D The Solution Algorithm

In this section, we derive the efficient solution Algorithm 1 to minimize the objective in Eq. (4). In Algorithm 1, we repeat the primal-dual updates until the gap in constraints from the augmented Lagrangian terms in Eq. (4) becomes smaller than a predefined tolerance.

W update (without kernel). We discard all terms in Eq. (4) which do not include $\mathbf{W}$ and optimize the columns of $\mathbf{W}$ separately by solving the following K problems for $m = 1, \cdots, K$:

$$\begin{aligned}\mathbf{w}_m^* = & \\ \arg\min_{\mathbf{w}_m} \ & \frac{1}{2}\left\|\mathbf{w}_m\right\|_2^2 + \frac{\mu}{2}\sum_{i=1}^{N}\left[\left\|\mathbf{t}_i^m - \left(\mathbf{w}_m^T\mathbf{X}_i + \mathbf{1}b_m\right) + \boldsymbol{\theta}_i^m/\mu\right\|_2^2\right] \\ + & \sum_{i'=1}^{N'}\sum_{m'=1}^{K}\left[\frac{\mu}{2}\left\|\mathbf{u}_{i'}^{m'} - \left(\mathbf{w}_m^T\mathbf{X}_{i'} + \mathbf{1}b_m\right) + \boldsymbol{\xi}_{i'}^{m'}/\mu\right\|_2^2\right] ,\end{aligned} \tag{5}$$

where N' is the number of bags which belongs to m-th class, and i' denotes the indices of column blocks of $\mathbf{X}$ and the corresponding columns of $\mathbf{U}$ and $\boldsymbol{\Xi}$. Finally $\mathbf{t}_i^m$, $\boldsymbol{\theta}_i^m$, $\mathbf{u}_{i'}^{m'}$, and $\boldsymbol{\xi}_{i'}^{m'}$ are row vectors corresponding to the i-th bag and m-th class in $\mathbf{T}$, $\boldsymbol{\Theta}$, $\mathbf{U}$, and $\boldsymbol{\Xi}$. By letting the derivative of Eq. (5) with respect to $\mathbf{w}_m$ equal zero, we attain the following closed form solution:

$$\begin{aligned}(\mathbf{w}_m^*)^T = & \Big(\textstyle\sum_{i=1}^{N}\left[\left(\mathbf{t}_i^m - \mathbf{1}b_m + \boldsymbol{\theta}_i^m/\mu\right)\mathbf{X}_i^T\right] \\ & + \textstyle\sum_{i'=1}^{N'}\sum_{m'=1}^{K}\left[\left(\mathbf{u}_{i'}^{m'} - \mathbf{1}b_m + \boldsymbol{\xi}_{i'}^{m'}/\mu\right)\mathbf{X}_{i'}^T\right]\Big) \\ & * \left(\mathbf{I}/\mu + \textstyle\sum_{i=1}^{N}\mathbf{X}_i\mathbf{X}_i^T + K\sum_{i'=1}^{N'}\mathbf{X}_{i'}\mathbf{X}_{i'}^T\right)^{-1}.\end{aligned} \tag{6}$$

In the calculation of Eq. (6) we can avoid an inverse calculation through a least-squares solver.

W update (with kernel). The kernel method [18] is widely used in classification tasks to deal with non-linearity. We provide the kernel extension of our method to learn the non-linear relationship between bag and target label. For the arbitrary (possibly non-linear) kernel function ϕ, we map all the columns (instances) of $\mathbf{X}_i \in \Re^{d\times n_i}$ to feature vectors $\phi(\mathbf{X}_i) = \boldsymbol{\Phi}_i \in \Re^{d_z\times n_i}$, and Eq. (5) can be rewritten into:

$$\begin{aligned}\mathbf{w}_m^* = & \\ \arg\min_{\mathbf{w}_m} \ & \frac{1}{2}\left\|\mathbf{w}_m\right\|_2^2 + \frac{\mu}{2}\sum_{i=1}^{N}\left[\left\|\mathbf{t}_i^m - \left(\mathbf{w}_m^T\boldsymbol{\Phi}_i + \mathbf{1}b_m\right) + \boldsymbol{\theta}_i^m/\mu\right\|_2^2\right] \\ + & \sum_{i'=1}^{N'}\sum_{m'=1}^{K}\left[\frac{\mu}{2}\left\|\mathbf{u}_{i'}^{m'} - \left(\mathbf{w}_m^T\boldsymbol{\Phi}_{i'} + \mathbf{1}b_m\right) + \boldsymbol{\xi}_{i'}^{m'}/\mu\right\|_2^2\right].\end{aligned} \tag{7}$$

We take the derivative with respect to $\mathbf{w}_m$ and set it equal to zero to solve for $\mathbf{w}_m$:

$$\begin{aligned}(\mathbf{w}_m^*)^T = \Big(\left[(\mathbf{t}^m - \mathbf{1}b_m + \boldsymbol{\theta}^m/\mu)\,\boldsymbol{\Phi}^T\right] + \textstyle\sum_{m'=1}^{K} \left[(\mathbf{u}_{\prime}^{m'} \right. \\ \left. - \mathbf{1}b_m + \boldsymbol{\xi}_{\prime}^{m'}/\mu)\boldsymbol{\Phi}_{\prime}^T\right]\Big) * \left(\mathbf{I}/\mu + \boldsymbol{\Phi}\boldsymbol{\Phi}^T + K\boldsymbol{\Phi}_{\prime}\boldsymbol{\Phi}_{\prime}^T\right)^{-1} \ ,\end{aligned} \tag{8}$$

where $\boldsymbol{\Phi} = [\boldsymbol{\Phi}_1,\ \ldots, \boldsymbol{\Phi}_N] \in \Re^{d_z \times N_t}$ and $\boldsymbol{\Phi}_{\prime} = [\boldsymbol{\Phi}_{1'},\ \ldots, \boldsymbol{\Phi}_{N'}] \in \Re^{d_z \times N_t'}$. Here $N_t = \sum_{i=1}^{N} n_i$ and $N_t' = \sum_{i'=1}^{N'} n_i$ denote the total number of instances which belongs to all classes and m-th class respectively, and $\boldsymbol{\Phi}_{\prime}$ contains the N' column blocks of $\boldsymbol{\Phi}$ corresponding to the m-th class.

However, the dimensionality d_z of feature vectors $\boldsymbol{\Phi}$ of kernel function can be very large (possibly infinitely large), thus calculating $(\mathbf{I}/\mu + \boldsymbol{\Phi}\boldsymbol{\Phi}^T + K\boldsymbol{\Phi}_{\prime}\boldsymbol{\Phi}_{\prime}^T)^{-1}$ in Eq. (8) may not be computationally feasible. In order to derive the scalable solution against arbitrary kernel function, we rewrite Eq. (8) to the following matrix form:

$$(\mathbf{w}_m^*)^T = \mathbf{s}_m \mathbf{D}\hat{\boldsymbol{\Phi}}^T * \left(\mathbf{I}/\mu + \hat{\boldsymbol{\Phi}}\mathbf{D}\hat{\boldsymbol{\Phi}}^T\right)^{-1} \ , \tag{9}$$

where $\mathbf{s}_m = [\mathbf{t}^m - \mathbf{1}b_m + \boldsymbol{\theta}^m/\mu, \quad 1/K\sum_{m'=1}^{K}(\mathbf{u}^{m'} - \mathbf{1}b_m + \boldsymbol{\xi}_{\prime}^{m'}/\mu)]$, $\mathbf{D} = [\mathbf{I},\ \mathbf{0};\ \mathbf{0},\ K\mathbf{I}]$, and $\hat{\boldsymbol{\Phi}} = [\boldsymbol{\Phi},\ \boldsymbol{\Phi}_{\prime}]$. Then we can apply the following kernel trick [25] to Eq. (9):

$$(\mathbf{P}^{-1} + \mathbf{m}^T\mathbf{R}^{-1}\mathbf{m})^{-1}\mathbf{m}^T\mathbf{R}^{-1} = \mathbf{P}\mathbf{m}^T(\mathbf{m}\mathbf{P}\mathbf{m}^T + \mathbf{R})^{-1} \ ,$$

which gives:

$$(\mathbf{w}_m^*)^T = \mathbf{s}_m(\hat{\boldsymbol{\Phi}}^T\hat{\boldsymbol{\Phi}} + \mathbf{D}^{-1}/\mu)^{-1}\hat{\boldsymbol{\Phi}}^T \ . \tag{10}$$

In Eq. (10), we avoid to compute the feature vectors $\boldsymbol{\Phi}$ in the possibly large dimensionality d_z. Instead we need to compute the inner product of feature vectors $\hat{\boldsymbol{\Phi}}^T\hat{\boldsymbol{\Phi}} \in \Re^{(N_t+N_t')\times(N_t+N_t')}$ which is usually more efficient than directly computing $\boldsymbol{\Phi}\boldsymbol{\Phi}^T \in \Re^{d_z \times d_z}$.

b update. Similarly, differentiating Eq. (5) element-wise with respect to b_m, setting the result equal to zero, gives the update

$$\begin{aligned}b_m = \arg\min_{b_m} \sum_{i=1}^{N} \left[\left\|\mathbf{t}_i^m - \left(\mathbf{w}_m^T\mathbf{X}_i + \mathbf{1}b_m\right) + \theta_i^m/\mu\right\|_2^2\right] \\ + \sum_{i'=1}^{N'}\sum_{m'=1}^{K}\left[\left\|\mathbf{u}_{i'}^{m'} - \left(\mathbf{w}_m^T\mathbf{X}_{i'} + \mathbf{1}b_m\right) + \xi_{i'}^m/\mu\right\|_2^2\right] \ .\end{aligned} \tag{11}$$

Once again, i' indicates the column blocks that belong to the m-th class are chosen from $\mathbf{X}$. Taking the derivative of Eq. (11) with respect to b_m, setting the derivative equal to zero, and solving for b_m gives

$$b_m = \frac{\sum_{i=1}^{N}\left[\mathbf{t}_i^m - \mathbf{w}_m^T\mathbf{X}_i + \boldsymbol{\theta}_i^m/\mu\right] + \sum_{i'=1}^{N'}\sum_{m=1}^{K}\left[\mathbf{u}_{i'}^m - \mathbf{w}_m^T\mathbf{X}_{i'} + \boldsymbol{\xi}_{i'}^m/\mu\right]}{N + KN'} \ . \tag{12}$$

where N' is the total number of patients belonging to the m-th class.

Algorithm 1 The multiblock ADMM updates to optimize Eq. (4)

Data: $\mathbf{X} \in \mathbb{R}^{D\times(n_1+\cdots+n_N)}$ and $\mathbf{Y} \in \{-1,1\}^{K\times N}$.
Hyperparameters: $C > 0$, $\mu > 0$, $\rho > 1$ and $tolerance > 0$.
Initialize: primal variables $\mathbf{W}, \mathbf{b}, \mathbf{E}, \mathbf{Q}, \mathbf{R}, \mathbf{T}, \mathbf{U}$ and dual variables $\boldsymbol{\Lambda}, \boldsymbol{\Sigma}, \boldsymbol{\Theta}, \boldsymbol{\Omega}, \boldsymbol{\Xi}$.
while residual $>$ $tolerance$ **do**
  for $m \in K$ **do**
    Update $\mathbf{w}_m \in \mathbf{W}$ **by** *Eq.* (22).
    Update $b_m \in \mathbf{b}$ **by** $b_m =$

$$\frac{\sum_{i=1}^{N}\left[\mathbf{t}_i^m - \mathbf{w}_m^T\mathbf{X}_i + \boldsymbol{\theta}_i^m/\mu\right] + \sum_{i'=1}^{N'}\sum_{m=1}^{K}\left[\mathbf{u}_{i'}^m - \mathbf{w}_m^T\mathbf{X}_{i'} + \boldsymbol{\xi}_{i'}^m/\mu\right]}{N + KN'} \ .$$

  end for
  for $(p,m) \in \{N,K\}$ **do**
    Update $e_p^m \in \mathbf{E}$ **by** $e_i^m = \begin{cases} n_i^m - \frac{C}{\mu}y_i^m & \text{when} \quad y_i^m n_i^m > \frac{C}{\mu} \ , \\ 0 & \text{when} \quad 0 \le y_i^m n_i^m \le \frac{C}{\mu} \ , \\ n_i^m & \text{when} \quad y_i^m n_i^m < 0 \ , \end{cases}$
    where $n_i^m = y_i^m - q_i^m + r_i^m - \lambda_i^m/\mu$.
    Update $q_p^m \in \mathbf{Q}$ **by** $q_i^m = \frac{\left(y_i^m - e_i^m + r_i^m - \lambda_i^m/\mu + \max\left(\mathbf{t}_i^m\right) - \sigma_i^m/\mu\right)}{2}$.
    Update $r_p^m \in \mathbf{R}$ **by** $r_i^m = \frac{\left(e_i^m - y_i^m + q_i^m + \lambda_i^m/\mu + \max\left(\mathbf{u}_i^m\right) - \omega_i^m/\mu\right)}{2}$.
    for $j \in n_p$ **do**
      Update $t_{p,j}^m \in \mathbf{T}$ **by**

$$t_{i,j}^m = \begin{cases} \frac{\max(\boldsymbol{\phi}_i^m) + q_i^m + \sigma_i^m/\mu}{2} & \text{if } j = \arg\max(\boldsymbol{\phi}_i^m) \ , \\ \phi_{i,j}^m & \text{else} \ , \end{cases}$$

      where $\boldsymbol{\phi}_i^m = \mathbf{w}_m^T\mathbf{X}_i + \mathbf{1}b_m - \boldsymbol{\theta}_i^m/\mu$.
      Update $u_{p,j}^m \in \mathbf{U}$ **by**

$$u_{i,j}^m = \begin{cases} \frac{\max(\boldsymbol{\psi}_i^m) + r_i^m + \omega_i^m/\mu}{2} & \text{if } j = \arg\max(\boldsymbol{\psi}_i^m) \ , \\ \psi_{i,j}^m & \text{else} \ , \end{cases}$$

      where $\boldsymbol{\psi}_i^m = \mathbf{w}_y^T\mathbf{X}_i + \mathbf{1}b_y - \boldsymbol{\xi}_i^m/\mu$.
    end for
    Update $\lambda_p^m, \sigma_p^m, \omega_p^m, \boldsymbol{\theta}_p^m, \boldsymbol{\xi}_p^m$ **by**
    $\lambda_i^m = \lambda_i^m + \mu(e_i^m - (y_i^m - q_i^m + r_i^m))$;
    $\sigma_i^m = \sigma_i^m + \mu(q_i^m - \max(\mathbf{t}_i^m))$;
    $\omega_i^m = \omega_i^m + \mu(r_i^m - \max(\mathbf{u}_i^m))$;
    $\boldsymbol{\theta}_i^m = \boldsymbol{\theta}_i^m + \mu(\mathbf{t}_i^m - (\mathbf{w}_m^T\mathbf{X}_i + \mathbf{1}b_m))$;
    $\boldsymbol{\xi}_i^m = \boldsymbol{\xi}_i^m + \mu(\mathbf{u}_i^m - (\mathbf{w}_y^T\mathbf{X}_i + \mathbf{1}b_y))$.
  end for
  Update $\mu = \rho\mu$.
end while

return $(\mathbf{w}_m, \ldots, \mathbf{w}_K) \in \mathbf{W}$ and $(b_1, \ldots, b_K) \in \mathbf{b}$.

E update. Dropping terms from Eq. (4), that do not contain $\mathbf{E}$ and decoupling element-wise gives $K \times N$ problems

$$e_i^m = \underset{e_i^m}{\arg\min}\; C\left(y_i^m e_i^m\right)_+ + \frac{\mu}{2}\left(e_i^m - n_i^m\right)^2 \quad , \tag{13}$$

where $n_i^m = y_i^m - q_i^m + r_i^m - \frac{\lambda_i^m}{\mu}$. Equation (13) can be differentiated with respect to e_i^m via the sub-gradient method, and solved in three cases

$$e_i^m = \begin{cases} n_i^m - \frac{C}{\mu} y_i^m & \text{when} \quad y_i^m n_i^m > \frac{C}{\mu} \;, \\ 0 & \text{when} \quad 0 \le y_i^m n_i^m \le \frac{C}{\mu} \;, \\ n_i^m & \text{when} \quad y_i^m n_i^m < 0 \;, \end{cases} \quad . \tag{14}$$

Q & R update. Keeping only terms with $\mathbf{Q}$ in Eq. (4) and decoupling element-wise gives $K \times N$ problems

$$\begin{aligned} q_i^m = \underset{q_i^m}{\arg\min}\; & \left(e_i^m - y_i^m + q_i^m - r_i^m + \lambda_i^m/\mu\right)^2 \\ & + \left(q_i^m - \max\left(\mathbf{t}_i^m\right) + \sigma_i^m/\mu\right)^2 \quad . \end{aligned} \tag{15}$$

Taking the derivative of Eq. (15) with respect to q_i^m, setting the result equal to zero, and solving for q_i^m gives the update

$$q_i^m = \frac{\left(y_i^m - e_i^m + r_i^m - \lambda_i^m/\mu + \max\left(\mathbf{t}_i^m\right) - \sigma_i^m/\mu\right)}{2} \quad . \tag{16}$$

Following the same steps for each $r_i^m \in \mathbf{R}$ we derive the element-wise updates

$$r_i^m = \frac{\left(e_i^m - y_i^m + q_i^m + \lambda_i^m/\mu + \max\left(\mathbf{u}_i^m\right) - \omega_i^m/\mu\right)}{2} \quad . \tag{17}$$

T & U update. Keeping terms in Eq. (4) containing $\mathbf{T}$ and decoupling across K and N gives the following

$$\begin{aligned} \mathbf{t}_i^m = \underset{\mathbf{t}_i^m}{\arg\min}\; & \left(q_i^m - \max\left(\mathbf{t}_i^m\right) + \sigma_i^m/\mu\right)^2 \\ & + \left\|\mathbf{t}_i^m - \left(\mathbf{w}_m^T \mathbf{X}_i + \mathbf{1} b_m\right) + \theta_i^m/\mu\right\|_2^2 \quad , \end{aligned} \tag{18}$$

which can be further decoupled element-wise for each $t_{i,j}^m \in \mathbf{t}_i^m$ giving $K \times N \times (n_1 + \cdots + n_N)$ problems

$$t_{i,j}^m = \underset{t_{i,j}^m}{\arg\min} \begin{cases} \left(q_i^m - t_{i,j}^m + \sigma_i^m/\mu\right)^2 + \left(t_{i,j}^m - \phi_{i,j}^m\right)^2 \\ \text{when } t_{i,j}^m = \max\left(\mathbf{t}_i^m\right), \\ \left(t_{i,j}^m - \phi_{i,j}^m\right)^2 \;\; \text{else} \;, \end{cases} \tag{19}$$

where $\boldsymbol{\phi}_i^m = \mathbf{w}_m^T \mathbf{X}_i + \mathbf{1} b_m - \boldsymbol{\theta}_i^m/\mu$. Taking the derivative of Eq. (19) with respect to $t_{i,j}^m$, setting the result equal to zero, and solving for $t_{i,j}^m$, gives the updates

$$t_{i,j}^m = \begin{cases} \frac{\max(\boldsymbol{\phi}_i^m) + q_i^m + \sigma_i^m/\mu}{2} & \text{if } j = \arg\max(\boldsymbol{\phi}_i^m) \;, \\ \phi_{i,j}^m & \text{else} \;, \end{cases} \quad . \tag{20}$$

This same strategy is applied to derive the element-wise updates of $\mathbf{U}$, giving

$$u_{i,j}^m = \begin{cases} \frac{\max(\boldsymbol{\psi}_i^m) + r_i^m + \omega_i^m/\mu}{2} & \text{if } j = \arg\max(\boldsymbol{\psi}_i^m) \ , \\ \psi_{i,j}^m & \text{else} \ , \end{cases} \ . \tag{21}$$

where $\boldsymbol{\psi}_i^m = \mathbf{w}_y^T \mathbf{X}_i + \mathbf{1} b_y - \boldsymbol{\xi}_i^m/\mu$. The associated dual variable updates are provided in Algorithm 1.

E Avoiding a Least Squares Calculation

As we can see in Eq. (6) the update for $\mathbf{w}_m$ is reliant on solving a least squares problem in every iteration. However, the least squares solver has complexity $O(Nd^2)$ and will have to be solved every iteration which may not be computationally feasible if the number of features d is very large. To avoid this problem we can instead utilize an optimal line search method [15] and update $\mathbf{w}_m$ via gradient descent:

$$\mathbf{w}_m = \mathbf{w}_m - s_m \nabla_{\mathbf{w}_m}, \tag{22}$$

where $\nabla_{\mathbf{w}_m}$ is the analytical gradient of Eq. (4) with respect to $\mathbf{w}_m$:

$$\begin{aligned} \nabla_{\mathbf{w}_m} = \mathbf{w}_m &- \mu \mathbf{X}_i \sum_{i=1}^{N} [\mathbf{t}_i^m - \mathbf{w}_m^T \mathbf{X}_i - \mathbf{1} b_m + \boldsymbol{\theta}_i^m/\mu]^T \\ &- \mu \mathbf{X}_{i'} \sum_{i'=1}^{N'} \sum_{m'=1}^{K} [\mathbf{u}_{i'}^{m'} - \mathbf{w}_m^T \mathbf{X}_{i'} - \mathbf{1} b_m + \boldsymbol{\xi}_{i'}^{m'}/\mu]^T \ , \end{aligned} \tag{23}$$

and it can be used to define a minimization:

$$\begin{aligned} s_m^* = \arg\min_{s_m} \ & \frac{1}{2} \left\| \mathbf{w}_m^T - s_m \nabla_{\mathbf{w}_m}^T \right\|_2^2 \\ + \frac{\mu}{2} & \sum_{i=1}^{N} \left[\left\| \mathbf{t}_i^m - (\mathbf{w}_m^T - s_m \nabla_{\mathbf{w}_m}^T) \mathbf{X}_i - \mathbf{1} b_m + \theta_i^m/\mu \right\|_2^2 \right] \\ + \sum_{i'=1}^{N'} & \sum_{m=1}^{K} \left[\frac{\mu}{2} \left\| \mathbf{u}_{i'}^m - (\mathbf{w}_m^T - s_m \nabla_{\mathbf{w}_m}^T) \mathbf{X}_{i'} - \mathbf{1} b_m + \xi_{i'}^m/\mu \right\|_2^2 \right] \ , \end{aligned} \tag{24}$$

in terms of s_m instead of $\mathbf{w}_m$. Differentiating Eq. (24) with respect to s_m, setting the result equal to zero gives:

$$s_m^* = \frac{\left(\mathbf{w}_m^T - \mu \sum_{i=1}^{N} \hat{\mathbf{t}}_i^m \mathbf{X}_i^T - \mu \sum_{i'=1}^{N'} \sum_{m'}^{K} \hat{\mathbf{u}}_{i'}^{m'} \mathbf{X}_{i'}^T \right) \nabla_{\mathbf{w}_m}}{\nabla_{\mathbf{w}_m}^T \left(\mathbf{I} + \mu \sum_{i=1}^{N} \mathbf{X}_i \mathbf{X}_i^T + \mu K \sum_{i'=1}^{N'} \mathbf{X}_{i'} \mathbf{X}_{i'}^T \right) \nabla_{\mathbf{w}_m}} \tag{25}$$

where $\hat{\mathbf{t}}_i^m = \mathbf{t}_i^m - \mathbf{w}_m^T \mathbf{X}_i - \mathbf{1} b_m + \boldsymbol{\theta}_i^m/\mu$ and $\hat{\mathbf{u}}_{i'}^{m'} = \mathbf{u}_{i'}^{m'} - \mathbf{w}_m^T \mathbf{X}_{i'} - \mathbf{1} b_m + \xi_{i'}^{m'}/\mu$. Finally we plug Eq. (23) and Eq. (25) into Eq. (22) to earn a efficient update

equation which avoids the least squares problem in Eq. (6). The time complexity of the proposed method is $O(Nd\bar{n})$, where $\bar{n}$ is the average number of instances per bag. The number of instances $\bar{n}$ is typically smaller than the number of features d (the multiple of 162 in our experiments), therefore our model with the solution in Eq. (22) (inexact *pdMISVM*) is more scalable compared to Eq. (6) (exact *pdMISVM*).