^\top Q = QQ^\top = I$, Λ 是对角阵 $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_p)$. 对等式(36)两端左乘 $Q^\top$ 右乘 Q 得

$$\rho Q^\top\Theta Q - Q^\top\Theta^{-1}Q = \Lambda. \tag{37}$$

令 $\tilde{\Theta} = Q^\top\Theta Q$, 则 $\tilde{\Theta}^{-1} = Q^\top\Theta^{-1}Q$. 由(37)得

$$\rho\tilde{\Theta} - \tilde{\Theta}^{-1} = \Lambda. \tag{38}$$

我们可以找到满足(38)的一个对角阵 $\tilde{\Theta}$. 假设 $\tilde{\Theta} = \operatorname{diag}(\theta_1, \ldots, \theta_p)$，则对角线元素满足

$$\rho\theta_j - \frac{1}{\theta_j} = \lambda_j,\ j = 1, \ldots, p. \tag{39}$$

所以 $\theta_j > 0$ 的解为

$$\theta_j = \frac{\lambda_j + \sqrt{\lambda_j^2 + 4\rho}}{2\rho},\ j = 1, \ldots, p.$$

由此得到的 $\Theta = Q\tilde{\Theta}Q^\top$ 一定是对称正定矩阵且满足一阶条件(35)，记为 $\Theta^{(t+1)}$.

References

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122.

Duchi, J., Gould, S., and Koller, D. (2012). Projected subgradient methods for learning sparse gaussians. *arXiv preprint arXiv:1206.3249*.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Lu, Z. (2009). Smooth optimization approach for sparse covariance selection. *SIAM Journal on Optimization*, 19(4):1807–1827.

Meinshausen, N., Bühlmann, P., et al. (2006). High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, 34(3):1436–1462.

Scheinberg, K., Ma, S., and Goldfarb, D. (2010). Sparse inverse covariance selection via alternating linearization methods. In *Advances in neural information processing systems*, pages 2101–2109.

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35.

