第十二章 ADMM 算法

对偶上升法

- **交替方向乘子法 (ADMM)** 建立在在一些凸优化算法的基础上，如对偶上升法 (dual ascent), 加强拉格朗日法 (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
- 考虑以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)的拉格朗日函数为 $L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^\top (A\boldsymbol{x} - \boldsymbol{b})$
  - (1)的对偶目标函数为 $\Theta_D(\boldsymbol{\lambda}) = \min_{\boldsymbol{x}} \ L(\boldsymbol{x}, \boldsymbol{\lambda})$
  - 当强对偶性成立时，

$$\min_{\boldsymbol{x}} \left[\max_{\boldsymbol{\lambda}} \ L(\boldsymbol{x}, \boldsymbol{\lambda}) \right] = \max_{\boldsymbol{\lambda}} \ \Theta_D(\boldsymbol{\lambda}) \tag{2}$$

对偶上升法

对偶上升法是使用梯度上升法求解对偶优化问题 (2)

- 如果 $\boldsymbol{x}^\star = \underset{\boldsymbol{x}}{\operatorname{argmin}}\ L(\boldsymbol{x}, \boldsymbol{\lambda})$，那么 $\nabla \Theta_D(\boldsymbol{\lambda}) = A\boldsymbol{x}^\star - \boldsymbol{b}$
- **对偶上升法**可总结为按如下迭代不断更新 $\boldsymbol{x}$ 和 $\boldsymbol{\lambda}$:

$$\begin{aligned} \boldsymbol{x}^{(t+1)} &= \underset{\boldsymbol{x}}{\operatorname{argmin}}\ L(\boldsymbol{x}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{\lambda}^{(t+1)} &= \boldsymbol{\lambda}^{(t)} + \alpha_t (A\boldsymbol{x}^{(t+1)} - \boldsymbol{b}) \end{aligned} \tag{3}$$

- 如果每步选择合适的步长 α_t, 对偶目标函数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(3)收敛时，$(A\boldsymbol{x}^{(t+1)} - \boldsymbol{b})$ 会收敛到 0，保证得到的解 $\boldsymbol{x}^\star$ 是原始可行的
- 在一些假设条件成立的情况下 (如 f 是有界的严格凸函数)，对偶上升法会收敛到 $(\boldsymbol{x}, \boldsymbol{\lambda})$ 的最优解

加强拉格朗日法和乘子法

- **加强拉格朗日法**可以增强对偶上升法的稳定性，从而放松对偶上升法的一些假设条件，比如严格凸或有界
- 优化问题(1)的加强拉格朗日函数 (augmented Lagrangian)定义为：

$$L_\rho(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^\top (A\boldsymbol{x} - \boldsymbol{b}) + \frac{\rho}{2} \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2 \tag{4}$$

其中 $\rho > 0$ 是惩罚系数

- 加强拉格朗日函数(4)可以看作以下优化问题的拉格朗日函数：

$$\begin{aligned} \min_{\boldsymbol{x}} \ & f(\boldsymbol{x}) + \frac{\rho}{2} \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2 \\ \text{s.t.} \ & A\boldsymbol{x} = \boldsymbol{b} \end{aligned} \tag{5}$$

  - 加入 $\rho/2 \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2$ 的目的：使(5)中的目标函数变为严格凸函数，避免在(3)中更新 $\boldsymbol{x}$ 时出现 $\boldsymbol{x}$ 的某些分量为 $\pm\infty$ 的情况

加强拉格朗日法和乘子法

- 将(5)的对偶目标函数记为：$\Theta_{D,\rho}(\boldsymbol{\lambda}) = \min_{\boldsymbol{x}} \ L_\rho(\boldsymbol{x}, \boldsymbol{\lambda})$
- 若 $\boldsymbol{x}^\star = \underset{\boldsymbol{x}}{\operatorname{argmin}} \ L_\rho(\boldsymbol{x}, \boldsymbol{\lambda})$，上述对偶目标函数的梯度 $\nabla\Theta_{D,\rho}(\boldsymbol{\lambda}) = A\boldsymbol{x}^\star - \boldsymbol{b}$
- 对优化问题(5)使用对偶上升法可总结为以下迭代：

$$\begin{aligned} \boldsymbol{x}^{(t+1)} &= \underset{\boldsymbol{x}}{\operatorname{argmin}} \ L_\rho(\boldsymbol{x}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{\lambda}^{(t+1)} &= \boldsymbol{\lambda}^{(t)} + \rho(A\boldsymbol{x}^{(t+1)} - \boldsymbol{b}) \end{aligned} \tag{6}$$

  - 每步步长取为 ρ 是为了保证每步更新的 $(\boldsymbol{x}^{(t+1)}, \boldsymbol{\lambda}^{(t+1)})$ 满足优化问题(1)的拉格朗日不动性条件：

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

  - 算法(6)被称为乘子法 (method of multipliers)

- **缺点**：当目标函数 f 可分 (separable) 时，即 $f(\boldsymbol{x}) = \sum_{i=1}^n f_i(x_i)$，对应的加强拉格朗日函数 (4) 并不可分，这导致(6)不具有分解和并行计算的能力

ADMM 算法

- ADMM 的提出是为了弥补加强拉格朗日法不能分解的缺点，通过与交替方向法结合实现变量的单独交替迭代
- 考虑具有如下形式的优化问题：

$$\begin{aligned} &\min_{\boldsymbol{x},\boldsymbol{z}} \ f(\boldsymbol{x}) + g(\boldsymbol{z}) \\ &\text{s.t. } A\boldsymbol{x} + B\boldsymbol{z} = \boldsymbol{c} \end{aligned} \tag{7}$$

其中 f 和 g 都是凸函数

- 写出(7)的加强拉格朗日函数:

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

ADMM 算法求解(7)的过程可总结为以下迭代：

$$\begin{aligned} \boldsymbol{x}^{(t+1)} &= \operatorname*{argmin}_{\boldsymbol{x}} \ L_\rho(\boldsymbol{x}, \boldsymbol{z}^{(t)}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{z}^{(t+1)} &= \operatorname*{argmin}_{\boldsymbol{z}} \ L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(t)}) \\ \boldsymbol{\lambda}^{(t+1)} &= \boldsymbol{\lambda}^{(t)} + \rho(A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c}) \end{aligned} \tag{8}$$

  - ADMM 算法与乘子法(6)的区别

Scaled Form

如果对(8)中的 λ 做一些放缩，引入新变量 $\boldsymbol{u} \triangleq \boldsymbol{\lambda}/\rho$，ADMM 算法(8)可写为以下更容易求解的缩放形式 (scaled form):

$$\begin{aligned}
\boldsymbol{x}^{(t+1)} &= \underset{\boldsymbol{x}}{\operatorname{argmin}}\ f(\boldsymbol{x}) + \frac{\rho}{2}\left\|A\boldsymbol{x} + B\boldsymbol{z}^{(t)} - \boldsymbol{c} + \boldsymbol{u}^{(t)}\right\|_2^2 \\
\boldsymbol{z}^{(t+1)} &= \underset{\boldsymbol{z}}{\operatorname{argmin}}\ g(\boldsymbol{z}) + \frac{\rho}{2}\left\|A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z} - \boldsymbol{c} + \boldsymbol{u}^{(t)}\right\|_2^2 \\
\boldsymbol{u}^{(t+1)} &= \boldsymbol{u}^{(t)} + A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c}.
\end{aligned} \tag{9}$$

- 将每步的残差记为 $\boldsymbol{r}^{(t)} = A\boldsymbol{x}^{(t)} + B\boldsymbol{z}^{(t)} - \boldsymbol{c}$, 由(9)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

ADMM 收敛性

Boyd et al. (2011) 证明了以下有关 ADMM 收敛性的定理

定理

当优化问题(7)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
- **假设 2.** (7)的 (非增广) 拉格朗日函数 L_0 至少有一个驻点

ADMM 算法(8)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 $(\boldsymbol{x}^{(t)}, \boldsymbol{z}^{(t)})$ 趋于原始可行
- 目标函数收敛：$t \to \infty$ 时, $f(\boldsymbol{x}^{(t)}) + g(\boldsymbol{z}^{(t)}) \to p^\star$, 其中 $p^\star = \inf\{f(\boldsymbol{x}) + g(\boldsymbol{z}) : A\boldsymbol{x} + B\boldsymbol{z} = \boldsymbol{c}\}$
- 对偶变量收敛: $t \to \infty$ 时, $\boldsymbol{\lambda}^{(t)} \to \boldsymbol{\lambda}^\star$, 其中 $\boldsymbol{\lambda}^\star$ 是(7)对偶问题的一个最优解, 即 $\boldsymbol{\lambda}^\star \in \underset{\boldsymbol{\lambda}}{\operatorname{argmax}}\, \Theta_D(\boldsymbol{\lambda})$

ADMM 收敛性

- 称函数 $f: \mathbb{R}^n \to \mathbb{R}$ 为闭凸函数 (closed convex functions)当且仅当集合

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

  是一个非空的闭凸集 (closed convex set)

- 假设 1 保证了 ADMM 每步迭代(8)中 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ 都是可解的，但假设 1 并不要求函数 f 或 g 有界

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

$$\nabla_{\boldsymbol{x}} L_0(\boldsymbol{x}^\star, \boldsymbol{z}^\star, \boldsymbol{\lambda}^\star) = \nabla f(\boldsymbol{x}^\star) + A^\top \boldsymbol{\lambda}^\star = \mathbf{0} \tag{10}$$

$$\nabla_{\boldsymbol{z}} L_0(\boldsymbol{x}^\star, \boldsymbol{z}^\star, \boldsymbol{\lambda}^\star) = \nabla g(\boldsymbol{z}^\star) + B^\top \boldsymbol{\lambda}^\star = \mathbf{0} \tag{11}$$

$$\nabla_{\boldsymbol{\lambda}} L_0(\boldsymbol{x}^\star, \boldsymbol{z}^\star, \boldsymbol{\lambda}^\star) = A\boldsymbol{x}^\star + B\boldsymbol{z}^\star - \boldsymbol{c} = \mathbf{0}. \tag{12}$$

  由 KKT 条件可得 $(\boldsymbol{x}^\star, \boldsymbol{z}^\star)$ 是(7)的原始问题最优解，$\boldsymbol{\lambda}^\star$ 是(7)的对偶问题最优解

ADMM 算法的终止条件

根据 KKT 条件，$(\boldsymbol{x}^\star, \boldsymbol{z}^\star, \boldsymbol{\lambda}^\star)$ 是优化问题(7)最优解的充分条件是(10) - (12)，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：

- 由于 $\boldsymbol{z}^{(t+1)}$ 最小化 $L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}, \boldsymbol{\lambda}^{(t)})$，所以

$$0 = \nabla_{\boldsymbol{z}} L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}^{(t+1)}, \boldsymbol{\lambda}^{(t)}) = \nabla g(\boldsymbol{z}^{(t+1)}) + B^\top \boldsymbol{\lambda}^{(t+1)}$$

  即 $\boldsymbol{z}^{(t+1)}$ 和 $\boldsymbol{\lambda}^{(t+1)}$ 总满足条件(11)

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

$$0 = \nabla_{\boldsymbol{x}} L_\rho(\boldsymbol{x}^{(t+1)}, \boldsymbol{z}^{(t)}, \boldsymbol{\lambda}^{(t)}) = \nabla f(\boldsymbol{x}^{(t+1)}) + A^\top \boldsymbol{\lambda}^{(t+1)} + \rho A^\top B(\boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t+1)})$$

  令

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

  称 $\boldsymbol{s}^{(t+1)}$ 为**对偶残差**，$\boldsymbol{r}^{(t+1)} = A\boldsymbol{x}^{(t+1)} + B\boldsymbol{z}^{(t+1)} - \boldsymbol{c}$ 为**原始残差**

- ADMM 算法一个合理的终止条件是保证原始和对偶残差都很小，即

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

ADMM 应用举例

- 很多机器学习问题的目标函数由一个可导的损失函数和一个惩罚项成，ADMM 的优势在于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提供了一种新思路
- **加入 L_1 惩罚的统计模型估计**. 为保证模型的稀疏性，系数向量 β 的最优估计为以下凸优化问题的解：

$$\min_{\boldsymbol{\beta}} \ -l(\boldsymbol{\beta}) + \lambda \left\|\boldsymbol{\beta}\right\|_1 \tag{13}$$

  - (13)可以写为如下等价形式，然后使用 ADMM 求解

$$\begin{aligned} &\min_{\boldsymbol{\beta},\boldsymbol{z}} \ -l(\boldsymbol{\beta}) + g(\boldsymbol{z}) \\ &\text{s.t. } \boldsymbol{\beta} = \boldsymbol{z} \end{aligned} \tag{14}$$

    其中 $g(\boldsymbol{z}) = \lambda \left\|\boldsymbol{z}\right\|_1$
  - ADMM 求解(14)的迭代格式 (scaled form) 为：

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

逆协方差矩阵的稀疏估计

- 对于多元正态分布的逆协方差矩阵 (precision matrix) Σ^{-1}，第 (j,k) 元素为 0 表明：给定随机向量 $\boldsymbol{x}$ 其他分量的取值，$\boldsymbol{x}$ 的第 j 和第 k 个分量是条件独立的
- 图模型 (graphical model) 把随机向量 $\boldsymbol{x}$ 的各分量看作节点，将条件相关的分量用边连接得到一个无向图，Σ^{-1} 越稀疏得到的无向图就越稀疏
- 当样本 n 较小时，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
- 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:

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

重新参数化，令 $\Theta = \Sigma^{-1}$，则 n 个样本下 Θ 的似然函数为

$$l(\Theta) \propto [\det(\Theta)]^{n/2} \exp\left\{-\frac{1}{2}\sum_{i=1}^{n} \boldsymbol{x}_i^\top \Theta \boldsymbol{x}_i\right\}$$

逆协方差矩阵的稀疏估计

- 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 的估计值为以下优化问题的解：

$$\begin{aligned} & \min_{\Theta \in \mathcal{S}_+} \; -\frac{2}{n} \log l(\Theta) + \lambda \|\Theta\|_1 \\ = & \min_{\Theta \in \mathcal{S}_+} \; -\log[\det(\Theta)] + \mathrm{tr}(S\Theta) + \lambda \|\Theta\|_1 \end{aligned} \tag{15}$$

  - $S = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top$
  - $\|\Theta\|_1$ 表示 Θ 非对角线元素绝对值的和
  - $\mathcal{S}_+$ 表示所有 $p \times p$ 对称正定矩阵的集合

- 函数 $f(\Theta) = -\log[\det(\Theta)] + \mathrm{tr}(S\Theta)$ 是凸函数，因此(15)是一个凸优化问题，文献中有很多算法可以求解(15), 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., 2008), smoothing method (Lu, 2009) 等，Scheinberg et al. (2010) 使用 ADMM 算法求解(15)，并展示它的效率超越后两种算法

逆协方差矩阵的稀疏估计

- 将优化问题(15)写为以下等价形式：

$$\begin{aligned} \min_{\Theta \in \mathcal{S}_+} \quad & -\log\left[\det(\Theta)\right] + \mathrm{tr}(S\Theta) + \lambda \left\|Z\right\|_1 \\ \text{s.t.} \quad & \Theta = Z \end{aligned} \tag{16}$$

使用 ADMM 求解(16)的迭代格式 (scaled form) 为：

$$\begin{aligned} \Theta^{(t+1)} &= \underset{\Theta \in \mathcal{S}_+}{\mathrm{argmin}} \; -\log\left[\det(\Theta)\right] + \mathrm{tr}(S\Theta) + \frac{\rho}{2}\left\|\Theta - Z^{(t)} + U^{(t)}\right\|_F^2 \\ Z^{(t+1)} &= \underset{Z}{\mathrm{argmin}} \; \lambda \left\|Z\right\|_1 + \frac{\rho}{2}\left\|\Theta^{(t+1)} - Z + U^{(t)}\right\|_F^2 \\ U^{(t+1)} &= U^{(t)} + \Theta^{(t+1)} - Z^{(t+1)} \end{aligned} \tag{17}$$

- (17)中对矩阵 Z 每个元素的更新存在解析解
- (17)中对矩阵 Θ 的更新也存在解析解