

Learning Deep Sparse Regularizers with Applications to Multi-View Clustering and Semi-Supervised Classification (Appendix)

Shiping Wang, Zhaoliang Chen, Shide Du and Zhouchen Lin, *Fellow, IEEE*

A. Conditions for Equation (9)

Associated with the explicit form (8) of

$$g(x) = \begin{cases} \left(\frac{1}{2w_2} - \frac{1}{2}\right)x^2 + \left(b_2 - \frac{w_1(b_2-b_1)}{w_2}\right)x + \frac{w_1(w_1-w_2)}{2w_2}(b_2-b_1)^2, & x \geq w_1(b_2-b_1), \\ \left(\frac{1}{2w_1} - \frac{1}{2}\right)x^2 + b_1x, & 0 \leq x < w_1(b_2-b_1), \\ g(-x), & x < 0 \end{cases} \quad (1)$$

to be learned, more details for the deduction of conditions (9) for the learned parameters are provided. We consider the following two cases:

Case 1: Suppose $0 \leq x < w_1(b_2 - b_1)$, it is evident that we have $g(x) = \left(\frac{1}{2w_1} - \frac{1}{2}\right)x^2 + b_1x \geq 0$ when $w_1 \in (0, 1]$. If $w_1 \in (1, +\infty)$, the condition for $g(x) \geq 0$ becomes

$$\left(\frac{1}{2w_1} - \frac{1}{2}\right)(w_1(b_2 - b_1))^2 + b_1(w_1(b_2 - b_1)) \geq 0, \quad (2)$$

this is,

$$(w_1 - 1)b_2 - (w_1 + 1)b_1 \leq 0. \quad (3)$$

Case 2: Suppose $x \geq w_1(b_2 - b_1)$, because $\frac{1}{2w_2} - \frac{1}{2} < 0$ when $w_2 \in (1, +\infty)$, it can be verified that $g(x) < 0$ when x is large enough. Therefore we only consider $w_2 \in (0, 1]$. It is required that $g(x)$ is non-decreasing for $x \in [w_1(b_2 - b_1), +\infty)$, leading to $\nabla g(x) \geq 0$. According to the expression of $g(x)$ given in (8), we obtain

$$\nabla g(x) = \left(\frac{1}{w_2} - 1\right)x + \left(b_2 - \frac{w_1(b_2 - b_1)}{w_2}\right). \quad (4)$$

Together with $w_2 \in (0, 1]$, we know that $\nabla g(x)$ is also non-decreasing when $x \geq w_1(b_2 - b_1)$. Therefore, $\nabla g(x) \geq 0$ for $x \in [w_1(b_2 - b_1), +\infty)$ is equivalent to $\nabla g(x)|_{x=w_1(b_2-b_1)} \geq 0$, resulting in

$$\left(\frac{1}{w_2} - 1\right)w_1(b_2 - b_1) + \left(b_2 - \frac{w_1(b_2 - b_1)}{w_2}\right) = -w_1(b_2 - b_1) + b_2 \geq 0, \quad (5)$$

which indicates $b_1 \geq \frac{w_1-1}{w_1}b_2$. Because $g(x)$ is non-decreasing and $g(w_1(b_2 - b_1)) \geq 0$ when $b_1 \geq \frac{w_1-1}{w_1}b_2$, we have $g(x) \geq 0$ in this case. Simultaneously,

$$b_1 \geq \frac{w_1-1}{w_1}b_2 > \frac{w_1-1}{w_1+1}b_2, \quad (6)$$

which also guarantees that Inequality (3) holds.

In summary, combing $w_1, w_2 > 0, b_2 \geq b_1 > 0$ from the aforementioned analyses, the conditions for making $g(x)$ nonnegative become

$$\begin{aligned} w_1 &> 0, 1 \geq w_2 > 0, \\ b_2 &\geq b_1 \geq \max\left\{0, \frac{w_1-1}{w_1}b_2\right\}. \end{aligned} \quad (7)$$

B. Gradients of Learnable Parameters in DSRL

The learnable parameters of the proposed DSRL framework is updated via back propagation, where the loss function is defined as the form

$$\mathcal{J}(\tilde{\mathbf{X}}, \mathbf{X}_{(t)}) = \frac{1}{2} \|\tilde{\mathbf{X}} - \mathbf{X}_{(t)}\|_F^2. \quad (8)$$

We need to optimize parameters $\Theta = [w_1, w_2, b_1, b_2]$ and L according to their gradients. Here, we provide the gradients of these parameters to be learned. For convenience, we denote $\mathbf{X}_{(i)} = [\mathbf{X}_{pq}^{(i)}]_{n \times m}$ for any $i \in \{1, \dots, t\}$, and

$$\mathbf{X}_{(t)} \triangleq \mathbf{X}_{(t)}(\mathbf{X}_{(t-1)}, \Theta) = \mathbf{X}_{(t)}(\mathbf{X}_{(t-1)}(\mathbf{X}_{(t-2)}, \Theta), \Theta) = \mathbf{X}_{(t)}(\mathbf{X}_{(t-1)}(\dots \mathbf{X}_{(1)}(\mathbf{X}_{(0)}, \Theta), \Theta), \Theta) \quad (9)$$

where $\mathbf{X}_{(i)}(\mathbf{X}_{(i-1)}, \Theta) = \xi_{\Theta}(\mathbf{X}_{(i-1)} - \frac{1}{L} \nabla f(\mathbf{X}_{(i-1)}))$. According to the chain rule of multi-variable functions, we know

$$\frac{d\mathcal{J}}{d\Theta} = \frac{d\mathbf{X}_{(t)}}{d\Theta} \frac{d\mathcal{J}}{d\mathbf{X}_{(t)}}, \quad (10)$$

where $\frac{d\mathcal{J}}{d\mathbf{X}_{(t)}} = \mathbf{X}_{(t)} - \tilde{\mathbf{X}}$ is an $n \times m$ matrix, $\frac{d\mathbf{X}_{(t)}}{d\Theta}$ is an $n \times m \times 4$ tensor, and $\frac{d\mathcal{J}}{d\Theta} \in \mathbb{R}^4$ is a column vector. The multiplication between the tensor $\frac{d\mathbf{X}_{(t)}}{d\Theta}$ and the matrix $\frac{d\mathcal{J}}{d\mathbf{X}_{(t)}}$ means the tensor contraction. It is worth pointing out that

$$\frac{d\mathbf{X}_{(t)}}{d\Theta} = \frac{d\mathbf{X}_{(t)}(\mathbf{X}_{(t-1)}, \Theta)}{d\Theta} = \frac{\partial \mathbf{X}_{(t)}}{\partial \Theta} + \frac{d\mathbf{X}_{(t-1)}}{d\Theta} \frac{\partial \mathbf{X}_{(t)}}{\partial \mathbf{X}_{(t-1)}}, \quad (11)$$

where $\frac{d\mathbf{X}_{(t)}(\Theta)}{d\Theta}$ and $\frac{d\mathbf{X}_{(t-1)}}{d\Theta}$ are $n \times m \times 4$ tensors, $\frac{\partial \mathbf{X}_{(t)}}{\partial \mathbf{X}_{(t-1)}}$ is an $n \times m \times n \times m$ four-order tensor, and the multiplication between tensors also means tensor contraction. For any $i \in \{1, \dots, t\}$, the gradient of $\mathbf{X}_{(i)}(\Theta)$ can be calculated via the coordinate-wise derivatives of

$$\frac{\partial [\mathbf{X}_{(i)}(\Theta)]_{pq}}{\partial w_1} = \begin{cases} b_2 - b_1, & b_2 \leq [\mathbf{Z}_{(i)}]_{pq}, \\ [\mathbf{Z}_{(i)}]_{pq} - b_1, & b_1 \leq [\mathbf{Z}_{(i)}]_{pq} < b_2, \\ 0, & -b_1 \leq [\mathbf{Z}_{(i)}]_{pq} < b_1, \\ [\mathbf{Z}_{(i)}]_{pq} + b_1, & -b_2 \leq [\mathbf{Z}_{(i)}]_{pq} < -b_1, \\ b_1 - b_2, & [\mathbf{Z}_{(i)}]_{pq} < -b_2, \end{cases} \quad \frac{\partial [\mathbf{X}_{(i)}(\Theta)]_{pq}}{\partial w_2} = \begin{cases} [\mathbf{Z}_{(i)}]_{pq} - b_2, & b_2 \leq [\mathbf{Z}_{(i)}]_{pq}, \\ 0, & -b_2 \leq [\mathbf{Z}_{(i)}]_{pq} < b_2, \\ [\mathbf{Z}_{(i)}]_{pq} + b_2, & [\mathbf{Z}_{(i)}]_{pq} < -b_2, \end{cases} \quad (12)$$

$$\frac{\partial [\mathbf{X}_{(i)}(\Theta)]_{pq}}{\partial b_1} = \begin{cases} -w_1, & b_1 \leq [\mathbf{Z}_{(i)}]_{pq}, \\ 0, & -b_1 \leq [\mathbf{Z}_{(i)}]_{pq} < b_1, \\ w_1, & [\mathbf{Z}_{(i)}]_{pq} < -b_1, \end{cases} \quad \frac{\partial [\mathbf{X}_{(i)}(\Theta)]_{pq}}{\partial b_2} = \begin{cases} w_1 - w_2, & b_2 \leq [\mathbf{Z}_{(i)}]_{pq}, \\ 0, & -b_2 \leq [\mathbf{Z}_{(i)}]_{pq} < b_2, \\ w_2 - w_1, & [\mathbf{Z}_{(i)}]_{pq} < -b_2, \end{cases} \quad (13)$$

where $\mathbf{Z}_{(i)} = \mathbf{X}_{(i-1)} - \frac{1}{L} \nabla f(\mathbf{X}_{(i-1)})$, $\mathbf{X}_{(i)}(\Theta) \in \mathbb{R}^{n \times m}$ and $\mathbf{Z}_{(i)} \in \mathbb{R}^{n \times m}$ with each entry as $[\mathbf{X}_{(i)}(\Theta)]_{pq}$ and $[\mathbf{Z}_{(i)}]_{pq}$, for all $p \in \{1, \dots, n\}$ and $q \in \{1, \dots, m\}$. Furthermore, for any $i \in \{1, \dots, t\}$, we can compute the $[p, q, j, k]$ -th entry of $\frac{\partial \mathbf{X}_{(i)}}{\partial \mathbf{X}_{(i-1)}}$ by

$$\left[\frac{\partial \mathbf{X}_{(i)}}{\partial \mathbf{X}_{(i-1)}} \right]_{pqjk} = \frac{\partial [\xi_{\Theta}(\mathbf{Z}_{(i)})]_{pq}}{\partial [\mathbf{X}_{(i-1)}]_{jk}} = \begin{cases} w_2 \frac{\partial [\mathbf{Z}_{(i)}]_{pq}}{\partial [\mathbf{X}_{(i-1)}]_{jk}}, & b_2 \leq [\mathbf{Z}_{(i)}]_{pq}, \\ w_1 \frac{\partial [\mathbf{Z}_{(i)}]_{pq}}{\partial [\mathbf{X}_{(i-1)}]_{jk}}, & b_1 \leq [\mathbf{Z}_{(i)}]_{pq} < b_2, \\ 0, & -b_1 \leq [\mathbf{Z}_{(i)}]_{pq} < b_1, \\ w_1 \frac{\partial [\mathbf{Z}_{(i)}]_{pq}}{\partial [\mathbf{X}_{(i-1)}]_{jk}}, & -b_2 \leq [\mathbf{Z}_{(i)}]_{pq} < -b_1, \\ w_2 \frac{\partial [\mathbf{Z}_{(i)}]_{pq}}{\partial [\mathbf{X}_{(i-1)}]_{jk}}, & [\mathbf{Z}_{(i)}]_{pq} < -b_2. \end{cases} \quad (14)$$

For the updating rule of the variable L to be learned, we still have

$$\frac{d\mathcal{J}}{dL} = \frac{d\mathbf{X}_{(t)}}{dL} \frac{d\mathcal{J}}{d\mathbf{X}_{(t)}}, \quad (15)$$

where $\frac{d\mathbf{X}_{(t)}}{dL}$ is regarded as an $n \times m \times 1$ tensor for simplicity. Analogously, it is also noted that $\mathbf{X}_{(i)}$ is a parameterized function of L , then

$$\frac{d\mathbf{X}_{(t)}}{dL} = \frac{d\mathbf{X}_{(t)}(\mathbf{X}_{(t-1)}, L)}{dL} = \frac{\partial \mathbf{X}_{(t)}}{\partial L} + \frac{d\mathbf{X}_{(t-1)}}{dL} \frac{\partial \mathbf{X}_{(t)}}{\partial \mathbf{X}_{(t-1)}}. \quad (16)$$

For any $i \in \{1, \dots, t\}$, while keeping $\mathbf{X}_{(i-1)}$, the gradient of $\mathbf{X}_{(i)}$ can be represented as the following form

$$\frac{\partial [\mathbf{X}_{(i)}(L)]_{pq}}{\partial L} = \begin{cases} w_2 \nabla f(\mathbf{X}_{(i-1)})/L^2, & b_2 \leq [\mathbf{Z}_{(i)}]_{pq}, \\ w_1 \nabla f(\mathbf{X}_{(i-1)})/L^2, & b_1 \leq [\mathbf{Z}_{(i)}]_{pq} < b_2, \\ 0, & -b_1 \leq [\mathbf{Z}_{(i)}]_{pq} < b_1, \\ w_1 \nabla f(\mathbf{X}_{(i-1)})/L^2, & -b_2 \leq [\mathbf{Z}_{(i)}]_{pq} < -b_1, \\ w_2 \nabla f(\mathbf{X}_{(i-1)})/L^2, & [\mathbf{Z}_{(i)}]_{pq} < -b_2. \end{cases} \quad (17)$$

Correspondingly, the total derivatives of \mathcal{J} with respect to Θ and L can be computed recursively using the components as given above. Actually, we did not implement the above derivatives as the deep learning platform (e.g., PyTorch) can compute them automatically via automated differentiation.