Title

Chao Cheng No.2017310132

February 3, 2021

Contents

1 Algorithm Details

In this section, we will provide the details of each upstating step of the algorithm.

1.1 Update of $\boldsymbol{\beta}$

$$\begin{aligned}\boldsymbol{\beta}^{(k+1)} &= \arg\min_{\boldsymbol{\beta}} L\left(\boldsymbol{\beta}, \boldsymbol{\mu}^{(k)}, \boldsymbol{z}^{(k)}, \boldsymbol{s}^{(k)}, \boldsymbol{w}^{(k)}, \boldsymbol{q}_1^{(k)}, \boldsymbol{q}_2^{(k)}, \boldsymbol{q}_3^{(k)}\right) \\ &= \arg\min_{\boldsymbol{\beta}} \left\{ \begin{array}{l} \frac{r_1}{2}\left\|\boldsymbol{y}-\boldsymbol{\mu}^{(k)}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}^{(k)}\right\|_2^2 + \frac{r_3}{2}\left\|\boldsymbol{\beta}-\boldsymbol{w}^{(k)}\right\|_2^2 \\ +\left\langle \boldsymbol{y}-\boldsymbol{\mu}^{(k)}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}^{(k)}, \boldsymbol{q}_1^{(k)}\right\rangle + \left\langle \boldsymbol{\beta}-\boldsymbol{w}^{(k)}, \boldsymbol{q}_3^{(k)}\right\rangle \end{array} \right\}\end{aligned}$$

This is quadratic in $\boldsymbol{\beta}$ and we can take the first derivative and set it to 0.

$$\begin{aligned}\mathbf{0}_p =& \frac{\partial}{\partial\boldsymbol{\beta}}\left(\frac{r_1}{2}\left\|\boldsymbol{y}-\boldsymbol{u}^{(k)}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}^{(k)}\right\| + \frac{r_3}{2}\left\|\boldsymbol{\beta}-\boldsymbol{w}^{(k)}\right\|_2^2 - \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{q}_1^{(k)} + \boldsymbol{\beta}^T\boldsymbol{q}_3^{(k)}\right) \\ =& \frac{\partial}{\partial\boldsymbol{\beta}}\left(\boldsymbol{\beta}^T\left(\frac{r_1}{2}\boldsymbol{X}^T\boldsymbol{X}+\frac{r_3}{2}\boldsymbol{I}_p\right)\boldsymbol{\beta} + \boldsymbol{\beta}^T\left(-r_1\boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{z}\right)-r_3\boldsymbol{w}-\boldsymbol{X}^T\boldsymbol{q}_1+\boldsymbol{q}_3\right)\right) \\ =& \left(r_1\boldsymbol{X}^T\boldsymbol{X}+r_3\boldsymbol{I}_p\right)\boldsymbol{\beta} + \left(-r_1\boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{z}\right)-r_3\boldsymbol{w}-\boldsymbol{X}^T\boldsymbol{q}_1+\boldsymbol{q}_3\right)\end{aligned}$$

Then the solution is

$$\boldsymbol{\beta}^{(k+1)} = \left(r_1\boldsymbol{X}^T\boldsymbol{X}+r_3\boldsymbol{I}_p\right)^{-1}\left(r_1\boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{z}\right)+r_3\boldsymbol{w}+\boldsymbol{X}^T\boldsymbol{q}_1-\boldsymbol{q}_3\right)$$

In this solution we have to solve the inverse of a $p\times p$ matrix. But if $p > n$ or even $p >> n$, this would require a lot of compuational resources. Nevertheless, if we left multiply $\boldsymbol{X}$ on both side of the first derivative equation, then by some algebra we can show that

$$\boldsymbol{\beta}^{(k+1)} = \frac{1}{r_3}\left[\boldsymbol{I}_p - r_1\boldsymbol{X}^T\left(r_1\boldsymbol{X}\boldsymbol{X}^T+r_3\boldsymbol{I}_n\right)^{-1}\boldsymbol{X}\right]\left(r_1\boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{z}\right)+r_3\boldsymbol{w}+\boldsymbol{X}^T\boldsymbol{q}_1-\boldsymbol{q}_3\right)$$

which only requres to solve the inverse of a $n \times n$ matrix.

1.2 Update of $\boldsymbol{\mu}$

Like in 1.1, we can write

$$\begin{aligned}\boldsymbol{\mu}^{(k+1)} &= \arg\min_{\boldsymbol{\mu}} L\left(\boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{s}, \boldsymbol{w}, \boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3\right) \\ &= \arg\min_{\boldsymbol{\mu}} \left\{ \begin{array}{l} \frac{r_1}{2}\left\|\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}\right\|_2^2 + \frac{r_2}{2}\left\|\boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s}\right\|_2^2 \\ +\left\langle \boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}, \boldsymbol{q}_1\right\rangle + \left\langle \boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s}, \boldsymbol{q}_2\right\rangle \end{array} \right\}\end{aligned}$$

and take the first derivative then set it to 0

$$\begin{aligned}0 &= \frac{\partial}{\partial\boldsymbol{\mu}}\left(\frac{r_1}{2}\left\|\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}\right\|_2^2 + \frac{r_2}{2}\left\|\boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s}\right\|_2^2 - \boldsymbol{\mu}^T\boldsymbol{q}_1 + \boldsymbol{\mu}^T\boldsymbol{D}^T\boldsymbol{q}_2\right) \\ &= \frac{\partial}{\partial\boldsymbol{\mu}}\left(\boldsymbol{\mu}^T\left(\frac{r_1}{2}\boldsymbol{I}_n+\frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{D}\right)\boldsymbol{\mu} + \boldsymbol{\mu}^T\left(-r_1\left(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}\right)-r_2\boldsymbol{D}^T\boldsymbol{s}-\boldsymbol{q}_1+\boldsymbol{D}^T\boldsymbol{q}_2\right)\right) \\ &= \left(r_1\boldsymbol{I}_n+r_2\boldsymbol{D}^T\boldsymbol{D}\right)\boldsymbol{\mu} + \left(-r_1\left(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}\right)-r_2\boldsymbol{D}^T\boldsymbol{s}-\boldsymbol{q}_1+\boldsymbol{D}^T\boldsymbol{q}_2\right)\end{aligned}$$

So the solution is

$$\boldsymbol{\mu} = \left(r_1\boldsymbol{I}_n+r_2\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\left(r_1\left(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{z}\right)+r_2\boldsymbol{D}^T\boldsymbol{s}+\boldsymbol{q}_1-\boldsymbol{D}^T\boldsymbol{q}_2\right)$$

1.3 Update of $\boldsymbol{z}$

Here, we use L-1 loss as an example.

$$\begin{aligned}
\boldsymbol{z}^{(k+1)} &= \arg\min_{\boldsymbol{z}} L\left(\boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{s}, \boldsymbol{w}, \boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3\right) \\
&= \arg\min_{\boldsymbol{z}} \left\{ \frac{1}{n} \sum_{i=1}^{n} |z_i| + \frac{r_1}{2} \|\boldsymbol{y} - \boldsymbol{\mu} - \boldsymbol{X\beta} - \boldsymbol{z}\|_2^2 + \langle \boldsymbol{y} - \boldsymbol{\mu} - \boldsymbol{X\beta} - \boldsymbol{z}, \boldsymbol{q}_1 \rangle \right\} \\
&= \arg\min_{\boldsymbol{z}} \left\{ \frac{1}{n} \sum_{i=1}^{n} |z_i| + \frac{r_1}{2} \left(\boldsymbol{z}^T \boldsymbol{z} - 2\boldsymbol{z}^T \left(\boldsymbol{y} - \boldsymbol{\mu} - \boldsymbol{X}^T \boldsymbol{\beta} \right) \right) - \boldsymbol{z}^T \boldsymbol{q}_1 \right\}
\end{aligned}$$

Note that this problem can be solved coordinate-wisely by

$$z_i^{(k+1)} = \arg\min_{z_i} \frac{1}{n} |z_i| + \frac{r_1}{2} \left(z_i^2 - 2z_i \left(y_i - \mu_i - \boldsymbol{x}_i^T \boldsymbol{\beta} \right) \right) - z_i q_{1,i}$$

By the subgradient method, we have

$$\begin{aligned}
& 0 \in \partial \left(\frac{1}{n} |z_i| + \frac{r_1}{2} \left(z_i^2 - 2z_i \left(y_i - \mu_i - \boldsymbol{x}_i^T \boldsymbol{\beta} \right) \right) - z_i q_{1,i} \right) \Bigg|_{z_i = z_i^{(k+1)}} \\
\Longrightarrow\ & 0 \in \frac{\partial \left(|z_i| \right)\big|_{z_i = z_i^{(k+1)}}}{n} + r_1 \left(z_i^{(k+1)} - y_i + \mu_i + \boldsymbol{x}_i^T \boldsymbol{\beta} - \frac{q_{1,i}}{r_1} \right)
\end{aligned}$$

where

$$\partial \left(|z_i| \right)\big|_{z_i = z_i^{(k+1)}} = \begin{cases} 1 & z_i^{(k+1)} > 0 \\ -1 & z_i^{(k+1)} < 0 \\ [-1, 1] & z_i^{(k+1)} = 0 \end{cases}$$

Therefore

$$\begin{cases}
0 = \frac{1}{n} + r_1 \left(z_i^{(k+1)} - y_i + \mu_i + \boldsymbol{x}_i^T \boldsymbol{\beta} - \frac{q_{1,i}}{r_1} \right) & z_i^{(k+1)} > 0 \\
0 = -\frac{1}{n} + r_1 \left(z_i^{(k+1)} - y_i + \mu_i + \boldsymbol{x}_i^T \boldsymbol{\beta} - \frac{q_{1,i}}{r_1} \right) & z_i^{(k+1)} < 0 \\
0 \in \left[-\frac{1}{n}, \frac{1}{n} \right] + r_1 \left(z_i^{(k+1)} - y_i + \mu_i + \boldsymbol{x}_i^T \boldsymbol{\beta} - \frac{q_{1,i}}{r_1} \right) & z_i^{(k+1)} = 0
\end{cases}$$

Hence

$$z_i^{(k+1)} = \begin{cases}
y_i - \mu_i - \boldsymbol{x}_i^T \boldsymbol{\beta} + \frac{q_{1,i}}{r_1} - \frac{1}{nr_1} & \frac{1}{nr_1} < y_i - \mu_i - \boldsymbol{x}_i^T \boldsymbol{\beta} + \frac{q_{1,i}}{r_1} \\
0 & -\frac{1}{nr_1} \leq y_i - \mu_i - \boldsymbol{x}_i^T \boldsymbol{\beta} + \frac{q_{1,i}}{r_1} \leq \frac{1}{nr_1} \\
y_i - \mu_i - \boldsymbol{x}_i^T \boldsymbol{\beta} + \frac{q_{1,i}}{r_1} + \frac{1}{nr_1} & y_i - \mu_i - \boldsymbol{x}_i^T \boldsymbol{\beta} + \frac{q_{1,i}}{r_1} < -\frac{1}{nr_1}
\end{cases}$$

Here we introduce the soft-thresholding function $S\left(x, \lambda\right)$ for $x \in \mathcal{R}$ and $\lambda > 0$:

$$S\left(x, \lambda\right) = \begin{cases} x - \lambda & \lambda < x \\ 0 & |x| \leq \lambda \\ x + \lambda & x < -\lambda \end{cases}$$

Then we can summary the update of $z_i^{(k+1)}$ by

$$z_i^{(k+1)} = S\left(y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta} + \frac{q_{1,i}}{r_1}, \frac{1}{nr_1}\right)$$

Note: If we use the Huber loss with parameter c instead of the absolute value loss, then similarly the update of $z_i^{(k+1)}$ would be

$$z_i^{(k+1)} = \begin{cases} y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta} + \dfrac{q_{1,i}}{r_1} - \dfrac{c}{nr_1} & \dfrac{c}{nr_1} + c < y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta} + \dfrac{q_{1,i}}{r_1} \\ \dfrac{y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta} + \frac{q_{1,i}}{r_1}}{1 + \frac{1}{nr_1}} & -\dfrac{c}{nr_1} - c \leq y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta} + \dfrac{q_{1,i}}{r_1} \leq \dfrac{c}{nr_1} + c \\ y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta} + \dfrac{q_{1,i}}{r_1} + \dfrac{c}{nr_1} & y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta} + \dfrac{q_{1,i}}{r_1} < -\dfrac{c}{nr_1} - c \end{cases}$$

FYI: The huber loss with parameter c is defined as

$$\rho\left(x; c\right) = \begin{cases} \dfrac{1}{2}x^2 & |x| \leq c \\ c\left|x\right| - \dfrac{1}{2}c^2 & |x| > c \end{cases}$$

1.4 Update of $\boldsymbol{s}$

Here we only consider the SCAD penalty case, which means $P_{\lambda_1}\left(x\right) = P_{\lambda_1,\gamma_1}\left(x\right)$ where γ_1 is a SCAD parameter.

$$\begin{aligned} \boldsymbol{s}^{(k+1)} &= \arg\min_{\boldsymbol{s}} L\left(\boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{s}, \boldsymbol{w}, \boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3\right) \\ &= \arg\min_{\boldsymbol{s}} \left\{\sum_{1\leq i<j\leq n} P_{\lambda_1,\gamma_1}\left(s_{ij}\right) + \frac{r_2}{2}\left\|\boldsymbol{D\mu} - \boldsymbol{s}\right\|_2^2 + \left\langle \boldsymbol{D\mu} - \boldsymbol{s}\right\rangle\right\} \\ &= \arg\min_{\boldsymbol{s}} \left\{\sum_{1\leq i<j\leq n} P_{\lambda_1,\gamma_1}\left(s_{ij}\right) + \frac{r_2}{2}\left(\boldsymbol{s}^T\boldsymbol{s} - 2\boldsymbol{s}^T\boldsymbol{D\mu}\right) - \boldsymbol{s}^T\boldsymbol{q}_2\right\} \end{aligned}$$

Note that $\{s_{ij}\}$ are actually mutually independent in this optimization problem. So like in 1.3, this can also be solved coordinate-wisely. And we try to solve

$$s_{ij}^{(k+1)} = \arg\min_{s_{ij}} \left\{P_{\lambda_1,\gamma_1}\left(s_{ij}\right) + \frac{r_2}{2}\left(s_{ij}^2 - 2s_{ij}\left(\mu_i - \mu_j\right)\right) - q_{2,ij}s_{ij}\right\}$$

and likewise

$$\begin{aligned} & 0 \in \partial\left(P_{\lambda_1,\gamma_1}\left(s_{ij}\right) + \frac{r_2}{2}\left(s_{ij}^2 - 2s_{ij}\left(\mu_i - \mu_j\right)\right) - q_{2,ij}s_{ij}\right)\bigg|_{s_{ij}=s_{ij}^{(k+1)}} \\ \Longrightarrow\ & 0 \in \partial P_{\lambda_1,\gamma_1}\left(s_{ij}\right)\big|_{s_{ij}=s_{ij}^{(k+1)}} + r_2\left(s_{ij}^{(k+1)} - \left(\mu_i - \mu_j\right) - \frac{q_{2,ij}}{r_2}\right) \end{aligned}$$

For the SCAD penalty, we have

$$P'_{\lambda_1,\gamma_1}\left(s_{ij}\right) = \lambda_1\left\{I\left(s_{ij} \leq \lambda_1\right) + \frac{\left(\gamma_1\lambda_1 - s_{ij}\right)_+}{\left(\gamma_1 - 1\right)\lambda_1} I\left(s_{ij} > \lambda_1\right)\right\}$$

for some $\gamma_1 > 2$ and $s_{ij} > 0$. And $P'_{\lambda_1,\gamma_1}(s_{ij}) = -P'_{\lambda_1,\gamma_1}(-s_{ij})$ when $s_{ij} < 0$. Therefore

$$\begin{cases} 0 = 0 + r_2\left(s_{ij}^{(k+1)} - (\mu_i - \mu_j) - \dfrac{q_{2,ij}}{r_2}\right) & \gamma_1\lambda_1 < s_{ij} \\ 0 = \dfrac{\gamma_1\lambda_1 - s_{ij}}{\gamma_1 - 1} + r_2\left(s_{ij}^{(k+1)} - (\mu_i - \mu_j) - \dfrac{q_{2,ij}}{r_2}\right) & \lambda_1 < s_{ij} \le \gamma_1\lambda_1 \\ 0 = \lambda_1 + r_2\left(s_{ij}^{(k+1)} - (\mu_i - \mu_j) - \dfrac{q_{2,ij}}{r_2}\right) & 0 < s_{ij} \le \lambda_1 \\ 0 \in [-\lambda_1, \lambda_1] + r_2\left(s_{ij}^{(k+1)} - (\mu_i - \mu_j) - \dfrac{q_{2,ij}}{r_2}\right) & s_{ij} = 0 \\ 0 = -\lambda_1 + r_2\left(s_{ij}^{(k+1)} - (\mu_i - \mu_j) - \dfrac{q_{2,ij}}{r_2}\right) & -\lambda_1 \le s_{ij} < 0 \\ 0 = -\dfrac{\gamma_1\lambda_1 + s_{ij}}{\gamma_1 - 1} + r_2\left(s_{ij}^{(k+1)} - (\mu_i - \mu_j) - \dfrac{q_{2,ij}}{r_2}\right) & -\gamma_1\lambda_1 \le s_{ij} < -\lambda_1 \\ 0 = 0 + r_2\left(s_{ij}^{(k+1)} - (\mu_i - \mu_j) - \dfrac{q_{2,ij}}{r_2}\right) & s_{ij} \le -\gamma_1\lambda_1 \end{cases}$$

Hence we have

$$s_{ij}^{(k+1)} = \begin{cases} \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} & \gamma_1\lambda_1 < \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} \\ \dfrac{r_2(\gamma_1 - 1)\left(\mu_i - \mu_j + \frac{q_{2,ij}}{r_2}\right) - \gamma_1\lambda_1}{r_2(\gamma_1 - 1) - 1} & \left(1 + \dfrac{1}{r_2}\right)\lambda_1 < \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} \le \gamma_1\lambda_1 \\ \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} - \dfrac{\lambda_1}{r_2} & \dfrac{\lambda_1}{r_2} < \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} \le \left(1 + \dfrac{1}{r_2}\right)\lambda_1 \\ 0 & -\dfrac{\lambda_1}{r_2} \le \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} \le \dfrac{\lambda_1}{r_2} \\ \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} + \dfrac{\lambda_1}{r_2} & -\left(1 + \dfrac{1}{r_2}\right)\lambda_1 \le \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} < -\dfrac{\lambda_1}{r_2} \\ \dfrac{r_2(\gamma_1 - 1)\left(\mu_i - \mu_j + \frac{q_{2,ij}}{r_2}\right) + \gamma_1\lambda_1}{r_2(\gamma_1 - 1) - 1} & -\gamma_1\lambda_1 \le \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} < -\left(1 + \dfrac{1}{r_2}\right)\lambda_1 \\ \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} & \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} \le -\gamma_1\lambda_1 \end{cases}$$

Note: During the deduction, condition $r_2(\gamma_1 - 1) > 1$(i.e. $r_2 > \frac{1}{\gamma_1 - 1}$) is needed to guarantee the result. Also with this condition, the objective function is convex in s_{ij}.

This update can be summaried with the help of the soft-thresholding function

$$s_{ij}^{(k+1)} = \begin{cases} S\left(\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}, \dfrac{\lambda_1}{r_2}\right) & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| \le \left(1 + \dfrac{1}{r_2}\right)\lambda_1 \\ \dfrac{S\left(\mu_i - \mu_j + \frac{q_{2,ij}}{r_2}, \frac{\gamma_1\lambda_1}{r_2(\gamma_1 - 1)}\right)}{1 - \frac{1}{r_2(\gamma_1 - 1)}} & \left(1 + \dfrac{1}{r_2}\right)\lambda_1 < \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| \le \gamma_1\lambda_1 \\ \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| > \gamma_1\lambda_1 \end{cases}$$

For MCP or LASSO, the technique are quite similar. (And the result would be shown below:) If the penalty is Lasso with parameter λ_1 then

$$s_{ij}^{(k+1)} = S\left(\mu_i - \mu_j + \frac{q_{2,ij}}{r_2}, \frac{\lambda_1}{r_2}\right)$$

If the penalty is MCP with parameter λ_1 and γ_1, then the panalty takes the form

$$P_{\lambda_1,\gamma_1}\left(s_{ij}\right) = \lambda_1 \int_0^{s_{ij}} \left(1 - \frac{x}{\lambda_1\gamma_1}\right)_+ \mathrm{d}x$$

for some $\gamma_1 > 1$ and $s_{ij} \geq 0$. And $P'_{\lambda_1,\gamma_1}\left(s_{ij}\right) = -P'_{\lambda_1,\gamma_1}\left(-s_{ij}\right)$ when $s_{ij} < 0$. Therefore the solution is

$$s_{ij}^{(k+1)} = \begin{cases} \dfrac{S\left(\mu_i - \mu_j + \frac{q_{2,ij}}{r_2}, \frac{\lambda_1}{r_2}\right)}{1 - \frac{1}{r_2\gamma_1}} & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| \leq \gamma_1\lambda_1 \\ \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| > \gamma_1\lambda_1 \end{cases}$$

Note: Here the condition $r_2\gamma_1 > 1$(i.e. $r_2 > \frac{1}{\gamma_1}$) is needed, and the objective function is convex in s_{ij}.

1.5 Update of $\boldsymbol{w}$

Like in 1.4, here we consider SCAD penalty as an example. Therefore $P_{\lambda_2}(x) = P_{\lambda_2,\gamma_2}(x)$ where γ_2 is a SCAD parameter.

$$\begin{aligned} \boldsymbol{w}^{(k+1)} &= \arg\min_{\boldsymbol{w}} L\left(\boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{s}, \boldsymbol{w}, \boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3\right) \\ &= \arg\min_{\boldsymbol{w}} \left\{\sum_{j=1}^{p} P_{\lambda_2,\gamma_2}\left(w_j\right) + \frac{r_3}{2}\left\|\boldsymbol{\beta} - \boldsymbol{w}\right\|_2^2 + \left\langle\boldsymbol{\beta} - \boldsymbol{w}, \boldsymbol{q}_3\right\rangle\right\} \\ &= \arg\min_{\boldsymbol{w}} \left\{\sum_{j=1}^{p} P_{\lambda_2,\gamma_2}\left(w_j\right) + \frac{r_3}{2}\left(\boldsymbol{w}^T\boldsymbol{w} - 2\boldsymbol{w}^T\boldsymbol{\beta}\right) - \boldsymbol{w}^T\boldsymbol{q}_3\right\} \end{aligned}$$

Again, this can be solved elementwisely by

$$w_j^{(k+1)} = \arg\min_{w_j} \left\{P_{\lambda_2,\gamma_2}\left(w_j\right) + \frac{r_3}{2}\left(w_j^2 - 2w_j\beta_j\right) - w_j q_{3,j}\right\}$$

Like before, subgradient method provides us

$$\begin{aligned} & 0 \in \partial\left(P_{\lambda_2,\gamma_2}\left(w_j\right) + \frac{r_3}{2}\left(w_j^2 - 2w_j\beta_j\right) - w_j q_{3,j}\right)\Big|_{w_j = w_j^{(k+1)}} \\ \Longrightarrow\ & 0 \in \partial P_{\lambda_2,\gamma_2}\left(w_j\right)\big|_{w_j = w_j^{(k+1)}} + r_3\left(w_j - \beta_j - \frac{q_{3,j}}{r_3}\right) \end{aligned}$$

which means

$$\begin{cases} 0 = 0 + r_3\left(w_j - \beta_j - \dfrac{q_{3,j}}{r_3}\right) & \gamma_2\lambda_2 < w_j \\ 0 = \dfrac{\gamma_2\lambda_2 - w_j}{\gamma_2 - 1} + r_3\left(w_j - \beta_j - \dfrac{q_{3,j}}{r_3}\right) & \lambda_2 < w_j \leq \gamma_2\lambda_2 \\ 0 = \lambda_2 + r_3\left(w_j - \beta_j - \dfrac{q_{3,j}}{r_3}\right) & 0 < w_j \leq \lambda_2 \\ 0 \in [-\lambda_2, \lambda_2] + r_3\left(w_j - \beta_j - \dfrac{q_{3,j}}{r_3}\right) & w_j = 0 \\ 0 = -\lambda_2 + r_3\left(w_j - \beta_j - \dfrac{q_{3,j}}{r_3}\right) & -\lambda_2 \leq w_j < 0 \\ 0 = -\dfrac{\lambda_2\gamma_2 + w_j}{\gamma_2 - 1} + r_3\left(w_j - \beta_j - \dfrac{q_{3,j}}{r_3}\right) & -\gamma_2\lambda_2 \leq w_j < -\lambda_2 \\ 0 = 0 + r_3\left(w_j - \beta_j - \dfrac{q_{3,j}}{r_3}\right) & w_j \leq -\gamma_2\lambda_2 \end{cases}$$

Therefore

$$w_j^{(k+1)} = \begin{cases} \beta_j + \dfrac{q_{3,j}}{r_3} & \gamma_2\lambda_2 < \beta_j + \dfrac{q_{3,j}}{r_3} \\ \dfrac{r_3(\gamma_2 - 1)\left(\beta_j + \frac{q_{3,j}}{r_3}\right) - \gamma_2\lambda_2}{r_3(\gamma_2 - 1) - 1} & \left(1 + \dfrac{1}{r_3}\right)\lambda_2 < \beta_j + \dfrac{q_{3,j}}{r_3} \leq \gamma_2\lambda_2 \\ \beta_j + \dfrac{q_{3,j}}{r_3} - \dfrac{\lambda_2}{r_3} & \dfrac{\lambda_2}{r_3} < \beta_j + \dfrac{q_{3,j}}{r_3} \leq \left(1 + \dfrac{1}{r_3}\right)\lambda_2 \\ 0 & -\dfrac{\lambda_2}{r_3} \leq \beta_j + \dfrac{q_{3,j}}{r_3} \leq \dfrac{\lambda_2}{r_3} \\ \beta_j + \dfrac{q_{3,j}}{r_3} + \dfrac{\lambda_2}{r_3} & -\left(1 + \dfrac{1}{r_3}\right)\lambda_2 \leq \beta_j + \dfrac{q_{3,j}}{r_3} < -\dfrac{\lambda_2}{r_3} \\ \dfrac{r_3(\gamma_2 - 1)\left(\beta_j + \frac{q_{3,j}}{r_3}\right) + \lambda_2\gamma_2}{r_3(\gamma_2 - 1) - 1} & -\gamma_2\lambda_2 \leq \beta_j + \dfrac{q_{3,j}}{r_3} < -\left(1 + \dfrac{1}{r_3}\right)\lambda_2 \\ \beta_j + \dfrac{q_{3,j}}{r_3} & \beta_j + \dfrac{q_{3,j}}{r_3} < -\gamma_2\lambda_2 \end{cases}$$

Note: Here we need $r_3(\gamma_2 - 1) > 1$(i.e. $r_3 > \frac{1}{\gamma_2 - 1}$) to guarantee the resutl.

We can summary the result of $w_j^{(k+1)}$ as

$$w_j^{(k+1)} = \begin{cases} S\left(\beta_j + \dfrac{q_{3,j}}{r_3}, \dfrac{\lambda_2}{r_3}\right) & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| \leq \left(1 + \dfrac{1}{r_3}\right)\lambda_2 \\ \dfrac{S\left(\beta_j + \frac{q_{3,j}}{r_3}, \frac{\gamma_2\lambda_2}{r_3(\gamma_2 - 1)}\right)}{1 - \frac{1}{r_3(\gamma_2 - 1)}} & \left(1 + \dfrac{1}{r_3}\right)\lambda_2 < \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| \leq \gamma_2\lambda_2 \\ \beta_j + \dfrac{q_{3,j}}{r_3} & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| > \gamma_2\lambda_2 \end{cases}$$

Results for MCP and Lasso are presented as following. If the penalty is Lasso with parameter λ_2 then

$$w_j^{(k+1)} = S\left(\beta_j + \frac{q_{3,j}}{r_3}, \frac{\lambda_2}{r_3}\right)$$

If the penalty is MCP with parameter λ_2 and γ_2, then

$$w_j^{(k+1)} = \begin{cases} \dfrac{S\left(\beta_j + \frac{q_{3,j}}{r_3}, \frac{\lambda_2}{r_3}\right)}{1 - \frac{1}{r_3\gamma_2}} & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| \le \gamma_2\lambda_2 \\ \beta_j + \dfrac{q_{3,j}}{r_3} & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| > \gamma_2\lambda_2 \end{cases}$$

1.6 Choice of $\lambda_1^{(0)}$ and $\lambda_2^{(0)}$

Need to be double checked.

Apparently for some large enough $\lambda_1^{(0)}$ and $\lambda_2^{(0)}$, $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ have to be shrinked to $\mathbf{0}_p$ and $\boldsymbol{c}_n$ respectively, where $\mathbf{0}_p$ is a length-p zero vector and $\boldsymbol{c}_n$ is a length-n constant vector. Actually, it's clear that all elements of $\boldsymbol{c}_n$ have to be the medium of $\{y_i,\ i = 1, 2, \cdots, n\}$. Therefore we can write

$$\begin{pmatrix} \boldsymbol{c}_n \\ \mathbf{0}_p \end{pmatrix} \in \arg\min_{\boldsymbol{\mu},\boldsymbol{\beta}} \frac{1}{n}\sum_{i=1}^n \left|y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta}\right| + \sum_{i<j} P_{\lambda_1}(\mu_i - \mu_j) + \sum_{j=1}^p P_{\lambda_2}(\beta_j)$$

which means

$$\begin{pmatrix} \mathbf{0}_n \\ \mathbf{0}_p \end{pmatrix} \in \partial \left\{ \frac{1}{n}\sum_{i=1}^n \left|y_i - \mu_i - \boldsymbol{x}_i^T\boldsymbol{\beta}\right| + \sum_{i<j} P_{\lambda_1}(\mu_i - \mu_j) + \sum_{j=1}^p P_{\lambda_2}(\beta_j) \right\}\Bigg|_{\begin{pmatrix} \boldsymbol{\mu} \\ \boldsymbol{\beta} \end{pmatrix} = \begin{pmatrix} \boldsymbol{c}_n \\ \mathbf{0}_p \end{pmatrix}}$$

That is to say

$$\begin{pmatrix} \mathbf{0}_n \\ \mathbf{0}_p \end{pmatrix} = \begin{pmatrix} \boldsymbol{d}_{1,n} \\ \boldsymbol{d}_{1,p} \end{pmatrix} + \begin{pmatrix} \boldsymbol{d}_{2,n} \\ \mathbf{0}_p \end{pmatrix} + \begin{pmatrix} \mathbf{0}_n \\ \boldsymbol{d}_{3,p} \end{pmatrix}$$

where

$$\begin{pmatrix} \boldsymbol{d}_{1,n} \\ \boldsymbol{d}_{1,p} \end{pmatrix} \in \partial \left\{ \frac{1}{n}\sum_{i=1}^n \left|y_{it} - \mu_i - \boldsymbol{x}_{it}^\top\boldsymbol{\beta}\right| \right\}\Bigg|_{\begin{pmatrix} \boldsymbol{\mu} \\ \boldsymbol{\beta} \end{pmatrix} = \begin{pmatrix} \boldsymbol{c}_n \\ \mathbf{0}_p \end{pmatrix}} = \frac{1}{n}\begin{pmatrix} \partial\left|y_1 - c\right| \\ \vdots \\ \partial\left|y_n - c\right| \\ \sum_{i=1}^n \left(\partial\left|y_i - c\right|\right)\boldsymbol{x}_i \end{pmatrix}$$

$$\boldsymbol{d}_{2,n} \in \partial \sum_{i<j} P_{\lambda_1}(\mu_i - \mu_j)\Bigg|_{\boldsymbol{\mu} = \boldsymbol{c}_n}$$

and

$$\boldsymbol{d}_{3,p} \in \partial \sum_{j=1}^p P_{\lambda_2}(\beta_j)\Bigg|_{\boldsymbol{\beta} = \mathbf{0}_p}$$

Note that

$$\begin{aligned}
&\mathbf{0}_p \in -\boldsymbol{d}_{3,p} + \partial \sum_{j=1}^{p} P_{\lambda_2}\left(\hat{\beta}_j\right) && (\text{where } \hat{\boldsymbol{\beta}} = \mathbf{0}_p) \\
\Longrightarrow\ &\mathbf{0}_p \in \hat{\boldsymbol{\beta}} - \left(\hat{\boldsymbol{\beta}} + \boldsymbol{d}_{3,p}\right) + \partial \sum_{j=1}^{p} P_{\lambda_2}\left(\hat{\beta}_j\right) \\
\Longrightarrow\ &\mathbf{0}_p \in \partial \left\{ \frac{1}{2} \left\| \boldsymbol{\beta} - \left(\hat{\boldsymbol{\beta}} + \boldsymbol{d}_{3,p}\right)\right\|_2^2 + \sum_{j=1}^{p} P_{\lambda_2}\left(\beta_j\right) \right\}\Bigg|_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}} \\
\Longrightarrow\ &\hat{\boldsymbol{\beta}} \in \arg\min_{\boldsymbol{\beta}} \frac{1}{2} \left\| \boldsymbol{\beta} - \left(\hat{\boldsymbol{\beta}} + \boldsymbol{d}_{3,p}\right)\right\|_2^2 + \sum_{j=1}^{p} P_{\lambda_2}\left(\beta_j\right)
\end{aligned}$$

Again, we use SCAD penalty with parameter γ_2 as an example here, and we can solve this penalized OLS by

$$\mathbf{0}_p = \hat{\boldsymbol{\beta}} = T_{\lambda_2,\gamma_2}^{SCAD}\left(\hat{\boldsymbol{\beta}} + \boldsymbol{d}_{3,p}\right) = T_{\lambda_2,\gamma_2}^{SCAD}\left(\boldsymbol{d}_{3,p}\right)$$

and by property of the SCAD thresholding function $T_{\lambda_2,\gamma_2}^{SCAD}$, we have

$$\begin{aligned}
&\left\|\boldsymbol{d}_{3,p}\right\|_\infty \leq \lambda_2 \\
\Longrightarrow\ &\left\|\boldsymbol{d}_{1,p}\right\|_\infty \leq \lambda_2 \qquad (\text{since } \mathbf{0}_p = \boldsymbol{d}_{1,p} + \boldsymbol{d}_{3,p})
\end{aligned}$$

Note that

$$\boldsymbol{d}_{1,p} = \frac{1}{n} \sum_{i=1}^{n} \left(\partial \left|y_i - c\right|\right) \boldsymbol{x}_{it}$$

Therefore

$$\lambda_2^{(0)} \geq \frac{\left\|\boldsymbol{d}_{1,p}\right\|_\infty}{n} = \frac{1}{n} \left\| \sum_{i=1}^{n} \left(\partial \left|y_i - c\right|\right) \boldsymbol{x}_i \right\|_\infty$$

Choose of $\lambda_1^{(0)}$:

$$\boldsymbol{d}_{1,n} \in \frac{1}{n} \begin{pmatrix} \partial\left|y_1 - c\right| \\ \partial\left|y_2 - c\right| \\ \vdots \\ \partial\left|y_n - c\right| \end{pmatrix}$$

$$\boldsymbol{d}_{2,n} \in \partial \sum_{i<j} P_{\lambda_1}\left(\mu_i - \mu_j\right)\Bigg|_{\boldsymbol{\mu}=\boldsymbol{c}_n}$$

and

$$\boldsymbol{d}_{1,n} + \boldsymbol{d}_{2,n} = \mathbf{0}_n$$

By the chain rule we have

$$\begin{aligned} &\partial \sum_{i<j} P_{\lambda_1}\left(\mu_i-\mu_j\right)\Big|_{\boldsymbol{\mu}=\boldsymbol{c}_n} \\ =&\boldsymbol{D}^T \partial \sum_{1\le i<j\le n} P_{\lambda_1}\left(s_{ij}\right)\Big|_{\boldsymbol{s}=\mathbf{0}} \\ \in&\boldsymbol{D}^T\begin{pmatrix}[-\lambda_1,\ \lambda_1]\\ [-\lambda_1,\ \lambda_1]\\ \vdots\\ [-\lambda_1,\ \lambda_1]\end{pmatrix}_{\frac{n(n-1)}{2}\times 1}\end{aligned}$$

Therefore we can deduce

$$\begin{aligned} \left\|\boldsymbol{d}_{2,n}\right\|_\infty &\in \left\|\boldsymbol{D}^\top\begin{pmatrix}[-\lambda_1,\ \lambda_1]\\ [-\lambda_1,\ \lambda_1]\\ \vdots\\ [-\lambda_1,\ \lambda_1]\end{pmatrix}\right\|_\infty \le (n-1)\,\lambda_1 \\ \implies \left\|\boldsymbol{d}_{1,n}\right\|_\infty &\le (n-1)\,\lambda_1 \qquad (\text{since } \mathbf{0}_n=\boldsymbol{d}_{1,n}+\boldsymbol{d}_{2,n})\end{aligned}$$

Note that

$$\boldsymbol{d}_{1,n}\in\frac{1}{n}\begin{pmatrix}\partial\left|y_1-c\right|\\ \partial\left|y_2-c\right|\\ \vdots\\ \partial\left|y_n-c\right|\end{pmatrix}$$

Therefore

$$\lambda_1^{(0)}\ge\frac{\left\|\boldsymbol{d}_{1,n}\right\|_\infty}{n\,(n-1)}=\frac{1}{n\,(n-1)}\left\|\begin{pmatrix}\partial\left|y_1-c\right|\\ \partial\left|y_2-c\right|\\ \vdots\\ \partial\left|y_n-c\right|\end{pmatrix}\right\|_\infty$$

and a sufficient choice would be

$$\lambda_1^{(0)}=\frac{1}{n\,(n-1)}$$

Another way to look at $\lambda_1^{(0)}$:
Note that

$$\boldsymbol{d}_{1,n}+\boldsymbol{d}_{2,n}=\mathbf{0}_n,$$

$$\boldsymbol{d}_{2,n}\in\boldsymbol{D}^T\begin{pmatrix}[-\lambda_1,\lambda_1]\\ [-\lambda_1,\lambda_1]\\ \vdots\\ [-\lambda_1,\lambda_1]\end{pmatrix}_{\frac{n(n-1)}{2}\times 1} \qquad \text{and} \qquad \boldsymbol{d}_{1,n}\in\frac{-1}{n}\begin{pmatrix}\partial\left|y_1-c\right|\\ \partial\left|y_2-c\right|\\ \vdots\\ \partial\left|y_n-c\right|\end{pmatrix}.$$

Also by convex optimization theory, we have $\mathbf{1}^T\boldsymbol{d}_{1,n}=0$. Therefore there exists a n-dinmensional vector $\boldsymbol{\theta}$ such that

$$\left(\boldsymbol{D}^T\boldsymbol{D}\right)\boldsymbol{\theta}=-\boldsymbol{d}_{1,n},$$

since rank $\left(\boldsymbol{D}^T\boldsymbol{D}\right) = n-1$. Hence one possible solution of $\boldsymbol{d}_{2,n}$ is

$$\boldsymbol{d}_{2,n} = -\boldsymbol{D}^T\boldsymbol{D}\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\boldsymbol{d}_{1,n},$$

as long as

$$\left\|\boldsymbol{D}\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\boldsymbol{d}_{1,n}\right\|_\infty \le \lambda_1,$$

where $\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}$ is the Moore-Penrose generalized inverse of $\boldsymbol{D}^T\boldsymbol{D}$. And the choice of $\lambda_1^{(0)}$ is

$$\lambda_1^{(0)} = \left\|\boldsymbol{D}\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\boldsymbol{d}_{1,n}\right\|_\infty$$

Note that it's easy to verity

$$\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1} = \begin{pmatrix} \frac{n-1}{n^2} & \frac{-1}{n^2} & \cdots & \frac{-1}{n^2} \\ \frac{-1}{n^2} & \frac{n-1}{n^2} & \cdots & \frac{-1}{n^2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{-1}{n^2} & \cdots & \frac{-1}{n^2} & \frac{n-1}{n^2} \end{pmatrix},$$

and there are two and only two entries in each row of $\boldsymbol{D}\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}$ with their values being $\frac{1}{n}$ and $\frac{-1}{n}$. Hence we know that

$$\left\|\boldsymbol{D}\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\boldsymbol{d}_{1,n}\right\|_\infty \le \frac{2}{n}\left\|\boldsymbol{d}_{1,n}\right\|_\infty = \frac{2}{n^2}\left\|\begin{pmatrix} \partial\left|y_1-c\right| \\ \partial\left|y_2-c\right| \\ \vdots \\ \partial\left|y_n-c\right| \end{pmatrix}\right\|_\infty,$$

which is a less compuation intensive bound for $\lambda_1^{(0)}$.

1.7 Algorithm with L_2 Loss

If we use L_2 loss, then there is no need for the $\boldsymbol{z}$ part in the algorithm and the augmented lagrangian form would be

$$\begin{aligned} & L\left(\boldsymbol{\beta},\boldsymbol{\mu},\boldsymbol{s},\boldsymbol{w},\boldsymbol{q}_2,\boldsymbol{q}_3\right) \\ =& \frac{1}{n}\sum_{i=1}^n\left(y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}\right)^2 + \sum_{1\le i<j\le n}P_{\lambda_1}\left(s_{ij}\right) + \sum_{j=1}^p P_{\lambda_2}\left(w_j\right) \\ & + \frac{r_2}{2}\left\|\boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s}\right\|_2^2 + \frac{r_3}{2}\left\|\boldsymbol{\beta}-\boldsymbol{w}\right\|_2^2 + \left\langle\boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s},\boldsymbol{q}_2\right\rangle + \left\langle\boldsymbol{\beta}-\boldsymbol{w},\boldsymbol{q}_3\right\rangle \end{aligned}$$

The update of $\boldsymbol{s}$ and $\boldsymbol{w}$ is just the same as before. And the update of $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ is just to find the minimum of

$$\begin{aligned} & \boldsymbol{\beta}^T\left(\frac{1}{n}\boldsymbol{X}^T\boldsymbol{X}+\frac{r_3}{2}\boldsymbol{I}_p\right)\boldsymbol{\beta} + \boldsymbol{\mu}^T\left(\frac{1}{n}\boldsymbol{I}_n+\frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{D}\right)\boldsymbol{\mu} + 2\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{\mu} \\ -& 2\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - r_3\boldsymbol{w}^T\boldsymbol{\beta} + \boldsymbol{q}_3^T\boldsymbol{\beta} - 2\boldsymbol{Y}^T\boldsymbol{\mu} - r_2\boldsymbol{s}^T\boldsymbol{D}\boldsymbol{\mu} + \boldsymbol{q}_2^T\boldsymbol{D}^T\boldsymbol{\mu} \end{aligned}$$

Taking subgradient and set it to $\boldsymbol{0}$ we can find the linear system is

$$\begin{pmatrix} \frac{1}{n}\boldsymbol{I}_n+\frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{D} & \frac{1}{n}\boldsymbol{X} \\ \frac{1}{n}\boldsymbol{X}^T & \frac{1}{n}\boldsymbol{X}^T\boldsymbol{X}+\frac{r_3}{2}\boldsymbol{I}_p \end{pmatrix}\begin{pmatrix}\boldsymbol{\mu} \\ \boldsymbol{\beta}\end{pmatrix} = \begin{pmatrix} \frac{1}{n}\boldsymbol{Y}+\frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{s}-\frac{1}{2}\boldsymbol{D}^T\boldsymbol{q}_2 \\ \frac{1}{n}\boldsymbol{X}^T\boldsymbol{Y}+\frac{r_3}{2}\boldsymbol{w}-\frac{1}{2}\boldsymbol{q}_3 \end{pmatrix}$$

If the dimension is not large, we can directly find the update by

$$\begin{pmatrix}\boldsymbol{\mu}^{(k+1)}\\ \boldsymbol{\beta}^{(k+1)}\end{pmatrix} = \begin{pmatrix}\frac{1}{n}\boldsymbol{I}_n + \frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{D} & \frac{1}{n}\boldsymbol{X}\\ \frac{1}{n}\boldsymbol{X}^T & \frac{1}{n}\boldsymbol{X}^T\boldsymbol{X} + \frac{r_3}{2}\boldsymbol{I}_p\end{pmatrix}^{-1} \begin{pmatrix}\frac{1}{n}\boldsymbol{Y} + \frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{s} - \frac{1}{2}\boldsymbol{D}^T\boldsymbol{q}_2\\ \frac{1}{n}\boldsymbol{X}^T\boldsymbol{Y} + \frac{r_3}{2}\boldsymbol{w} - \frac{1}{2}\boldsymbol{q}_3\end{pmatrix}$$

If the dimension is large, we can use a coordinate descent strategy like before.

$$\boldsymbol{\beta}^{(k+1)} = \left(\frac{1}{n}\boldsymbol{X}^T\boldsymbol{X} + \frac{r_3}{2}\boldsymbol{I}_p\right)^{-1}\left(\frac{1}{n}\boldsymbol{X}^T\boldsymbol{Y} - \frac{1}{n}\boldsymbol{X}^T\boldsymbol{\mu} + \frac{r_3}{2}\boldsymbol{w} - \frac{1}{2}\boldsymbol{q}_3\right)$$

and

$$\boldsymbol{\mu}^{(k+1)} = \left(\frac{1}{n}\boldsymbol{I}_n + \frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\left(\frac{1}{n}\boldsymbol{Y} - \frac{1}{n}\boldsymbol{X}\boldsymbol{\beta} + \frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{s} - \frac{1}{2}\boldsymbol{D}^T\boldsymbol{q}_2\right)$$

A more accurate solution, which is essentially equal to solve the whole linear system would beto be add.

A Re-deduction of the Algorithm with More Parameters

In this section, we again give the details of the algorithm, but with more flexibility in the loss function. We consider the loss function

$$L\left(\boldsymbol{\mu},\boldsymbol{\beta}\right)=\frac{1}{a}\sum_{i=1}^{n}\rho\left(y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}\right)+b\sum_{i<j}P_{\lambda_1}\left(\mu_i-\mu_j\right)+c\sum_{j=1}^{p}P_{\lambda_2}\left(\beta_j\right)$$

where a, b and c are constants. The augmented lagrangian format is

$$\begin{aligned}&L\left(\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{z},\boldsymbol{s},\boldsymbol{w}\right)\\=&\frac{1}{a}\sum_{i=1}^{n}\rho\left(z_i\right)+b\sum_{i<j}P_{\lambda_1}\left(s_{ij}\right)+c\sum_{j=1}^{p}P_{\lambda_2}\left(w_j\right)\\&+\frac{r_1}{2}\left\|\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X\beta}-\boldsymbol{z}\right\|_2^2+\frac{r_2}{2}\left\|\boldsymbol{D\mu}-\boldsymbol{s}\right\|_2^2+\frac{r_3}{2}\left\|\boldsymbol{\beta}-\boldsymbol{w}\right\|_2^2\\&+\left\langle\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X\beta}-\boldsymbol{z},\boldsymbol{q}_1\right\rangle+\left\langle\boldsymbol{D\mu}-\boldsymbol{s},\boldsymbol{q}_2\right\rangle+\left\langle\boldsymbol{\beta}-\boldsymbol{w},\boldsymbol{q}_3\right\rangle\end{aligned}$$

The update of $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ is unchanged as before. But the update of $\boldsymbol{z}$, $\boldsymbol{s}$ and $\boldsymbol{w}$ will be affected by the choice of a, b and c.

A.1 Update steps for $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$

We can update $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ in a coordinate descent fashion. **NOTE:** the previous updating steps of $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ is a special case of coordinate-descent algorithm, with max number of iteration set fixed to 1.

A.1.1 Update $\boldsymbol{\beta}$, Coordinate Descent

If $p\le n$, then

$$\boldsymbol{\beta}^{(k+1)}=\left(r_1\boldsymbol{X}^T\boldsymbol{X}+r_3\boldsymbol{I}_p\right)^{-1}\left(r_1\boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{z}\right)+r_3\boldsymbol{w}+\boldsymbol{X}^T\boldsymbol{q}_1-\boldsymbol{q}_3\right).$$

If $p>n$, then

$$\boldsymbol{\beta}^{(k+1)}=\frac{1}{r_3}\left[\boldsymbol{I}_p-r_1\boldsymbol{X}^T\left(r_1\boldsymbol{X}\boldsymbol{X}^T+r_3\boldsymbol{I}_n\right)^{-1}\boldsymbol{X}\right]\left(r_1\boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{z}\right)+r_3\boldsymbol{w}+\boldsymbol{X}^T\boldsymbol{q}_1-\boldsymbol{q}_3\right).$$

A.1.2 Update $\boldsymbol{\mu}$, Coordinate Descent

The update of $\boldsymbol{\mu}$ is simply

$$\boldsymbol{\mu}=\left(r_1\boldsymbol{I}_n+r_2\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\left(r_1\left(\boldsymbol{y}-\boldsymbol{X\beta}-\boldsymbol{z}\right)+r_2\boldsymbol{D}^T\boldsymbol{s}+\boldsymbol{q}_1-\boldsymbol{D}^T\boldsymbol{q}_2\right).$$

A.1.3 Update $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ simultaneously

It means to minimize the following equation with respect to $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$:

$$
\begin{aligned}
&\frac{r_1}{2}\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}\right)^T\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}\right)+\frac{r_2}{2}\left(\boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s}\right)^T\left(\boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s}\right)+\frac{r_3}{2}\left(\boldsymbol{\beta}-\boldsymbol{w}\right)^T\left(\boldsymbol{\beta}-\boldsymbol{w}\right)\\
&\quad+\left(\boldsymbol{y}-\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}\right)^T\boldsymbol{q}_1+\left(\boldsymbol{D}\boldsymbol{\mu}-\boldsymbol{s}\right)^T\boldsymbol{q}_2+\left(\boldsymbol{\beta}-\boldsymbol{w}\right)^T\boldsymbol{q}_3\\
&\propto\frac{r_1}{2}\left(\boldsymbol{\mu}^T\boldsymbol{\mu}+\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta}-2\boldsymbol{\mu}^T\boldsymbol{y}-2\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{y}+2\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{\mu}+2\boldsymbol{\mu}^T\boldsymbol{z}+2\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{z}\right)\\
&\quad+\frac{r_2}{2}\left(\boldsymbol{\mu}^T\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{\mu}-2\boldsymbol{\mu}^T\boldsymbol{D}^T\boldsymbol{s}\right)+\frac{r_3}{2}\left(\boldsymbol{\beta}^T\boldsymbol{\beta}-2\boldsymbol{\beta}^T\boldsymbol{w}\right)-\boldsymbol{\mu}^T\boldsymbol{q}_1-\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{q}_1+\boldsymbol{\mu}^T\boldsymbol{D}^T\boldsymbol{q}_2+\boldsymbol{\beta}^T\boldsymbol{q}_3\\
&=\begin{pmatrix}\boldsymbol{\mu}^T & \boldsymbol{\beta}^T\end{pmatrix}\begin{pmatrix}\frac{r_1}{2}\boldsymbol{I}_n & \frac{r_1}{2}\boldsymbol{X}\\ \frac{r_1}{2}\boldsymbol{X}^T & \frac{r_1}{2}\boldsymbol{X}^T\boldsymbol{X}\end{pmatrix}\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{\beta}\end{pmatrix}-r_1\left(\boldsymbol{y}-\boldsymbol{z}\right)^T\begin{pmatrix}\boldsymbol{I}_n & \boldsymbol{X}\end{pmatrix}\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{\beta}\end{pmatrix}\\
&\quad+\begin{pmatrix}\boldsymbol{\mu}^T & \boldsymbol{\beta}^T\end{pmatrix}\begin{pmatrix}\frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{D} & \boldsymbol{0}\\ \boldsymbol{0} & \frac{r_3}{2}\boldsymbol{I}_p\end{pmatrix}\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{\beta}\end{pmatrix}-\begin{pmatrix}r_2\boldsymbol{s}^T\boldsymbol{D}+\boldsymbol{q}_1^T-\boldsymbol{q}_2^T\boldsymbol{D} & r_3\boldsymbol{w}^T+\boldsymbol{q}_1^T\boldsymbol{X}-\boldsymbol{q}_3^T\end{pmatrix}\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{\beta}\end{pmatrix}\\
&=\begin{pmatrix}\boldsymbol{\mu}^T & \boldsymbol{\beta}^T\end{pmatrix}\begin{pmatrix}\frac{r_1}{2}\boldsymbol{I}_n+\frac{r_2}{2}\boldsymbol{D}^T\boldsymbol{D} & \frac{r_1}{2}\boldsymbol{X}\\ \frac{r_1}{2}\boldsymbol{X}^T & \frac{r_1}{2}\boldsymbol{X}^T\boldsymbol{X}+\frac{r_3}{2}\boldsymbol{I}_p\end{pmatrix}\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{\beta}\end{pmatrix}\\
&\quad-\left(r_1\left(\boldsymbol{y}-\boldsymbol{z}\right)^T+r_2\boldsymbol{s}^T\boldsymbol{D}+\boldsymbol{q}_1^T-\boldsymbol{q}_2^T\boldsymbol{D},\ \ r_1\left(\boldsymbol{y}-\boldsymbol{z}\right)^T\boldsymbol{X}+r_3\boldsymbol{w}^T+\boldsymbol{q}_1^T\boldsymbol{X}-\boldsymbol{q}_3^T\right)\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{\beta}\end{pmatrix}
\end{aligned}
$$

And the update is

$$
\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{\beta}\end{pmatrix}=\begin{pmatrix}r_1\boldsymbol{I}_n+r_2\boldsymbol{D}^T\boldsymbol{D} & r_1\boldsymbol{X}\\ r_1\boldsymbol{X}^T & r_1\boldsymbol{X}^T\boldsymbol{X}+r_3\boldsymbol{I}_p\end{pmatrix}^{-1}\begin{pmatrix}r_1\left(\boldsymbol{y}-\boldsymbol{z}\right)+r_2\boldsymbol{D}^T\boldsymbol{s}+\boldsymbol{q}_1-\boldsymbol{D}^T\boldsymbol{q}_2\\ r_1\boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{z}\right)+r_3\boldsymbol{w}+\boldsymbol{X}^T\boldsymbol{q}_1-\boldsymbol{q}_3\end{pmatrix}
$$

A.2 Update of $\boldsymbol{z}$

The update of $\boldsymbol{z}$ takes the same form as before, except for replacing n with a. Hence for L_1 loss, it's

$$
z_i^{(k+1)}=S\left(y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}+\frac{q_{1,i}}{r_1},\frac{1}{ar_1}\right)
$$

For Huber loss with parameter c_h, the update is

$$
z_i^{(k+1)}=\begin{cases}
y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}+\dfrac{q_{1,i}}{r_1}-\dfrac{c_h}{ar_1} & \dfrac{c_h}{ar_1}+c_h<y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}+\dfrac{q_{1,i}}{r_1}\\[2ex]
\dfrac{y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}+\frac{q_{1,i}}{r_1}}{1+\frac{1}{ar_1}} & -\dfrac{c_h}{ar_1}-c_h\le y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}+\dfrac{q_{1,i}}{r_1}\le\dfrac{c_h}{ar_1}+c_h\\[2ex]
y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}+\dfrac{q_{1,i}}{r_1}+\dfrac{c_h}{ar_1} & y_i-\mu_i-\boldsymbol{x}_i^T\boldsymbol{\beta}+\dfrac{q_{1,i}}{r_1}<-\dfrac{c_h}{ar_1}-c_h
\end{cases}
$$

A.3 Update of $\boldsymbol{s}$

For Lasso penalty with parameter λ_1, the update is given by

$$
s_{ij}^{(k+1)}=S\left(\mu_i-\mu_j+\frac{q_{2,ij}}{r_2},\frac{b\lambda_1}{r_2}\right).
$$

For SCAD penalty with paramters λ_1 and γ_1, the update is given by

$$s_{ij}^{(k+1)} = \begin{cases} S\left(\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}, \dfrac{b\lambda_1}{r_2}\right) & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| \leq \left(1 + \dfrac{b}{r_2}\right)\lambda_1 \\ \dfrac{S\left(\mu_i - \mu_j + \frac{q_{2,ij}}{r_2}, \frac{b\gamma_1\lambda_1}{r_2(\gamma_1-1)}\right)}{1 - \frac{b}{r_2(\gamma_1-1)}} & \left(1 + \dfrac{b}{r_2}\right)\lambda_1 < \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| \leq \gamma_1\lambda_1 \\ \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| > \gamma_1\lambda_1 \end{cases},$$

and the additional condition to guarantee the convergence during the algorithm is $r_2 > \frac{b}{\gamma_1-1}$.

For MCP penalty with parameters λ_1 and γ_1, the update is

$$s_{ij}^{(k+1)} = \begin{cases} \dfrac{S\left(\mu_i - \mu_j + \frac{q_{2,ij}}{r_2}, \frac{b\lambda_1}{r_2}\right)}{1 - \frac{b}{r_2\gamma_1}} & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| \leq \gamma_1\lambda_1 \\ \mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2} & \left|\mu_i - \mu_j + \dfrac{q_{2,ij}}{r_2}\right| > \gamma_1\lambda_1 \end{cases},$$

and the additional condition is $r_2 > \frac{b}{\gamma_1}$.

A.4 Update of $\boldsymbol{w}$

For Lasso penalty with paramter λ_2, the update is giiven by

$$w_j^{(k+1)} = S\left(\beta_j + \frac{q_{3,j}}{r_3}, \frac{\lambda_2}{r_3}\right).$$

For SCAD penalty with parameters λ_2 and γ_2, the update is given by

$$w_j^{(k+1)} = \begin{cases} S\left(\beta_j + \dfrac{q_{3,j}}{r_3}, \dfrac{c\lambda_2}{r_3}\right) & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| \leq \left(1 + \dfrac{c}{r_3}\right)\lambda_2 \\ \dfrac{S\left(\beta_j + \frac{q_{3,j}}{r_3}, \frac{c\gamma_2\lambda_2}{r_3(\gamma_2-1)}\right)}{1 - \frac{c}{r_3(\gamma_2-1)}} & \left(1 + \dfrac{c}{r_3}\right)\lambda_2 < \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| \leq \gamma_2\lambda_2 \\ \beta_j + \dfrac{q_{3,j}}{r_3} & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| > \gamma_2\lambda_2 \end{cases},$$

and the additional condition is $r_2 > \frac{c}{\gamma_2-1}$.

For MCP penalty with parameters λ_2 and γ_2, the update is given by

$$w_j^{(k+1)} = \begin{cases} \dfrac{S\left(\beta_j + \frac{q_{3,j}}{r_3}, \frac{c\lambda_2}{r_3}\right)}{1 - \frac{c}{r_3\gamma_2}} & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| \leq \gamma_2\lambda_2 \\ \beta_j + \dfrac{q_{3,j}}{r_3} & \left|\beta_j + \dfrac{q_{3,j}}{r_3}\right| > \gamma_2\lambda_2 \end{cases},$$

and the additional condition is $r_3 > \frac{c}{\gamma_2}$.

A.5 Choice of $\lambda_1^{(0)}$ and $\lambda_2^{(0)}$

The initial values of $\lambda_1^{(0)}$ and $\lambda_2^{(0)}$ which shrinks all penalty values to 0 is also affected by the choice of a, b and c, but the deduction is quite similar to before. In summary

$$\lambda_2^{(0)} \geq \frac{1}{ac} \left\| \sum_{i=1}^{n} \psi\left(y_i - m\right) \boldsymbol{x}_i \right\|_\infty ,$$

and

$$\lambda_1^{(0)} \geq \frac{1}{ab} \left\| \boldsymbol{D} \left(\boldsymbol{D}^T \boldsymbol{D}\right)^{-1} \begin{pmatrix} \psi\left(y_1 - m\right) \\ \vdots \\ \psi\left(y_n - m\right) \end{pmatrix} \right\|_\infty ,$$

where $\left(\boldsymbol{D}^T \boldsymbol{D}\right)^{-1}$ is the generalized inverse, $\psi = \partial\rho$ and m is the estimating value of μ when all penalties are pushed to 0, i.e.

$$m = \underset{\mu}{\operatorname{argmin}} \sum_{i=1}^{n} \rho\left(y_i - \mu\right).$$

To simplify the computation, we can choose

$$\lambda_1^{(0)} \geq \frac{2}{abn} \left\| \begin{pmatrix} \psi\left(y_1 - m\right) \\ \vdots \\ \psi\left(y_n - m\right) \end{pmatrix} \right\|_\infty$$