# PDM 的优化

Teng Zhang

2021 年 1 月 6 日

目前的优化问题为

$$\min_{\boldsymbol{w}} \ \frac{1}{2m} \sum_{i \in [m]} \max \left\{ 0, 1 - \frac{y_i \boldsymbol{w}^\top \boldsymbol{x}_i}{\bar{\gamma}} \right\}^2, \quad \text{s.t.} \ \bar{\gamma} > 0, \ \|\boldsymbol{w}\|_2 = 1$$

这个形式是有问题的，两个约束都是多余的。注意间隔均值 $\bar{\gamma} = \boldsymbol{w}^\top \overline{y\boldsymbol{x}}$ 关于 $\boldsymbol{w}$ 也是线性的，所以那个比值项是齐次的，放缩 $\boldsymbol{w}$ 不影响目标函数值。若求得的 $\boldsymbol{w}$ 使得 $\bar{\gamma} < 0$，则 $-\boldsymbol{w}$ 可使得 $\bar{\gamma} > 0$ 且不改变目标函数值、不破坏第二个约束。同理第二个约束也是多余的，若求得的 $\boldsymbol{w}$ 范数不为 1，可以将其缩放使范数为 1 且不改变目标函数值。

## 1 重新优化

将两个约束都去掉，问题变成

$$\min_{\boldsymbol{w}} \ \frac{1}{2m} \sum_{i \in [m]} \max \left\{ 0, 1 - \frac{y_i \boldsymbol{w}^\top \boldsymbol{x}_i}{\bar{\gamma}} \right\}^2$$

由于放缩 $\boldsymbol{w}$ 不影响目标函数值，不妨放缩使 $\bar{\gamma} = 1$，于是问题可进一步写成

$$\min_{\boldsymbol{w}} \ \frac{1}{2m} \sum_{i \in [m]} \max \left\{ 0, 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i \right\}^2, \quad \text{s.t.} \ \sum_{i \in [m]} y_i \boldsymbol{w}^\top \boldsymbol{x}_i = m$$

这个形式直觉上也是合理的，最优情况是所有样本间隔均为 1，都等于间隔均值，这样损失为零；若有样本间隔 $> 1$，由于间隔的和为 $m$，必然存在另一个样本间隔 $< 1$，这样就会产生损失。

记 $\epsilon_i = \max\{0, 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i\}$，则问题可写成

$$\min_{\boldsymbol{w}, \epsilon_i} \ \frac{1}{2m} \sum_{i \in [m]} \epsilon_i^2, \quad \text{s.t.} \ \epsilon_i \geq 0, \ \epsilon_i \geq 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i, \ \sum_{i \in [m]} y_i \boldsymbol{w}^\top \boldsymbol{x}_i = m$$

其中约束 $\epsilon_i \geq 0$ 是多余的，因为若求得的 $\epsilon_i < 0$，令其为零不破坏约束 $\epsilon_i \geq 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i$ 且可以使目标函数值变小，因此不存在这种情况。最终的问题为

$$\min_{\boldsymbol{w}, \epsilon_i} \ \frac{1}{2m} \sum_{i \in [m]} \epsilon_i^2, \quad \text{s.t.} \ \epsilon_i \geq 1 - y_i \boldsymbol{w}^\top \boldsymbol{x}_i, \ \sum_{i \in [m]} y_i \boldsymbol{w}^\top \boldsymbol{x}_i = m \tag{1}$$

目标函数是二次的，约束都是线性的，因此这是一个凸二次规划，直接调用成熟的 QP 优化包即可。

记 $\mathbf{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m] \in \mathbb{R}^{d\times m}$，$\mathbf{Y} = \operatorname{diag}\{y_1, y_2, \ldots, y_m\}$，$\boldsymbol{\epsilon} = [\epsilon_1; \epsilon_2; \ldots; \epsilon_m]$，式 (1) 的矩阵形式为

$$\min_{\boldsymbol{w},\boldsymbol{\epsilon}} \ \frac{1}{2m}\boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}, \quad \text{s.t. } \boldsymbol{\epsilon} \geq \boldsymbol{e} - \mathbf{Y}\mathbf{X}^\top \boldsymbol{w},\ \boldsymbol{e}^\top \mathbf{Y}\mathbf{X}^\top \boldsymbol{w} = m$$

记 $\boldsymbol{u} = [\boldsymbol{w}; \boldsymbol{\epsilon}] \in \mathbb{R}^{d+m}$，写成 QP 的标准形式是

$$\min_{\boldsymbol{u}} \ \frac{1}{2}\boldsymbol{u}^\top \begin{bmatrix} \mathbf{0} & \\ & \frac{\mathbf{I}}{m} \end{bmatrix} \boldsymbol{u}, \quad \text{s.t. } [-\mathbf{Y}\mathbf{X}^\top, -\mathbf{I}]\boldsymbol{u} \leq -\boldsymbol{e},\ [\mathbf{X}\mathbf{Y}\boldsymbol{e}; \mathbf{0}]^\top \boldsymbol{u} = m$$

这样就可以直接调包了。

## 2 核化

设 $\kappa(\boldsymbol{x}, \boldsymbol{z}) = \phi(\boldsymbol{x})^\top \phi(\boldsymbol{z})$，记 $\mathbf{X} = [\phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_2), \ldots, \phi(\boldsymbol{x}_m)]$，$\boldsymbol{w}$ 的正交分解为

$$\boldsymbol{w} = \underbrace{a_1\phi(\boldsymbol{x}_1) + a_2\phi(\boldsymbol{x}_2) + \cdots + a_m\phi(\boldsymbol{x}_m)}_{\boldsymbol{\phi}_{\boldsymbol{x}}} + \boldsymbol{\phi}_{\boldsymbol{x}}^\perp = \boldsymbol{\phi}_{\boldsymbol{x}} + \boldsymbol{\phi}_{\boldsymbol{x}}^\perp$$

其中 $\boldsymbol{\phi}_{\boldsymbol{x}}^\perp \in \ker(\mathbf{X}^\top)$。注意 $\boldsymbol{w}$ 在原问题中只以 $\mathbf{X}^\top \boldsymbol{w}$ 的形式出现，又

$$\mathbf{X}^\top \boldsymbol{w} = \mathbf{X}^\top (\boldsymbol{\phi}_{\boldsymbol{x}} + \boldsymbol{\phi}_{\boldsymbol{x}}^\perp) = \mathbf{X}^\top \boldsymbol{\phi}_{\boldsymbol{x}}$$

故原问题最优的 $\boldsymbol{w}$ 不是唯一的，$\boldsymbol{\phi}_{\boldsymbol{x}}^\perp$ 可以取 $\ker(\mathbf{X}^\top)$ 中的任意向量。传统的核方法目标函数中还有 $\boldsymbol{w}$ 的二次正则，所以最小化

$$\|\boldsymbol{w}\|_2^2 = \|\boldsymbol{\phi}_{\boldsymbol{x}}\|^2 + \|\boldsymbol{\phi}_{\boldsymbol{x}}^\perp\|^2 \geq \|\boldsymbol{\phi}_{\boldsymbol{x}}\|^2$$

必然导致 $\boldsymbol{\phi}_{\boldsymbol{x}}^\perp = \mathbf{0}$，从而有

$$\boldsymbol{w} = a_1\phi(\boldsymbol{x}_1) + a_2\phi(\boldsymbol{x}_2) + \cdots + a_m\phi(\boldsymbol{x}_m) = \mathbf{X}\boldsymbol{a}$$

这就是所谓的表示定理。

对于 PDM，没有二次正则，我们就直接取 $\boldsymbol{\phi}_{\boldsymbol{x}}^\perp = \mathbf{0}$ 好了，即 $\boldsymbol{w} = \mathbf{X}\boldsymbol{a}$(注意 $\tilde{\boldsymbol{w}} = \boldsymbol{w} + \ker(\mathbf{X}^\top)$ 其实都是原问题的最优解)，代入原问题有

$$\min_{\boldsymbol{a},\boldsymbol{\epsilon}} \ \frac{1}{2m}\boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}, \quad \text{s.t. } \boldsymbol{\epsilon} \geq \boldsymbol{e} - \mathbf{Y}\mathbf{X}^\top \mathbf{X}\boldsymbol{a},\ \boldsymbol{e}^\top \mathbf{Y}\mathbf{X}^\top \mathbf{X}\boldsymbol{a} = m$$

记 $\mathbf{K} = \mathbf{X}^\top \mathbf{X}$ 为核矩阵，$\boldsymbol{v} = [\boldsymbol{a}; \boldsymbol{\epsilon}] \in \mathbb{R}^{2m}$，写成 QP 的标准形式是

$$\min_{\boldsymbol{v}} \ \frac{1}{2}\boldsymbol{v}^\top \begin{bmatrix} \mathbf{0} & \\ & \frac{\mathbf{I}}{m} \end{bmatrix} \boldsymbol{v}, \quad \text{s.t. } [-\mathbf{Y}\mathbf{K}, -\mathbf{I}]\boldsymbol{v} \leq -\boldsymbol{e},\ [\mathbf{K}\mathbf{Y}\boldsymbol{e}; \mathbf{0}]^\top \boldsymbol{v} = m$$

解出 $\boldsymbol{v}$ 后，对于未知样本 $\boldsymbol{z}$，预测为

$$\begin{aligned} \boldsymbol{w}^\top \phi(\boldsymbol{z}) &= (a_1\phi(\boldsymbol{x}_1) + a_2\phi(\boldsymbol{x}_2) + \cdots + a_m\phi(\boldsymbol{x}_m))^\top \phi(\boldsymbol{z}) \\ &= a_1\kappa(\boldsymbol{x}_1, \boldsymbol{z}) + a_2\kappa(\boldsymbol{x}_2, \boldsymbol{z}) + \cdots + a_m\kappa(\boldsymbol{x}_m, \boldsymbol{z}) \end{aligned}$$

# 3 实验

如果是用 matlab 做实验，图省事可以直接调用 quadprog 函数，这是 matlab 优化工具箱自带的，但 matlab 的版本最好不要晚于 2018，越新越好，最好是 2020b，早期版本的 matlab 的优化工具箱很坑。如果没有新版本的 matlab，可以用 cvx 工具箱 (http://cvxr.com/cvx/)，这是凸优化那本书的作者 Stephen P. Boyd 他们搞的，可以无缝嵌入到 matlab 代码里，很好用。

如果是用 python 做实验，可以用 cvxopt 包 (https://cvxopt.org/userguide/coneprog.html#quadratic-programming)，或者 cvxpy 包 (https://www.cvxpy.org/examples/basic/quadratic_program.html)，后者是 matlab 下的 cvx 工具箱的 python 移植版。