Supplementary Material for "Progressive Label Propagation for Semi-Supervised Multi-Dimensional Classification"

Teng Huang, Bin-Bin Jia, Min-Ling Zhang

To facilitate understanding, Table 1 summarizes the notations used in Section 3 and Algorithm 1 further presents the pseudo code of the proposed PLAP approach.

Table 1: Summary of the notations used in our paper.

Notation	Descriptions
d	number of features in input space
q	number of class spaces (dimensions) in output space
K_j	number of class labels in the j-th class space $(1 \leq j \leq q)$
L	number of labeled MDC samples
U	number of unlabeled MDC samples
N	number of MDC samples, i.e., $N = L + U$
$\mathcal{X}$	the d-dimensional input (feature) space, i.e., $\mathcal{X} = \mathbb{R}^d$
C_j	the j-th class space where $C_j = \{c_1^j, c_2^j, \ldots, c_{K_j}^j\}$ $(1 \leq j \leq q)$
c_a^j	the a-th class label in C_j $(1 \leq a \leq K_j)$
$\mathcal{Y}$	the output space where $\mathcal{Y} = C_1 \times C_2 \times \ldots \times C_q$
$\mathcal{D}_l$	the set of labeled MDC samples where $\mathcal{D}_l = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid 1 \leq i \leq L\}$
$\mathcal{D}_u$	the set of unlabeled MDC samples where $\mathcal{D}_u = \{\boldsymbol{x}_i \mid L+1 \leq j \leq L+U\}$
$\mathcal{D}$	the set of MDC samples where $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$
$\boldsymbol{x}_i$	the i-th feature vector where $\boldsymbol{x}_i = [x_{i1}, x_{i2}, \ldots, x_{id}]^\top \in \mathcal{X}$
$\boldsymbol{y}_i$	the i-th class vector where $\boldsymbol{y}_i = [y_{i1}, y_{i2}, \ldots, y_{iq}]^\top \in \mathcal{Y}$
f	the SSMDC predictive model: $\mathcal{X} \mapsto \mathcal{Y}$ from $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$
$\mathbf{W}$	the affinity matrix defined based on the Gaussian function, i.e., $W_{ij} = e^{-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}}$ if $i \neq j$ and $W_{ii} = 0$
σ	the bandwidth parameter of Gaussian function
$\mathbf{S}$	the propagation matrix where $\mathbf{S} = \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$
$\mathbf{D}$	the diagonal matrix where $\mathbf{D} = diag(d_1, d_2, \ldots, d_N)$ with $d_i = \sum_{j=1}^N W_{ij}$
α	the hyper-parameter to be specified for balancing the importance of $\mathbf{SF}(t)$ and $\mathbf{Y}$ In Eq.(2)
$\mathbf{Y}$	the initial label matrix where Y_{ia} denotes the (i,a)-th item of $\mathbf{Y}$
$\mathbf{F}(t)$	the propagated label matrix in the t-th round $(t \in \{1, 2, \ldots, \})$ and $\mathbf{F}(0) = \mathbf{Y}$
$\phi_j(\cdot, \cdot, \ldots, \cdot)$	the injective function from the Cartesian product $\{1, 2, \ldots, K_1\} \times \{1, 2, \ldots, K_2\} \times \ldots \times \{1, 2, \ldots, K_j\}$ to natural numbers $\{1, 2, \ldots, K_1 \times K_2 \times \ldots \times K_j\}$
$\phi_j^{-1}(\cdot)$	the inverse function of $\phi_j(\cdot, \cdot, \ldots, \cdot)$
$\mathcal{N}(\boldsymbol{x}_i)$	the k nearest neighbors of $\boldsymbol{x}_i$ identified in labeled samples $\mathcal{D}_l$
n_{ia}^j	the number of samples with class label c_a^j w.r.t. C_j in $\mathcal{N}(\boldsymbol{x}_i)$
CL_i^j	the condition for labeled sample $\boldsymbol{x}_i$ to initialize $\mathbf{Y}$ where $\mathrm{CL}_i^j \triangleq (y_{i1} = c_{a_1}^1) \wedge \ldots \wedge (y_{ij} = c_{a_j}^j)$
CU_i^j	the condition for unlabeled sample $\boldsymbol{x}_i$ to initialize $\mathbf{Y}$ where $\mathrm{CU}_i^j \triangleq (a_1 = \hat{a}_1) \wedge \ldots \wedge (a_{j-1} = \hat{a}_{j-1})$

Algorithm 1 The PLAP approach

Input: The labeled MDC samples $\mathcal{D}_l$, unlabeled samples $\mathcal{D}_u$, the number of nearest neighbors k
Output: Predicted class vector $\hat{\boldsymbol{y}}_i$ for unlabeled sample $\boldsymbol{x}_i \in \mathcal{D}_u$

```
1: Construct the affinity matrix W according to Eq.(1) and then calculate the propagation matrix S based on W;
2: Set the value of label matrix Y according to Eq.(3);
3: Iterate Eq.(2) with calculated S and Y until convergence;
4: Obtain the predicted class label ŷ_i1 w.r.t. C_1 for unlabeled sample x_i ∈ D_u according to Eq.(4);
5: Identify the k nearest neighbors of unlabeled sample x_i ∈ D_u from labeled samples D_l and store them in N(x_i);
6: while 2 ≤ j ≤ q do
7:     Update Y according to Eq.(8) and Eq.(9);
8:     Iterate Eq.(2) with S and updated Y until convergence;
9:     Obtain the predicted class labels ŷ_i1, ..., ŷ_ij w.r.t. C_1, ..., C_j for unlabeled sample x_i ∈ D_u according to the
       generalized version for the j-th class sapce of Eq.(7);
10:    j ← j + 1;
11: end while
12: return Predicted results {ŷ_i | L + 1 ≤ i ≤ L + U}
```