Machine Learning (2020) 109:445–465
https://doi.org/10.1007/s10994-019-05844-9

Joint consensus and diversity for multi-view semi-supervised classification

Wenzhang Zhuge[1] · Chenping Hou[1] · Shaoliang Peng[2] · Dongyun Yi[1]

Received: 9 April 2019 / Revised: 2 July 2019 / Accepted: 16 September 2019 / Published online: 7 October 2019

Abstract

As data can be acquired in an ever-increasing number of ways, multi-view data is becoming more and more available. Considering the high price of labeling data in many machine learning applications, we focus on multi-view semi-supervised classification problem. To address this problem, in this paper, we propose a method called joint consensus and diversity for multi-view semi-supervised classification, which learns a common label matrix for all training samples and view-specific classifiers simultaneously. A novel classification loss named probabilistic square hinge loss is proposed, which avoids the incorrect penalization problem and characterizes the contribution of training samples according to its uncertainty. Power mean is introduced to incorporate the losses of different views, which contains the auto-weighted strategy as a special case and distinguishes the importance of various views. To solve the non-convex minimization problem, we prove that its solution can be obtained from another problem with introduced variables. And an efficient algorithm with proved convergence is developed for optimization. Extensive experimental results on nine datasets demonstrate the effectiveness of the proposed algorithm.

Keywords Consensus · Diversity · Multi-view · Semi-supervised classification

Editors: Kee-Eung Kim and Jun Zhu.

✉ Chenping Hou
hcpnudt@hotmail.com

✉ Shaoliang Peng
slpeng@hnu.edu.cn

Wenzhang Zhuge
zgwznudt@yeah.net

Dongyun Yi
dongyun.yi@gmail.com

[1] College of Liberal Arts and Science, National University of Defense Technology, Changsha, China

[2] College of Computer Science and Electronic Engineering and National Supercomputing Centre in Changsha, Hunan University, Changsha, China

1 Introduction

With the advent of vast data collection ways, in many real applications of machine learning, pattern recognition, computer vision and data mining, data are easier to have heterogeneous features representing samples from diverse information channels or different feature extractors. For example, in web data, a web page can be represented by its content and link information; in visual data, each image could be described by different descriptors, such as GIST (Oliva and Torralba 2001), HOG (Dalal and Triggs 2005) and SIFT (Lowe 2004). This kind of data is called multi-view data and each representation is referred to a view (Xu et al. 2013). In general, each representation captures specific characteristics of the studied object, therefore, different views have complementary and partly independent information to one another. On the other hand, since these representations describe the same object, there should be consensus information among views. In recent years, how to better manipulate multi-view data has aroused considerable research interests.

In many real applications, although data collection ways become various, labeling data is still a time consuming and biased task. Therefore, the collected data usually have multiple representations but scarce labels. For example, in image classification, extensive images are accessible from the internet and different descriptors are applied to extract features. However, obtaining labeled data is expensive because it requires efforts of human annotators who should often be quite skilled. The above mentioned two characters: multiple views and abundant unlabeled samples, suggest the multi-view semi-supervised learning (MVSSL) strategy. Many researches (Cai et al. 2013; Chen et al. 2012; Gong 2017; Guz and Tur 2009; Hou et al. 2010; Nie et al. 2018; Yu et al. 2012) have shown that using multiple representations and abundant unlabeled data jointly will boost performance. In this paper, we focus on the classification task.

Existing multi-view semi-supervised classification methods can be roughly categorized into three groups. The first group is known as co-training (Blum and Mitchell 1998), which is originally designed for two-view data. It firstly trains classifiers with the labeled data and classifies the unlabeled data on each view independently. Next the most confidently predicted samples of each classifier are added to the other classifier's training set, then the procedure repeats. Based on the thought of co-training, many algorithms (Mao et al. 2009; Nigam and Ghani 2000; Sun and Jin 2011) have been proposed. The second group is graph-based methods, which treats labeled and unlabeled instances as vertices of a common graph and uses edges to propagate the label information (Gong et al. 2016). Methods (Cai et al. 2013; Karasuyama and Mamitsuka 2013; Nie et al. 2016; Gong et al. 2016, 2017) firstly construct graphs on each individually, then learn view weights to combine a common graph and performs label propagation simultaneously. Nie et al. (2018) uses a parameter-free way to learn a common graph matrix, a common label indicator matrix and view weights simultaneously. The third group is regression-based methods (Tao et al. 2017; Yang et al. 2013), which learn view-specific projection matrices to exploit the diversity information and employ the label matrix as the common regression target across views to enhance consensus. Based on the projection matrices, out-of-sample data can be efficiently dealt with.

Compared with applying single-view semi-supervised methods on each view or the simply concatenated view, the aforementioned multi-view algorithms can achieve better performance in most cases. That is because these multi-view methods learn view-specific predictors to explore the diversity information, and enforce the predictions consensus, which maximizes the agreement among different views and exploits the consensus information. However, their performance can be further improved due to the following reasons. Co-training based methods require classification on each view to be accurate. This kind of methods ignore the diversity information of views and treat them equally. Their performance may suffer when there exists a difficult-to-classify view because the erroneous information will be provided to other classifiers. Graph-based methods have three main limitations. First, as transductive approaches, these methods have low-efficiency to classify out-of-sample data, since they need to rerun the algorithms. Second, due to the computational burdens of the graph construction and the label propagation, these methods can not be utilized on datasets with large data size. Last but not least, their performance may be deteriorated when two classes overlap significantly (Xu and King 2014). Regression-based methods employ the regression loss as the classification loss, which usually incorrectly penalize the right classification. Besides, they distinguish the importance of training samples by manually assigning small weights for unlabeled samples, which lacks a more reasonable learning mechanism.

In this paper, we propose a new method, named as joint consensus and diversity for multi-view semi-supervised classification (JCD). To facilitate consensus, JCD learns a common probability label matrix, which makes the classification consistent across views. To enhance diversity, JCD learns view-specific classifiers, proposes probabilistic square hinge loss as the classification loss, and incorporates the losses of multiple views by power mean. With the learned linear classifiers, predictions for out-of-sample data can be easily made. And the proposed classification loss fixes the incorrect penalization problem, and characterizes the contribution importance of different training samples according to the degree of classification uncertainty. Moreover, the power mean strategy distinguishes the importance of views according to their losses. Hence, the impacts of boundary unlabeled data points and low-quality views can be weaken. An efficient algorithm is developed to solve the non-convex problem. We summarize the contributions of this paper as follows.

- With the proposed probabilistic square hinge loss, the incorrect penalization problem of previous regression-based losses (Luo et al. 2017; Wang et al. 2014) has been overcame, which enables different classifiers to obey the consensus principle and the diversity principle simultaneously. And the importance diversity of different training samples is taken into consideration.
- With the power mean incorporation strategy, the proposed JCD is robust against the low-quality views. And we show that the auto-weighted strategy (Huang et al. 2019; Nie et al. 2016; Shu et al. 2017; Nie et al. 2018; Zhuge et al. 2017) is a special case of power mean strategy.
- We prove that the solution can be obtained by solving another problem with introduced variables and develop an efficient algorithm for optimization, which can be applied to large-scale multi-view semi-supervised classification. We also prove that the algorithm monotonically decreases the objective of the model until it converges to a stationary point.
- We verify the effectiveness of the proposed algorithm on nine real-world multi-view datasets. The experimental results indicate that JCD achieves better classification results than other compared methods.

2 Notations and related works

In this paper, matrices and vectors are written as boldface uppercase letters and boldface lowercase letters respectively. For a matrix $\mathbf{M}$, the ith row, jth column and (i, j)th element are denoted by $\mathbf{m}_i$, $\mathbf{m}_{:j}$ and m_{ij}, respectively. $Tr(\cdot)$ denotes the trace operation of a matrix and $||\cdot||_F$ is the matrix Frobenius norm. The ℓ_2 norm of a vector $\mathbf{m} \in \mathbb{R}^d$ is denoted by $||\mathbf{m}||_2 = (\sum_{i=1}^d |m_i|^2)^{\frac{1}{2}}$. $\mathbf{1}_q \in \mathbb{R}^q$ denotes a q-dimensional vector of all ones. The signal function is denoted by $\text{sgn}(\cdot)$. If $x \geq 0$, $\text{sgn}(x) = 1$; otherwise, $\text{sgn}(x) = -1$. The power mean of a set $\{x_i\}_n$ with order p is denoted as

$$\mathcal{M}_p(\{x_i\}_n) = \sqrt[p]{\frac{1}{n}\sum_{i=1}^n x_i^p} \tag{1}$$

Given n samples $\{\mathbf{x}_i\}_n$, the data matrix is denoted by $\mathbf{X} = [\mathbf{x}_1; \ldots; \mathbf{x}_n] \in \mathbb{R}^{n\times d}$. The ith sample $\mathbf{x}_i = [\mathbf{x}_i^{(1)}, \ldots, \mathbf{x}_i^{(V)}] \in \mathbb{R}^{1\times d}$ has features from V views, and the vth $\mathbf{x}_i^{(v)} \in \mathbb{R}^{1\times d^{(v)}}$ has $d^{(v)}$ features so that $d = \sum_{v=1}^V d^{(v)}$. $\mathbf{X}^{(v)} = [\mathbf{x}_1^{(v)}; \ldots; \mathbf{x}_n^{(v)}]$ denotes the data matrix on the vth view, thus $\mathbf{X} = [\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(V)}]$. Supposing that n data samples belong to C classes, the first l instances are already labeled and the rest $u = n - l$ samples $(l \ll u)$ are unlabeled. Denote $\mathbf{Y}_l = [\mathbf{y}_1; \ldots; \mathbf{y}_l]$ and $\mathbf{Y}_u = [\mathbf{y}_{l+1}; \ldots; \mathbf{y}_n]$ as the label matrices of l labeled samples and u labeled samples, respectively, where $\mathbf{y}_i \in \{0, 1\}^{1\times C}$ is a 1-of-C binary label vector for the ith sample $\mathbf{x}_i$. Therefore, the label matrix for all samples can be denoted as $\mathbf{Y} = [\mathbf{Y}_l; \mathbf{Y}_u] \in \{0, 1\}^{n\times C}$. To identify the C classes uniquely, the cth class is assigned with a coding $\mathbf{t}_{(c)} \in \{-1, 1\}^{1\times C}$, where only the cth element of $\mathbf{t}_{(c)}$ is 1 and the others are -1.

2.1 Multi-view learning with adaptive neighbors

The multi-view learning with adaptive neighbors (MLAN) is a graph-based semi-supervised method (Nie et al. 2018). Based on the view representations $\{\mathbf{X}^{(v)}\}_V$ and the given binary label matrix $\mathbf{Y}_l$, MLAN learns a graph matrix $\mathbf{S} \in \mathbb{R}^{n\times n}$ and a class indicator matrix $\mathbf{F} = [\mathbf{F}_l; \mathbf{F}_u] \in \mathbb{R}^{n\times C}$ across views simultaneously. The objective function of MLAN is

$$\begin{aligned} &\min_{\mathbf{S},\mathbf{F}} \sum_{v=1}^V \sqrt{\sum_{i,j} s_{ij}||\mathbf{x}_i^{(v)} - \mathbf{x}_j^{(v)}||_2^2} + \gamma||\mathbf{S}||_F^2 + \lambda Tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) \\ &s.t.\ \mathbf{F}_l = \mathbf{Y}_l, \quad \sum_{j=1}^n s_{ij} = 1, \quad s_{ij} \geq 0, (\forall i) \end{aligned} \tag{2}$$

where $\gamma > 0$ is used to adjust the distribution of each $\mathbf{s}_i$, $\lambda > 0$ is a balanced parameter and $\mathbf{L}$ is the Laplacian matrix of $\mathbf{S}$.

2.2 Multi-view semi-supervised classification via adaptive regression

The multi-view semi-supervised classification via adaptive regression (MVAR) is a regression-based semi-supervised algorithm (Tao et al. 2017). For each representations $\mathbf{X}^{(v)} \in \mathbb{R}^{n\times d^{(v)}}$, MVAR learns a corresponding projection matrix $\mathbf{W}^{(v)} \in \mathbb{R}^{d^{(v)}\times C}$ and a bias vector $\mathbf{b}^{(v)} \in \mathbb{R}^{1\times C}$ as the vth view classifier. To enforce view-specific predictor consensus, MVAR learns a shared binary label matrix $\mathbf{F} \in \{0, 1\}^{n\times C}$ as the common regression targets of different views. To be specific, the objective function of MVAR is

$$\begin{aligned} &\min_{\mathbf{W}^{(v)},\mathbf{b}^{(v)},\mathbf{F},\boldsymbol{\alpha}} \sum_{v=1}^{V} (\alpha^{(v)})^{\gamma} \left(u_i ||\mathbf{x}_i^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)} - \mathbf{f}_i||_2 + \lambda^{(v)}||\mathbf{W}^{(v)}||_F^2 \right) \\ &s.t.\ \mathbf{F}_l = \mathbf{Y}_l, \quad \sum_{v=1}^{V} \alpha^{(v)} = 1, \quad \alpha^{(v)} \geq 0 \end{aligned} \tag{3}$$

where $\{\alpha^{(v)}\}_V$ are the learnable view weight factors for each view, $\gamma > 1$ is to control the distribution of view weights, $\lambda^{(v)} > 0$ is the regularization parameter of the vth view, and $u_i > 0$ is the instance weight parameter of the ith sample. For labeled samples $\{\mathbf{x}_i\}_l$ and unlabeled samples $\{\mathbf{x}_i\}_{i=l+1}^{n}$, $\{u_i\}_n$ are manually assigned with different values to distinguish their importance.

2.3 Semi-supervised learning with discriminative least squares regression

The adaptive semi-supervised learning with discriminative least squares regression (ASL-DLSR) is a single-view linear regression model (Luo et al. 2017) designed for semi-supervised classification. Following (Wang et al. 2014), ASL-DLSR learns a transformation matrix $\mathbf{W} \in \mathbb{R}^{d\times C}$, a bias vector $\mathbf{b} \in \mathbb{R}^{1\times C}$ and a probability label matrix $\mathbf{F} \in \mathbb{R}^{n\times C}$ simultaneously. Different from Wang et al. (2014) which employs $\{\mathbf{t}_{(c)}\}_C$ as regression targets, ASL-DLSR introduces a adjustment vector $\mathbf{m}_{(c)} \in \mathbb{R}_{+}^{1\times C}$ for each class and employs $\{\mathbf{t}_{(c)} + \mathbf{m}_{(c)} \odot \mathbf{t}_{(c)}\}_C$ as regression targets, where $\odot$ is the hadamard product. By introducing $\{\mathbf{m}_{(c)}\}_C$, ASL-DLSR alleviates the incorrect penalization problem in Wang et al. (2014). The objective function of ASL-DLSR can be written as

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{b},\mathbf{F},\{\mathbf{m}_{(c)}\}_C} \sum_{i=1}^{n}\sum_{c=1}^{C} f_{ic}^{\gamma} ||\mathbf{x}_i\mathbf{W} + \mathbf{b} - \mathbf{m}_{(c)} \odot \mathbf{t}_{(c)} - \mathbf{t}_{(c)}||^2 + \lambda ||\mathbf{W}||_F^2 \\ &s.t.\ \mathbf{F}_l = \mathbf{Y}_l, \quad \sum_{c=1}^{C} f_{ic} = 1, \quad f_{ic} \geq 0, \quad \mathbf{m}_{(c)} \geq 0 \end{aligned} \tag{4}$$

where $\gamma \geq 1$ and $\lambda > 0$ are two hyper-parameters. Similar to Wang et al. (2014), $\sum_{c=1}^{C} f_{ic}^{\gamma}$ is regarded as the weight of the ith sample, which measure the importance of the ith sample according to its classification certainty.

We present the following example to show how introduced $\{\mathbf{m}_{(c)}\}_C$ alleviate the incorrect penalization problem. Suppose that a data set can be classified into 3 classes, and two data points $\mathbf{x}_i$ and $\mathbf{x}_i$ belong to the first class. If the predictions of $\mathbf{x}_i$ and $\mathbf{x}_j$ are $[2, -1, -1]$ and $[6, -1, -1]$, respectively, considering the first class indicator vector $\mathbf{t}_{(1)} = [1, -1, -1]$, they are both classified correctly. However, by calculating the regression losses to $\mathbf{t}_{(1)}$, the classification losses of $\mathbf{x}_i$ and $\mathbf{x}_j$ in Wang et al. (2014) are 1 and 25, respectively. By optimizing $\mathbf{m}_{(1)}$ and setting $\mathbf{m}_{(1)} = [3, 0, 0]$, the classification losses of $\mathbf{x}_i$ and $\mathbf{x}_j$ in ASL-DLSR become 4 and 4. Compared with Wang et al. (2014), ASL-DLSR reduces the sum of incorrect penalization of $\mathbf{x}_i$ and $\mathbf{x}_j$.

3 The proposed methodology

In this section, we propose the formulation of our model: joint consensus and diversity for multi-view semi-supervised classification (JCD). We first formulate the objective function for each single view and then integrate them to the multi-view scenario.

Based on the vth view data matrix $\mathbf{X}^{(v)}$ and the label matrix $\mathbf{Y}_l$ for labeled samples, we aim to train a classifier $f^{(v)}$ and learn a label matrix $\mathbf{F} = [\mathbf{f}_1; \ldots; \mathbf{f}_n] \in \mathbb{R}^{n\times C}$ for all samples simultaneously, where $\mathbf{f}_i \in \mathbb{R}^{1\times C}$ is the label vector of the ith sample $\mathbf{x}_i^{(v)}$. To fulfill this goal, the general objective function can be formulated as

$$\min_{f^{(v)},\mathbf{F},\mathbf{F}_l=\mathbf{Y}_l} \sum_{i=1}^{n} \ell\big(f^{(v)}(\mathbf{x}_i^{(v)}), \mathbf{f}_i\big) + \lambda\Omega\big(f^{(v)}\big) \tag{5}$$

where $\mathbf{F}_l \in \mathbb{R}^{l\times C}$ represents the first l rows of $\mathbf{F} = [\mathbf{F}_l; \mathbf{F}_u]$, $f^{(v)}(\mathbf{x}_i^{(v)}) \in \mathbb{R}^{1\times C}$ is the prediction of $\mathbf{x}_i^{(v)}$, $\ell\big(\cdot, \cdot\big)$ is the classification loss function, $\lambda > 0$ is a trade-off parameter, and $\Omega\big(\cdot\big)$ is the regularization term. By combining different classifiers, loss functions and regularization terms, the vth view semi-supervised classification can be implemented in a variety of ways.

In this paper, the predictions of $\mathbf{x}_i^{(v)}$ shall be parameterized as $f^{(v)}(\mathbf{x}_i^{(v)}) = \mathbf{x}_i^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)}$, where $\mathbf{W}^{(v)} \in \mathbb{R}^{d^{(v)}\times C}$ is the projection matrix and $\mathbf{b}^{(v)} \in \mathbb{R}^{1\times C}$ is the bias vector. Although we adopt a linear model here, our results can be extended for non-linear kernels as well. If $\mathbf{x}_i^{(v)}$ belongs to the cth class, the square hinge loss can be calculated as

$$H_{ic}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}; \mathbf{x}_i^{(v)}) = \sum_{j=1}^{C} \Big(1 - t_{(c)j}\big(\mathbf{x}_i^{(v)}\mathbf{w}_{:j}^{(v)} + b_j^{(v)}\big)\Big)_+^2 \tag{6}$$

where $t_{(c)j}$ and $b_j^{(v)}$ are the jth element of $\mathbf{t}_{(c)}$ and $\mathbf{b}^{(v)}$, $\mathbf{w}_{:j}^{(v)}$ is the jth column of $\mathbf{W}^{(v)}$, and the function $(a)_+$ is defined as $(a)_+ = \max(0, a)$.

Based on (6), we propose a novel probabilistic square hinge loss to characterize the contribution importance of varying training samples, i.e.

$$\ell\big(f^{(v)}(\mathbf{x}_i^{(v)}), \mathbf{f}_i\big) = \sum_{c=1}^{C} f_{ic}^{\gamma} H_{ic}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}; \mathbf{x}_i^{(v)}) \tag{7}$$

where $\mathbf{f}_i = [f_{i1}, \ldots, f_{iC}] \in [0, 1]^{1\times C}$ is the probability label vector for the ith sample, f_{ic} refers to the probability of ith instance belonging to the cth class, and $\gamma \geqslant 1$ is an adaptive parameter. The advantages of (7) are embodied in the following two aspects. (1)Similar to Luo et al. (2017), Wang et al. (2014), $\sum_{c=1}^{C} f_{ic}^{\gamma}$ can be regarded as the weight of the ith sample. Due to the constraint $\mathbf{F}_l = \mathbf{Y}_l$, the weights of labeled samples are always 1, which ensures the significance of them. When $\gamma > 1$, the weights of unlabeled samples are determined by the certainty degree of classification, which makes the more clearly classified unlabeled samples play more important roles on the training stage. (2)Taking the advantage of hinge loss, our proposed probabilistic fitting loss overcomes the incorrect penalization problem of previous regression-based losses (Luo et al. 2017; Wang et al. 2014). Considering the example introduced in Sect. 2.3, if (7) is used to calculated the classification losses of the two right classification points $\mathbf{x}_i$ and $\mathbf{x}_j$, their classification losses are both 0, which avoids the incorrect penalization.

To control the complexity of each single-view model, we adopt $\Omega\big(f^{(v)}\big) = ||\mathbf{W}^{(v)}||_F^2$ as the regularization term. Then, we obtain the objective functions of each view, and the vth one is denoted as $\mathcal{L}^{(v)}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F})$ $(v = 1, \ldots, V)$. After proposing the objective function of each view, we integrate them for multi-view data. A rough way to obtain the multi-view formulation is to add them up directly. However, this way neglects the different importance of views. To distinguish the importance of varying views, we adopt the power mean strategy and propose our JCD as the following form:

$$\begin{aligned}
&\min_{\{\mathbf{W}^{(v)},\mathbf{b}^{(v)}\}_V,\mathbf{F}} \mathcal{M}_p(\{\mathcal{L}^{(v)}(\mathbf{W}^{(v)},\mathbf{b}^{(v)},\mathbf{F})\}_V) \\
&= \sqrt[p]{\frac{1}{V}\sum_{v=1}^{V}\mathcal{L}^{(v)}(\mathbf{W}^{(v)},\mathbf{b}^{(v)},\mathbf{F})^p} \\
&= \sqrt[p]{\frac{1}{V}\sum_{v=1}^{V}\Big(\sum_{i=1}^{n}\sum_{c=1}^{C} f_{ic}^{\gamma}\sum_{j=1}^{C}\big(1 - t_{(c)j}(\mathbf{x}_i^{(v)}\mathbf{w}_{:j}^{(v)} + b_j^{(v)})\big)_+^2 + \lambda||\mathbf{W}^{(v)}||_F^2\Big)^p} \qquad (8)\\
&s.t.\ \mathbf{F}_l = \mathbf{Y}_l, \quad f_{ic} \geq 0, \quad \sum_{c=1}^{C} f_{ic} = 1\ (\forall i, c)
\end{aligned}$$

where p is a parameter and it satisfies $p < 1$ and $p \neq 0$. The power mean strategy distinguishes the importance of various views according to the view loss, which enables the views with smaller losses to play more important roles in classification. The auto-weighted strategy has been widely adopted by recent works (Huang et al. 2019; Nie et al. 2016; Shu et al. 2017; Nie et al. 2018; Zhuge et al. 2017) to incorporate the losses of different views, which essentially is a special case of power mean strategy with $p = \frac{1}{2}$.

4 Optimization procedure

The problem (8) is non-convex and difficult to solve directly. Different from Huang et al. (2019), Nie et al. (2016), Shu et al. (2017), Zhuge et al. (2017) which use re-weighted method Nie et al. (2017) to deal with the auto-weighted integration strategy, we will prove that the solution of (8) can be obtained by solving the following problem

$$\begin{aligned}
&\min \mathcal{J}\Big(\{\mathbf{W}^{(v)}\}_V, \{\mathbf{b}^{(v)}\}_V, \mathbf{F}, \boldsymbol{\alpha}, \{\mathbf{E}^{(v,c)}\}_{V,C}\Big) \\
&= \sum_{v=1}^{V}\Big(\alpha^{(v)}\mathcal{J}^{(v)}(\mathbf{W}^{(v)},\mathbf{b}^{(v)},\mathbf{F},\{\mathbf{E}^{(v,c)}\}_C) - \mathrm{sgn}(q)\cdot(\alpha^{(v)})^q\Big) \\
&= \sum_{v=1}^{V}\Big(\alpha^{(v)}\Big(\sum_{i=1}^{n}\sum_{c=1}^{C} f_{ic}^{\gamma}||\mathbf{x}_i^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)} - \mathbf{e}_i^{(v,c)}\odot\mathbf{t}_{(c)} - \mathbf{t}_{(c)}||_2^2 \qquad (9)\\
&\quad + \lambda||\mathbf{W}^{(v)}||_F^2\Big) - \mathrm{sgn}(q)\cdot(\alpha^{(v)})^q\Big) \\
&s.t.\ \mathbf{F}_l = \mathbf{Y}_l, \quad f_{ic} \geq 0, \quad \mathbf{f}_i\mathbf{1}_C = 1, \quad \alpha^{(v)} \geq 0, \quad \mathbf{e}_i^{(v,c)} \geq 0\ (\forall i, c, v)
\end{aligned}$$

where $\mathcal{J}(\cdot)$ and $\mathcal{J}^{(v)}(\cdot)$ represent the unified objective function and the vth view objective function, respectively; q is a hyper-parameter and satisfies $\frac{1}{p}+\frac{1}{q} = 1$; $\boldsymbol{\alpha} = [\alpha^{(1)}, \ldots, \alpha^{(V)}] \in \mathbb{R}_+^{1\times V}$ is a view weight vector, and $\alpha^{(v)}$ refers the importance of the vth view; $\mathbf{E}^{(v,c)} = [\mathbf{e}_1^{(v,c)}; \ldots; \mathbf{e}_n^{(v,c)}] \in \mathbb{R}_+^{n\times C}$ is an introduced adjustment matrix, and $\mathbf{e}_i^{(v,c)}$ is the ith row of $\mathbf{E}^{(v,c)}$. Different from (8), the contributions of various views are directly reflected by the explicitly defined view weight in (9). To solve (9), we adopt an alternative strategy to optimize four groups of variables $\mathbf{F}$, $\{\mathbf{E}^{(v,c)}\}_{V,C}$, $\{\mathbf{W}^{(v)}, \mathbf{b}^{(v)}\}_V$ and $\boldsymbol{\alpha}$ iteratively.

4.1 Optimize probability label matrix F

When $\boldsymbol{\alpha}$, $\{\mathbf{W}^{(v)}\}_V$, $\{\mathbf{b}^{(v)}\}_V$ and $\{\mathbf{E}^{(v,c)}\}_{V,C}$ are fixed, considering the constraint $\mathbf{F}_l = \mathbf{Y}_l$ and the independency of each $\mathbf{f}_i$, we can update $\{\mathbf{f}_i\}_{i=l+1}^{n}$ for unlabeled samples by solving the following u problems independently

$$\begin{aligned} &\min_{\mathbf{f}_i} \sum_{c=1}^{C} f_{ic}^{\gamma} \sum_{v=1}^{V} \alpha^{(v)} ||\mathbf{x}_i^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)} - \mathbf{e}_i^{(v,c)} \odot \mathbf{t}_{(c)} - \mathbf{t}_{(c)}||_2^2 \\ &s.t.\ \mathbf{f}_i \mathbf{1}_C = 1, \quad f_{ic} \geq 0 \ \ (i = l+1, \ldots, n) \end{aligned} \tag{10}$$

Denote $q_{ic} = \sum_{v=1}^{V} \alpha^{(v)} ||\mathbf{x}_i^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)} - \mathbf{e}_i^{(v,c)} \odot \mathbf{t}_{(c)} - \mathbf{t}_{(c)}||_2^2$ as the (i, c)th element of $\mathbf{Q} \in \mathbb{R}^{n \times C}$ and $\mathbf{Q}$ can be calculated based on fixed variables. If $\gamma = 1$, the problem (10) has a trivial solution

$$f_{ic} =< c = \arg\min_{j \in [1,C]} q_{ij} > \tag{11}$$

where $< \cdot >$ is 1 if the argument is true or 0 otherwise. If $\gamma > 1$, setting the derivative of the Lagrangian function of the problem (10) w.r.t f_{ic} to zero and combining the constraint $\sum_{c=1}^{C} f_{ic} = 1$, we arrive the following closed-form solution of the problem (10)

$$f_{ic} = \frac{\left(q_{ic}\right)^{\frac{1}{1-\gamma}}}{\sum_{c=1}^{C} \left(q_{ic}\right)^{\frac{1}{1-\gamma}}} \tag{12}$$

4.2 Optimize adjustment variables $\{\mathbf{E}^{(v,c)}\}_{V,C}$

When $\mathbf{F}$, $\boldsymbol{\alpha}$, $\{\mathbf{W}^{(v)}\}_V$, $\{\mathbf{b}^{(v)}\}_V$ are fixed, considering the independency of each independency of each $\mathbf{e}_i^{(v,c)}$, we can update $\{\mathbf{E}^{(v,c)}\}_{V,C}$ by solving the following $V \times n \times C$ problems simultaneously

$$\min_{\mathbf{e}_i^{(v,c)} \geq 0} \ f_{ic}^{\gamma} ||\mathbf{h}_i^{(v,c)} - \mathbf{e}_i^{(v,c)} \odot \mathbf{t}_{(c)}||^2 \ (\forall v, i, c) \tag{13}$$

where $\mathbf{h}_i^{(v,c)} = \mathbf{x}_i^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)} - \mathbf{t}_{(c)}$. If $i = 1, \ldots, l$ and $y_{ic} = 0$, $\mathbf{e}_i^{(v,c)}$ can be assigned with any values, so we need not to update the corresponding $\mathbf{e}_i^{(v,c)}$; Otherwise, based on the fact that the squared 2-norm of vector can be decoupled element by element, the problem (13) can be further decoupled equivalently into the following C subproblems:

$$\min_{e_{ij}^{(v,c)} \geq 0} \left(h_{ij}^{(v,c)} - e_{ij}^{(v,c)} \odot t_{(c)j}\right)^2 \ (k = 1, \ldots, C) \tag{14}$$

Note that $(t_{(c)j})^2 = 1$. Thus, it is easy to conclude that $\left(h_{ij}^{(v,c)} - e_{ij}^{(v,c)} \odot t_{(c)j}\right)^2 = \left(e_{ij}^{(v,c)} - h_{ij}^{(v,c)} \odot t_{(c)j}\right)^2$. Considering that $e_{ij}^{(v,c)}$ is nonnegative, we can obtain the optimal solution of (13)

$$e_{ij}^{(v,c)} = \begin{cases} e_{ij}^{(v,c)}, & \text{if } y_{ic} = 0 \text{ and } i = 1, \ldots, l \\ \max\left(t_{(c)j} \odot h_{ij}^{(v,c)}, 0\right), & \text{otherwise} \end{cases} \tag{15}$$

4.3 Optimize projection matrices $\{\mathbf{W}^{(v)}\}_V$ and bias vectors $\{\mathbf{b}^{(v)}\}_V$

Given $\boldsymbol{\alpha}$, $\mathbf{F}$ and $\{\mathbf{E}^{(v,c)}\}_{V,C}$ to update $\{\mathbf{W}^{(v)}, \mathbf{b}^{(v)}\}_V$, since the relations among views are decoupled, the problem disassembles into V separate subproblems. By removing the constant term, the vth subproblem ($v = 1, \ldots, V$) can be written as the following matrix form:

$$\begin{aligned}\min_{\mathbf{W}^{(v)},\mathbf{b}^{(v)}} & Tr\Big[\Big(\mathbf{X}^{(v)}\mathbf{W}^{(v)} + \mathbf{1}_n\mathbf{b}^{(v)}\Big)^T \mathbf{U}\Big(\mathbf{X}^{(v)}\mathbf{W}^{(v)} + \mathbf{1}_n\mathbf{b}^{(v)}\Big)\Big] \\ & - 2Tr\Big[\mathbf{M}^{(v)}\Big(\mathbf{X}^{(v)}\mathbf{W}^{(v)} + \mathbf{1}_n\mathbf{b}^{(v)}\Big)^T\Big] + \lambda Tr\Big[\Big(\mathbf{W}^{(v)}\Big)^T\mathbf{W}^{(v)}\Big]\end{aligned} \tag{16}$$

where $\mathbf{U} \in \mathbb{R}^{n\times n}$ is a diagonal matrix, and its (i,i)th element $u_{ii} = \sum_{c=1}^{C}(f_{ic})^{\gamma}$ reflects the importance of the ith data; the ith row of $\mathbf{M}^{(v)} \in \mathbb{R}^{n\times C}$ is computed by $\mathbf{m}_i^{(v)} = \sum_{c=1}^{C}(f_{ic})^{\gamma}\big(\mathbf{e}_i^{(v,c)} \odot \mathbf{t}_{(c)} + \mathbf{t}_{(c)}\big)$. $\mathbf{W}^{(v)}$ and $\mathbf{b}^{(v)}$ can be updated in an alternative way. Setting the derivative of (16) w.r.t variable $\mathbf{b}^{(v)}$ to zero, we have

$$\mathbf{b}^{(v)} = \mathbf{1}_n^T\Big(\mathbf{M}^{(v)} - \mathbf{U}\mathbf{X}^{(v)}\mathbf{W}^{(v)}\Big)/\mathbf{1}_n^T\mathbf{U}\mathbf{1}_n \tag{17}$$

Set the derivative of (16) w.r.t variable $\mathbf{W}^{(v)}$ to zero, then we have

$$\mathbf{W}^{(v)} = \Big(\Big(\mathbf{X}^{(v)}\Big)^T\mathbf{U}\mathbf{X}^{(v)} + \lambda\mathbf{I}_{d^{(v)}}\Big)^{-1}\Big(\mathbf{X}^{(v)}\Big)^T\mathbf{D}^{(v)} \tag{18}$$

where $\mathbf{D}^{(v)} = \mathbf{M}^{(v)} - \mathbf{U}\mathbf{1}_n\mathbf{b}^{(v)}$. When $d^{(v)} < n$, using (18) to update $\mathbf{W}^{(v)}$ is efficient. When $n < d^{(v)}$, since $\mathbf{U}$ is invertible, according to the following identity,

$$\Big(\mathbf{A}^T\mathbf{B}^{-1}\mathbf{A} + \mathbf{C}^{-1}\Big)^{-1}\mathbf{A}^T = \mathbf{C}\mathbf{A}^T\Big(\mathbf{A}\mathbf{C}\mathbf{A}^T + \mathbf{B}\Big)^{-1}\mathbf{B} \tag{19}$$

$\mathbf{W}^{(v)}$ can be efficiently calculated as follows

$$\mathbf{W}^{(v)} = \Big(\mathbf{X}^{(v)}\Big)^T\Big(\mathbf{X}^{(v)}\Big(\mathbf{X}^{(v)}\Big)^T + \lambda\mathbf{U}^{-1}\Big)^{-1}\mathbf{U}^{-1}\mathbf{D}^{(v)} \tag{20}$$

4.4 Optimize view weight vector $\boldsymbol{\alpha}$

With fixed $\{\mathbf{W}^{(v)}\}_V$, $\{\mathbf{b}^{(v)}\}_V$, $\{\mathbf{E}^{(v,c)}\}_{V,C}$ and $\mathbf{F}$, the losses of different views can be calculated accordingly, then we can update $\boldsymbol{\alpha}$ by solving the following V problems independently

$$\min_{\alpha^{(v)}\geq 0} \alpha^{(v)}\mathcal{J}^{(v)}\Big(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F}, \{\mathbf{E}^{(v,c)}\}_C\Big) - \mathrm{sgn}(q)\cdot(\alpha^{(v)})^q \tag{21}$$

Denote $\mathcal{J}^{(v)} = \mathcal{J}^{(v)}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F}, \{\mathbf{E}^{(v,c)}\}_C)$. Setting the derivative of (21) w.r.t $\alpha^{(v)}$ to zero and combining the constraint $\alpha^{(v)} \geq 0$, we obtain the following closed-form solution of the problem (21)

$$\alpha^{(v)} = \Big(\max\Big(\frac{\mathcal{J}^{(v)}}{q\cdot\mathrm{sgn}(q)}, 0\Big)\Big)^{\frac{1}{q-1}} = \Big(\frac{\mathcal{J}^{(v)}}{q\cdot\mathrm{sgn}(q)}\Big)^{\frac{1}{q-1}} \tag{22}$$

According to the above four steps, we alternatively update $\mathbf{F}$, $\{\mathbf{E}^{(v,c)}\}_{V,C}$, $\{\mathbf{W}^{(v)}, \mathbf{b}^{(v)}\}_V$ as well as $\boldsymbol{\alpha}$, and repeat these procedures iteratively until the objective function value of (8) converges. We summarize the iteration process in Algorithm 1. For a testing point $\mathbf{x}_t = [\mathbf{x}_t^{(1)}, \ldots, \mathbf{x}_t^{(V)}]$, its label vector $\mathbf{f}_t$ is calculated by $\mathbf{f}_t = \sum_{v=1}^{V}\alpha^{(v)}(\mathbf{x}_t^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)})$.

Supposing that $\mathbf{f}_i$ is a predicted label vector for an unlabeled training sample or a testing sample, the elements of its binary label vector $\mathbf{y}_i = [y_{i1}, \ldots, y_{iC}] \in \{0, 1\}^{1\times C}$ can be determined by

$$y_{ic} =< c = \mathop{\arg\max}_{j\in[1,C]} f_{ij} > \tag{23}$$

Algorithm 1 Algorithm to solve JCD in Eq. (8)

```
Input:
1. Data for $V$ views $\{\mathbf{X}^{(v)}\}_V, \mathbf{X}^{(v)} \in \mathbb{R}^{n\times d^{(v)}}$.
2. The label matrix $\mathbf{Y}_l \in \mathbb{R}^{l\times C}$ for $l$ labeled instances.
3. The parameters $p$, $\gamma$ and $\lambda$.
Output:
1. The probability label matrix $\mathbf{F} \in \mathbb{R}^{n\times C}$ for all samples.
2. The binary label matrix $\mathbf{Y}_u \in \mathbb{R}^{u\times C}$ for unlabeled training samples with (23).
3. The projection matrices $\{\mathbf{W}^{(v)}\}_V$ and bias vectors $\{\mathbf{b}^{(v)}\}_V$ for each view.
4. The view weight vector $\boldsymbol{\alpha} = [\alpha^{(1)}, \ldots, \alpha^{(V)}]$.
Initialization:
1. Use labeled representations on each view and label matrix to calculated $\mathbf{W}^{(v)}$ and $\mathbf{b}^{(v)}$ by least square classification, $\forall v$;
2. Initialize the weight factor $\alpha^{(v)} = \frac{1}{V}$ for each view, $\forall v$;
3. Initialize $\mathbf{E}^{(v,c)} = 0, \forall v, c$.
Procedure:
While not converged do
1: Calculate $\mathbf{F}$ according to (11) or (12).
2: Compute $\{\mathbf{E}^{(v,c)}\}_{V,C}$ using Eq. (15).
3: Calculate $\{\mathbf{b}^{(v)}\}_V$ by (17) and Compute $\{\mathbf{W}^{(v)}\}_V$ according to (18) or (20).
4: Compute $\boldsymbol{\alpha}$ by Eq. (22).
End While
```

5 Algorithm analysis

In this section, we will give analysis of the proposed Algorithm 1 in two aspects. The convergence behavior is first discussed, then time complexity is analyzed.

5.1 Convergency guarantee

Proposition 1 *The solution of the problem* (8) *can be obtained by solving the problem* (9).

Proof By introducing adjustment variables $\{\mathbf{E}^{(v,c)}\}_{V,C}$, we can infer that

$$\begin{aligned} &\min_{\mathbf{e}_i^{(v,c)}\geq 0} ||\mathbf{x}_i^{(v)}\mathbf{W}^{(v)} + \mathbf{b}^{(v)} - \mathbf{e}_i^{(v,c)} \odot \mathbf{t}_{(c)} - \mathbf{t}_{(c)}||_2^2 \\ &= \sum_{j=1}^{C} \left(1 - t_{(c)j}(\mathbf{x}_i^{(v)}\mathbf{w}_{:j}^{(v)} + b_j^{(v)})\right)_+^2 \end{aligned} \tag{24}$$

which indicates $\min_{\mathbf{E}^{(v,c)}} \mathcal{J}^{(v)}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F}, \{\mathbf{E}^{(v,c)}\}_C) = \mathcal{L}^{(v)}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F})$. Denote $\boldsymbol{\Phi}^{(v)} = \{\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F}\}$. The optimal $\boldsymbol{\Phi}^{(v)}$ and $\alpha^{(v)}$ of the problem (9) can be obtained by solving the following problem

$$\begin{aligned} &\min_{\{\boldsymbol{\Phi}^{(v)},\alpha^{(v)}\}_V} \sum_{v=1}^{V} \Big(\alpha^{(v)}\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)}) - \mathrm{sgn}(q)\cdot(\alpha^{(v)})^q\Big) \\ &s.t.\ \mathbf{F}_l = \mathbf{Y}_l,\quad f_{ic} \geq 0,\quad \sum_{c=1}^{C} f_{ic} = 1,\quad \alpha^{(v)} \geq 0(\forall i, c, v) \end{aligned} \tag{25}$$

Let $\boldsymbol{\Phi} = \{\{\mathbf{W}^{(v)}\}_V, \{\mathbf{b}^{(v)}\}_V, \mathbf{F}\}$ and $\boldsymbol{\Phi}^*$ denotes the optimal $\boldsymbol{\Phi}$ of (9). Combining $\alpha^{(v)} = \big(\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})/(q\cdot\mathrm{sgn}(q))\big)^{\frac{1}{q-1}}$ and considering $1/p + 1/q = 1$, $p < 1$ and $p \neq 0$, $\boldsymbol{\Phi}^*$ can be obtained from the following equivalent problems:

$$\begin{aligned} &\min_{\boldsymbol{\Phi}\in\mathcal{C}} \sum_{v=1}^{V} \left(\left(\frac{\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})}{q\cdot\mathrm{sgn}(q)}\right)^{\frac{1}{q-1}} \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)}) - \mathrm{sgn}(q)\cdot\left(\frac{\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})}{q\cdot\mathrm{sgn}(q)}\right)^{\frac{q}{q-1}}\right) \\ &\simeq \min_{\boldsymbol{\Phi}\in\mathcal{C}} \sum_{v=1}^{V} \left(\left(\frac{1}{q\cdot\mathrm{sgn}(q)}\right)^{\frac{1}{q-1}} - \mathrm{sgn}(q)\cdot\left(\frac{1}{q\cdot\mathrm{sgn}(q)}\right)^{\frac{q}{q-1}}\right)\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{\frac{q}{q-1}} \\ &\simeq \min_{\boldsymbol{\Phi}\in\mathcal{C}} \sum_{v=1}^{V} \left(\frac{1}{q\cdot\mathrm{sgn}(q)}\right)^{\frac{1}{q-1}} \left(1-\frac{1}{q}\right)\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{\frac{q}{q-1}} \end{aligned} \tag{26}$$

where $\mathcal{C}$ are the constraints corresponding to $\boldsymbol{\Phi}$. Denote $C_q = (q\cdot\mathrm{sgn}(q))^{1/(1-q)}$. When $p < 1$ and $p \neq 0$, according to $1/p + 1/q = 1$, we can conclude that $C_q > 0$, then it is equivalent to solve the following problems to obtain $\boldsymbol{\Phi}^*$:

$$\begin{aligned} \boldsymbol{\Phi}^* &= \arg\min_{\boldsymbol{\Phi}\in\mathcal{C}} \sum_{v=1}^{V} \frac{C_q\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^p}{p} = \arg\min_{\boldsymbol{\Phi}\in\mathcal{C}} \sum_{v=1}^{V} \mathrm{sgn}(p)\cdot\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^p \\ &= \arg\min_{\boldsymbol{\Phi}\in\mathcal{C}} \sqrt[p]{\frac{1}{V}\sum_{v=1}^{V}\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^p} = \arg\min_{\boldsymbol{\Phi}\in\mathcal{C}} \mathcal{M}_p(\{\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})\}_V) \end{aligned} \tag{27}$$

which completes the proof. □

Proposition 2 *Algorithm 1 will monotonically decrease the objective function of (8) in each iteration until it converges to a stationary point.*

Proof Suppose the updated $\mathbf{F}$, $\mathbf{E}^{(v,c)}$, $\mathbf{W}^{(v)}$ and $\mathbf{b}^{(v)}$ of Algorithm 1 are denoted as $\tilde{\mathbf{F}}$, $\tilde{\mathbf{E}}^{(v,c)}$, $\tilde{\mathbf{W}}^{(v)}$ and $\tilde{\mathbf{b}}^{(v)}$, respectively. As shown in Algorithm 1, the optimization of the problem (9) can be divided into four subproblems. Therefore, by finding the optimal solution of each subproblem, it can be concluded that

$$\begin{aligned} &\sum_{v=1}^{V} \Big(\alpha^{(v)}\mathcal{J}^{(v)}(\tilde{\mathbf{W}}^{(v)}, \tilde{\mathbf{b}}^{(v)}, \tilde{\mathbf{F}}, \{\tilde{\mathbf{E}}^{(v,c)}\}_C) - \mathrm{sgn}(q)\cdot(\alpha^{(v)})^q\Big) \\ &\leq \sum_{v=1}^{V} \Big(\alpha^{(v)}\mathcal{J}^{(v)}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F}, \{\mathbf{E}^{(v,c)}\}_C) - \mathrm{sgn}(q)\cdot(\alpha^{(v)})^q\Big) \end{aligned} \tag{28}$$

Denote the updated $\boldsymbol{\Phi}^{(v)}$ as $\tilde{\boldsymbol{\Phi}}^{(v)} = \{\tilde{\mathbf{W}}^{(v)}, \tilde{\mathbf{b}}^{(v)}, \tilde{\mathbf{F}}\}$. Based on (24) and (28), it can be inferred that

$$\sum_{v=1}^{V}\alpha^{(v)}\mathcal{L}^{(v)}(\tilde{\boldsymbol{\Phi}}^{(v)}) \leq \sum_{v=1}^{V}\alpha^{(v)}\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)}) \tag{29}$$

Combing $\alpha^{(v)} = C_q \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{\frac{1}{q-1}} = C_q \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{p-1}$, it can be concluded that

$$\sum_{v=1}^{V} \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{p-1} \mathcal{L}^{(v)}(\tilde{\boldsymbol{\Phi}}^{(v)}) \leq \sum_{v=1}^{V} \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{p-1} \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)}) \tag{30}$$

Since $p < 1$ and $p \neq 0$, we define the function $g(x) = \text{sgn}(p) \cdot x^p$, then

$$g\Big(\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})\Big) = \text{sgn}(p) \cdot \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^p \tag{31}$$

$g(\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)}))$ is a concave function in the domain of $\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})$. The supergradient of $g(x)$ can be calculated by $g'(x) = \text{sgn}(p) \cdot px^{p-1} = |p|x^{p-1}$, then

$$g'\Big(\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})\Big) = |p|\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{p-1} \tag{32}$$

According to the definition of supergradient, we have:

$$g\Big(\mathcal{L}^{(v)}(\tilde{\boldsymbol{\Phi}}^{(v)})\Big) - g\Big(\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})\Big) \leq g'\Big(\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})\Big)\Big(\mathcal{L}^{(v)}(\tilde{\boldsymbol{\Phi}}^{(v)}) - \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})\Big) \tag{33}$$

Thus, we have

$$\begin{aligned} &\sum_{v=1}^{V} \text{sgn}(p) \cdot \mathcal{L}^{(v)}(\tilde{\boldsymbol{\Phi}}^{(v)})^p - \sum_{v=1}^{V} |p|\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{p-1}\mathcal{L}^{(v)}(\tilde{\boldsymbol{\Phi}}^{(v)}) \\ &\quad \leq \sum_{v=1}^{V} \text{sgn}(p) \cdot \mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^p - \sum_{v=1}^{V} |p|\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)})^{p-1}\mathcal{L}^{(v)}(\boldsymbol{\Phi}^{(v)}) \end{aligned} \tag{34}$$

Combining (30) and (34), it arrives at

$$\begin{aligned} &\sum_{v=1}^{V} \text{sgn}(p) \cdot \mathcal{L}^{(v)}(\tilde{\mathbf{W}}^{(v)}, \tilde{\mathbf{b}}^{(v)}, \tilde{\mathbf{F}})^p \leq \sum_{v=1}^{V} \text{sgn}(p) \cdot \mathcal{L}^{(v)}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F})^p \\ &\Rightarrow \mathcal{M}_p\Big(\Big\{\mathcal{L}^{(v)}(\tilde{\mathbf{W}}^{(v)}, \tilde{\mathbf{b}}^{(v)}, \tilde{\mathbf{F}})\Big\}_V\Big) \leq \mathcal{M}_p\Big(\Big\{\mathcal{L}^{(v)}(\mathbf{W}^{(v)}, \mathbf{b}^{(v)}, \mathbf{F})\Big\}_V\Big) \end{aligned} \tag{35}$$

□

Thus Algorithm 1 will monotonically decrease the objective of (8) in each iteration until it converges. In the convergence, the equality in Eq. (35) holds, thus $\{\tilde{\mathbf{W}}^{(v)}\}_V$, $\{\tilde{\mathbf{b}}^{(v)}\}_V$, $\tilde{\mathbf{F}}$ will satisfy the KKT condition of problem (8). Therefore, Algorithm 1 will converge to a stationary point of the problem (8).

5.2 Computational complexity

As seen from Algorithm 1, we solve the problem (8) and (9) in an alternative way. The computation complexity of updating $\mathbf{F}$ is $O(uC^2)$. The updating of $\{\mathbf{E}^{(v,c)}\}_{V,C}$ and the calculation of $\boldsymbol{\alpha}$ can be completed together with $O(VnC)$ computations. The total time complexity of computing $\{\mathbf{b}^{(v)}\}_V$ is $O(ndC)$. To update $\mathbf{W}^{(v)}$, when $d^{(v)} < n$, it costs $O(n(d^{(v)})^2)$ for matrix multiplication and $O((d^{(v)})^3)$ for matrix inversion; when $d^{(v)} > n$, it costs $O(n^2 d^{(v)})$ for matrix multiplication and $O(n^3)$ for matrix inversion. The total time complexity of computing $\{\mathbf{W}^{(v)}\}_V$ is $O(\sum_{v=1}^{V} \max(n, d^{(v)}) \min(n, d^{(v)})^2)$. Since $C \ll n$ and $V \ll n$, the time complexity of Algorithm 1 is $O(T \sum_{v=1}^{V} \max(n, d^{(v)}) \min(n, d^{(v)})^2)$, where T is the total number of iterations.

6 Experiments

In this section, to validate the effectiveness and superiority of the proposed model, we compare our proposed JCD with related semi-supervised classification methods in terms of classification accuracy and F-score on nine benchmark datasets. Then we present the convergence behavior curves and comparison of computational time. Lastly, we evaluate the impact of parameters on our proposed algorithm.

6.1 Data set descriptions

MSRC-v1 data set is composed of 240 images and divided into 8 categories. Following (Lee and Kristen 2009), 7 classes composed of *tree*, *building*, *airplane*, *cow*, *face*, *car*, *bicycle* are selected, and each class has 30 images. Since there is no published image descriptors, six popular features are extracted for each image: i.e. 256 local binary pattern (LBP), 100 histogram of oriented gradient (HOG), 512 GIST, 1302 CENTRIST, 48 color moment (CMT) and 200 SIFT features.

Caltech7 data set consists 8677 objective images, each with 0.1 mega pixel resolution, belonging to 101 classes. Following (Dueck and Frey 2007), 7 widely used classes with total 441 images are selected, including *Dolla-Bill*, *Faces*, *Garfield*, *Motorbikes*, *Snoopy*, *Stop-Sign* and *Windsor-Chair*. For each image, the same six visual features are extracted as MSRC-v1 data set.

Digits data set (Asuncion and Newman 2007) contains 2,000 data points for 0 to 9 ten digit classes, and each class has 200 data points. Six public features are available: 76 Fourier coefficients of the character shapes (FOU), 216 profile correlations (FAC), 64 Karhunen-love coefficients (KAR), 240 pixel averages in 2 × 3 windows (PIX), 47 Zernike moment (ZER) and 6 morphological (MOR) features.

Scene15 data set is composed of 4485 images belonging to 15 categories: highway (260 images), inside of cities (308 images), tall buildings (356 images), streets (292 images), suburb residence (241 images), forest (328 images), coast (360 images), mountain (374 images), open country (410 images), bedroom (216 images), kitchen (210 images), livingroom (289 images), office (215 images), industrial (311 images) and store (315 images). Following (Tao et al. 2017), six visual features are extracted: 200 SIFT, 200 SURF, 680 PHOG, 256 LBP, 512 GIST and 32 WT features.

WebKB data set (Sindhwani et al. 2005) consists 1051 web documents from four universities. The 1051 pages are classified into 2 classes: 230 Course pages and 821 Non-Course pages. Each page has two views: Fulltext view with 2949 features represents the textual content on the web page, while Inlinks view with 334 features records that the anchor text on the hyperlinks pointing to the pages.

BBCsport (Greene and Cunningham 2009) consists news about athletics, cricket, football, rugby, tennis. Each raw document is split into segments, and segments are randomly assigned to views. Two datasets are used: **BBCsport2** consists 544 documents, which have 2 views with 3183 and 3203 features; **BBCsport3** consists 282 documents, which have 3 views with 2582, 2544 and 2465 features.

Kinect skeleton action (KSA) data set (Ma et al. 2014) includes four subjects performing five actions, namely boxing, gesturing, jogging, throw-catch and walking. KSA consists 20,000 video frames with 4,000 for each subject. Each frame has two views with 120 and 10 features.

MNIST8M data set (Loosli et al. 2007) is composed of 8100,000 handwritten digits from 0 to 9. The digits have been normalized in 28 × 28 images. From each digit, 10,000 examples are randomly selected, forming a subset (MNIST) of 100,000 examples. Three features are extracted: 100 SIFT, 100 SURF and 32 WT.

6.2 Experiment setup

We compare our proposed JCD with several state-of-the-art semi-supervised classification algorithms, including adaptive semi-supervised learning (ASL) (Wang et al. 2014), auto-weighted multiple graph learning (AMGL) (Nie et al. 2016), multi-view learning with adaptive neighbors (MLAN) (Nie et al. 2018), multi-feature learning via hierarchical regression (MLHR) (Yang et al. 2013) and multi-view semi-supervised learning via adaptive regression (MVAR) (Tao et al. 2017).

ASL is a single-view regression-based method, which learns a linear classifier and a probability matrix for unlabeled training samples simultaneously. ASL is implemented on each view matrix and the concatenated feature matrix of all views. The best single view results and the results corresponding to the the concatenated data are reported by S-ASL and C-ASL, respectively. AMGL is a multi-view graph-based method, which jointly performs label propagation and view weight learning. MLHR is a multi-view regression-based method, which learns local and global linear regression models. MLAN and MVAR are introduced in Sects. 2.1 and 2.2.

For each dataset except MNIST, two thirds of instances are randomly selected as the training data, while the remaining ones are served as the testing data. On MNIST, one fives of examples are randomly selected as the training set, and the remaining fours are testing data. To mimic the real situation ($l \ll u$), we choose only 10% or 20% samples with labels randomly in the training stage on these datasets except KSA and MNIST. On KSA dataset, only 1% or 2% samples are randomly selected to assign labels. And on MNIST, only 3% or 6% samples are randomly selected to assign labels.

The classification performance is evaluated in terms of classification accuracy and F-score. In the experiments, the stop criteria of our proposed JCD is defined as following:

$$\frac{\mathcal{L}(t-1)-\mathcal{L}(t)}{\mathcal{L}(t-1)} < 10^{-4}$$

where $\mathcal{L}(t)$ is the objective value of (8) in the tth iteration.

For JCD, the linear model is used in all experiments. The adaptive parameter γ is tuned in the range of $\{1.1, 1.3, 1.5, 1.7, 1.9, 2.5, 3.3\}$ and the balanced parameter λ is tuned from $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$. Following (Huang et al. 2019; Nie et al. 2016, 2018; Shu et al. 2017; Zhuge et al. 2017), the parameter p is set to be $\frac{1}{2}$. For the compared methods, we download their codes form authors' websites and determine the searching ranges of the parameters according to their papers. All hyper-parameters are tuned by grid search on the testing data, and the classification results of using the best tuned parameters are recorded.

6.3 Classification results comparison

For a fair comparison, each data set is randomly split into training and testing dataset 10 times, and we report the average accuracy with standard deviation (STD) and the average F-score for the unlabeled training and testing data. Tables 1 and 2 show the classification accuracy results on nine datasets with different percentage of training labeled samples, where "NA" indicates that the transductive methods can not predict labels for testing samples.

From Tables 1, 2 and Fig. 1, we can conclude that:

Table 1 The classification accuracy (%) on seven data sets with different percentages (τ%) of labeled samples

Data	τ%	10		20	
set	Method	Unlabeled	Testing	Unlabeled	Testing
MSRC-v1	S-ASL	72.94(4.42)	69.43(3.60)	78.93(5.40)	77.71(3.51)
	C-ASL	76.19(8.65)	75.43(11.08)	86.34(3.23)	84.86(3.70)
	AMGL	84.21(4.00)	NA	88.57(2.05)	NA
	MLAN	81.98(1.83)	NA	85.71(3.12)	NA
	MLHR	84.29(2.50)	78.86(5.50)	90.27(2.32)	89.29(2.54)
	MVAR	85.48(2.78)	84.86(2.54)	90.18(3.12)	90.29(2.84)
	JCD	88.89(2.67)	87.43(3.22)	93.48(2.92)	92.29(2.71)
Caltech7	S-ASL	82.84(3.30)	81.32(3.40)	86.14(2.61)	85.56(2.41)
	C-ASL	85.68(2.36)	84.77(2.32)	89.43(3.14)	89.27(2.75)
	AMGL	80.51(3.95)	NA	86.93(2.38)	NA
	MLAN	81.71(2.50)	NA	85.31(2.24)	NA
	MLHR	75.53(3.89)	74.44(2.80)	84.47(2.49)	82.91(1.39)
	MVAR	84.44(2.26)	85.63(3.30)	86.40(2.12)	88.28(1.71)
	JCD	86.85(2.03)	86.69(2.39)	90.26(1.70)	90.53(0.99)
Digits	S-ASL	94.23(0.43)	94.12(0.75)	95.21(0.58)	94.85(0.70)
	C-ASL	95.56(0.36)	95.66(0.79)	96.07(0.49)	95.85(0.74)
	AMGL	90.91(1.43)	NA	93.87(0.62)	NA
	MLAN	96.80(0.42)	NA	97.37(0.39)	NA
	MLHR	93.51(0.77)	93.55(0.76)	95.54(0.62)	95.57(0.69)
	MVAR	95.94(0.71)	95.43(0.96)	96.45(0.52)	95.84(0.58)
	JCD	96.95(0.55)	96.93(0.80)	97.45(0.51)	97.39(0.35)
Scene15	S-ASL	54.39(1.07)	54.37(2.06)	59.61(1.27)	59.09(1.29)
	C-ASL	54.67(1.53)	53.15(1.92)	56.78(1.60)	56.31(1.10)
	AMGL	60.88(1.30)	NA	67.64(0.91)	NA
	MLAN	60.64(0.76)	NA	64.07(1.15)	NA
	MLHR	62.97(1.63)	62.85(1.50)	69.92(1.39)	69.43(0.77)
	MVAR	59.78(0.82)	53.17(1.37)	68.65(0.96)	65.07(0.72)
	JCD	67.71(1.06)	68.94(1.43)	73.52(1.07)	73.28(1.29)
WebKB	S-ASL	92.10(1.23)	91.91(1.98)	92.79(1.02)	92.17(1.21)
	C-ASL	90.33(2.29)	88.75(1.82)	92.99(1.23)	91.40(1.00)
	AMGL	84.86(2.62)	NA	92.63(1.24)	NA
	MLAN	84.32(1.39)	NA	86.76(1.40)	NA
	MLHR	91.97(0.99)	91.42(1.81)	92.65(1.25)	93.11(1.05)
	MVAR	91.40(2.27)	91.51(1.89)	93.29(1.17)	93.05(1.38)
	JCD	92.88(0.97)	92.05(1.39)	94.51(1.43)	93.59(0.89)
BBCsport2	S-ASL	77.70(5.89)	70.38(6.54)	85.38(3.14)	77.72(3.15)
	C-ASL	85.00(3.24)	80.54(5.24)	89.58(2.17)	86.52(4.25)
	AMGL	55.65(9.72)	NA	70.66(5.09)	NA
	MLAN	41.74(6.61)	NA	47.41(4.38)	NA
	MLHR	71.02(6.52)	73.10(3.42)	83.50(2.54)	86.41(4.40)

Table 1 continued

Data	τ%	10		20	
set	Method	Unlabeled	Testing	Unlabeled	Testing
	MVAR	68.04(4.15)	67.88(5.92)	78.71(1.60)	85.00(4.66)
	JCD	89.81(2.61)	91.30(3.02)	93.36(2.21)	94.57(2.76)
BBCsport3	S-ASL	73.84(3.85)	66.04(5.22)	76.46(2.08)	71.04(3.32)
	C-ASL	81.71(2.07)	75.31(3.71)	82.86(2.54)	78.44(3.15)
	AMGL	50.79(4.71)	NA	64.15(6.91)	NA
	MLAN	38.90(6.07)	NA	44.90(8.20)	NA
	MLHR	69.27(4.27)	71.25(4.77)	80.95(2.74)	82.71(2.26)
	MVAR	67.68(7.77)	67.29(9.57)	78.23(2.57)	80.10(1.73)
	JCD	86.59(1.75)	86.04(4.40)	87.41(2.03)	88.33(1.28)

Standard deviation (%) is in the parentheses

Table 2 The classification accuracy (%) on two data sets with different percentages (τ%) of labeled samples

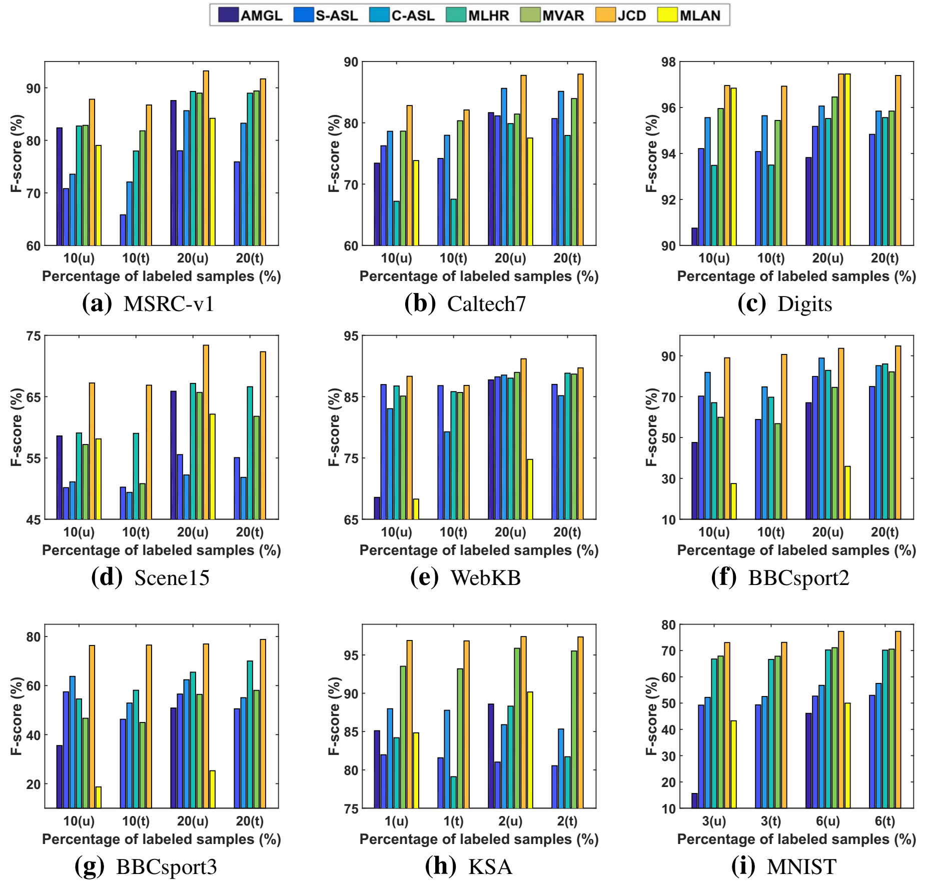
Data	τ%	τ_{KSA}=1, τ_{MNIST}=3		τ_{KSA}=2, τ_{MNIST}=6	
set	Method	Unlabeled	Testing	Unlabeled	Testing
KSA	S-ASL	82.39(3.26)	82.04(3.40)	81.63(2.03)	81.18(2.13)
	C-ASL	88.61(6.59)	88.45(6.79)	86.39(4.99)	85.84(5.11)
	AMGL	85.02(1.95)	NA	88.55(0.95)	NA
	MLAN	85.34(6.26)	NA	90.11(1.72)	NA
	MLHR	84.25(2.26)	79.33(2.59)	88.31(1.42)	81.80(1.62)
	MVAR	93.55(1.36)	93.22(1.59)	95.88(0.69)	95.52(0.73)
	JCD	96.90(1.88)	96.85(1.71)	97.41(1.13)	97.37(1.22)
MNIST	S-ASL	52.40(1.00)	52.35(0.87)	54.48(0.56)	54.72(0.35)
	C-ASL	53.83(0.85)	54.19(0.71)	58.21(0.62)	58.90(0.37)
	AMGL	20.63(0.42)	NA	45.29(1.15)	NA
	MLAN	44.67(2.61)	NA	50.82(2.43)	NA
	MLHR	68.13(0.78)	67.98(0.57)	71.35(0.57)	71.26(0.54)
	MVAR	69.12(1.09)	69.21(0.67)	72.16(0.72)	71.75(0.55)
	JCD	73.58(0.60)	73.65(0.27)	77.69(0.47)	77.70(0.31)

Standard deviation (%) is in the parentheses

(1) All methods achieve better performance as the increase of labeled data in the training stages in most cases, which is consistent with intuition.

(2) The performance of graph-based methods are unstable. On Digits, MLAN ranks second, while it performs worse than other multi-view methods on Sence15. On BBCsport2, BBCsport3 and MNIST, AMGL and MLAN perform much worse than other methods. This is probably because that the graph learning is based on the original data representations and the performance may suffer from redundant features.

(3) In Caltech7, WebKB, BBCsport2, BBCsport3 and MNIST, the performance of S-ASL is not the worst. This on the opposite side illustrates that the performance of multi-view methods will not be enhanced if the multiple representations are not properly integrated.



(a) MSRC-v1 **(b)** Caltech7 **(c)** Digits

(d) Scene15 **(e)** WebKB **(f)** BBCsport2

(g) BBCsport3 **(h)** KSA **(i)** MNIST

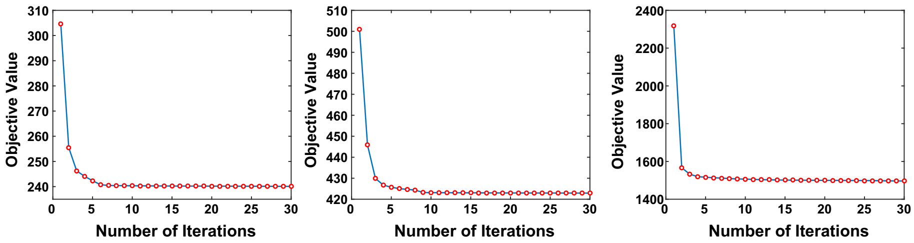
Fig. 1 F-score comparison on nine data sets with different percentages of labeled samples. (u) and (t) denote the results on unlabeled training data and testing data, respectively

(4) Since our model makes use of both the consensus and diversity information of multi-view data, and takes the contribution importance of instances into consideration, together with learning the view weight factors, it consistently outperforms the compared methods in terms of both classification accuracy and F-score over all datasets.

6.4 Convergence analysis and time comparison

In order to verify the convergence of Algorithm 1, we plot the corresponding convergence curves of the objective function (8) on datasets MSRC-v1, Caltech7 and Digits, when the percentage of labeled samples is 10%. As seen from Fig. 2, the objective function value monotonically decreases as the iteration round increases and converges to a fixed value. Additionally, the algorithm converges within 20 iterations over all datasets, validating the efficiency and fine convergence speed of this algorithm.

To demonstrate the efficiency of JCD, we reports the training time of six methods on nine datasets. Note that C-ASL is the only single-view method. All algorithms are performed on



(a) MSRC-v1 **(b)** Caltech7 **(c)** Digits

Fig. 2 Convergence curves of the objective function values in (8)

Table 3 Average training time (seconds) on nine datasets

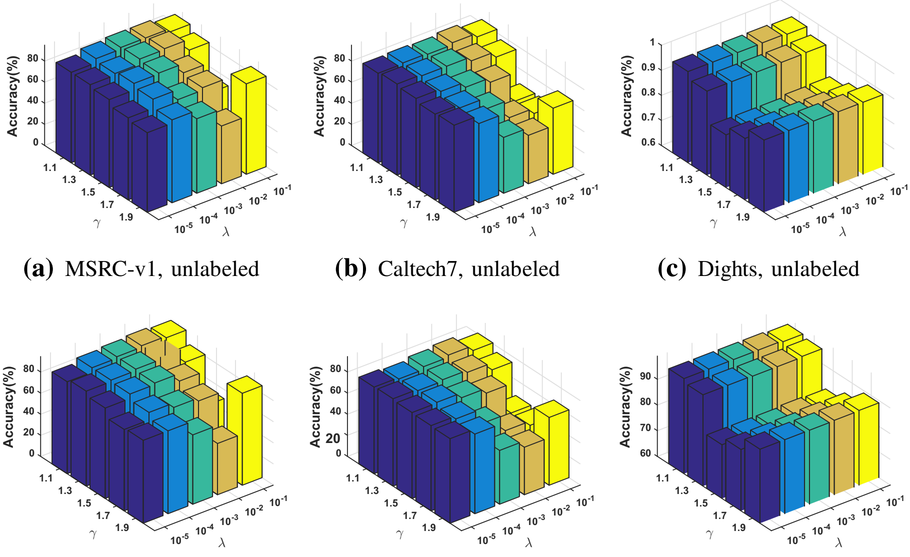
Data	C-ASL	AMGL	MLAN	MLHR	MVAR	JCD
MSRC-v1	4.0554	0.0204	0.0819	0.4559	0.0481	0.0472
Caltech7	7.3391	0.0604	0.2300	1.7165	0.2077	0.1667
Dights	3.2222	2.1310	5.7958	9.5113	1.0560	0.5195
Scene15	20.2448	12.5792	72.0071	177.5527	10.9998	5.3850
WebKB	16.6136	0.2418	1.1649	15.9556	1.0314	0.8337
BBCsport2	20.3666	0.1073	0.1798	9.9832	0.3838	0.1469
BBCsport3	24.1512	0.0528	0.0846	3.3156	0.1755	0.0671
KSA	679.1013	362.0398	3234.6878	407.3175	84.9592	36.7199
MNIST	1013.5178	1228.9081	10472.7038	1600.6276	109.9823	54.0728

a work station with 4 processors (3.4 GHz for each) and 32GB memory, using MATLAB R2017a. With predetermined parameters, each method is implemented for 5 independent times. The average time are reported in Table 3.

From the experimental comparison, we have the following observations: (1) On datasets MSRC-v1, Caltech7, WebKB, BBCsport2 and BBCsport3, which have much larger dimensionality than data size, AMGL takes the least time because it only has linear complexity w.r.t the dimensionality on the construction of graph matrices. The proposed JCD spends less time than other methods except AMGL. C-ASL spends the most time because it not only has cubic time complexity w.r.t the dimensionality but also needs iterations. (2) On datasets Digits and Scene15, which have comparable data size and dimensionality on some views, JCD spends the least time. MLHR consumes the most time because it has cubic complexity w.r.t both the data size and the dimensionality. (3) On datasets KSA and MNIST, which have much larger data size than dimensionality, JCD consumes less time than other methods. JCD and MVAR have comparable computational burden in each iteration. JCD costs less time because it has faster convergence speed. MLAN costs much more time than other methods because it not only has high complexity w.r.t the data size but also needs iterations.

6.5 Parameter determination

To illustrate the influence of parameters γ and λ on the performance of the proposed JCD, we present the classification accuracy results with varying parameters on three datasets, i.e., MSRC-v1, Caltech7 and Digits. We vary γ within the range $\{1.1, 1.3, 1.5, 1.7, 1.9\}$. Another



(a) MSRC-v1, unlabeled **(b)** Caltech7, unlabeled **(c)** Dights, unlabeled

(d) MSRC-v1, testing **(e)** Caltech7, testing **(f)** Dights, testing

Fig. 3 Sensitivity analysis on parameters γ and λ with 10% labeled samples

Table 4 The F-score (%) on three data sets with different p

Dataset	p	0.2	0.4	0.5	0.6	0.8	1
MSRC-v1	Unlabeled	88.83	89.19	89.17	89.50	90.09	90.17
	Testing	88.03	88.23	88.06	89.14	89.88	90.28
Caltech7	Unlabeled	81.03	80.14	79.65	79.43	75.92	64.83
	Testing	78.83	79.05	78.81	78.19	73.57	63.94
Dights	Unlabeled	97.12	96.99	96.97	96.91	96.89	96.87
	Testing	97.04	97.13	97.07	96.98	96.92	96.74

parameter λ is varied from $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. p is fixed as 0.5 and 10% training samples are randomly selected with labels.

As we can see from the results in Fig. 3, if the parameters are determined with suitable values on the training stage, the proposed JCD also achieves satisfactory performance on the testing stage with the same parameters. However, how to identify the optimal parameters is data dependent. Three datasets have different optimal parameters because their data characteristics are different.

To show the influence of the view weight parameter p on the performance of the proposed JCD, the F-score results with varying p on three datasets MSRC-v1, Caltech7 and Digits are presented. With fixed γ and λ, p is varied from $\{0.2, 0.4, 0.5, 0.6, 0.8, 1\}$. 20% training samples are randomly selected with labels. The average results of 5 independent times are reported in Table 4.

From Table 4, we have the following observations: (1) On MSRC-v1, JCD tends to achieve better performance with the increase of p. That is probably because the views with large losses contain complementary information on this dataset, and the performance may be improved by making full use of them. (2) As p increases, the performance of JCD drops significantly on Caltech 7, and it decreases slowly on Dights. That is probably because the views with large losses contain redundant information in these datasets, and the performance may be improved by reducing their influence. (3) When $0.4 \leq p \leq 0.6$, JCD achieves satisfactory performance on all three datasets.

7 Conclusion

In this paper, we propose a multi-view semi-classification algorithm named as JCD, which exploits both consensus and diversity information. Following the consensus principle, JCD learns a common probability label matrix, which ensures the classification consensus. Following the diversity principle, JCD learns view-specific classifiers, and weights various views and samples automatically, which make it robust against the existence of low-quality views and boundary instances. An optimization algorithm to efficiently solve the proposed non-smooth objective is introduced with proved convergence. Extensive experimental results show that JCD achieves superior performance.

There are several interesting directions to study in the future: First, we would like to design new regularization terms based on instance weight learning strategy for semi-supervised learning; Second, how to extend JCD for the incomplete multi-view data is also an interesting problem.

References

Asuncion, A., & Newman, D. J. (2007). *UCI machine learning repository*. Irvine: Irvine University of California.

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In: Conference on computational learning theory, pp. 92–100.

Cai, X., Nie, F., Cai, W., & Huang, H. (2013). Heterogeneous image features integration via multi-modal semi-supervised learning model. In: IEEE international conference on computer vision, pp. 1737–1744.

Chen, X., Chen, S., Xue, H., & Zhou, X. (2012). A unified dimensionality reduction framework for semi-paired and semi-supervised multi-view data. *Pattern Recognition*, *45*(5), 2005–2018.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In: IEEE conference on computer vision and pattern recognition, pp. 886–893.

Dueck, D., & Frey, B.J. (2007). Non-metric affinity propagation for unsupervised image categorization. In: IEEE international conference on computer vision, pp. 1–8.

Gong, C. (2017). Exploring commonality and individuality for multi-modal curriculum learning. In: Thirty-first AAAI conference on artificial intelligence, pp. 1926–1933.

Gong, C., Tao, D., Chang, X., & Yang, J. (2017). Ensemble teaching for hybrid label propagation. *IEEE transactions on cybernetics*, *49*(2), 388–402.

Gong, C., Tao, D., Maybank, S. J., Liu, W., Kang, G., & Yang, J. (2016). Multi-modal curriculum learning for semi-supervised image classification. *IEEE Transactions on Image Processing*, *25*(7), 3249–3260.

Gong, C., Tao, D., Yang, J., & Liu, W. (2016). Teaching-to-learn and learning-to-teach for multi-label propagation. In: Thirtieth AAAI conference on artificial intelligence, pp. 1610–1616.

Greene, D., & Cunningham, P. (2009). A matrix factorization approach for integrating multiple data views. In: European conference on machine learning and knowledge discovery in databases, pp. 423–438.

Guz, U., & Tur, G. (2009). Multi-view semi-supervised learning for dialog act segmentation of speech. *IEEE Transactions on Audio Speech and Language Processing*, *18*(2), 320–329.

Hou, C., Zhang, C., Wu, Y., & Nie, F. (2010). Multiple view semi-supervised dimensionality reduction. *Pattern Recognition*, *43*(3), 720–730.

Huang, S., Kang, Z., Tsang, I. W., & Xu, Z. (2019). Auto-weighted multi-view clustering via kernelized graph learning. *Pattern Recognition*, *88*, 174–184.

Karasuyama, M., & Mamitsuka, H. (2013). Multiple graph label propagation by sparse integration. *IEEE Transactions on Neural Networks and Learning Systems*, *24*(12), 1999–2012.

Lee, Y. J., & Kristen, G. (2009). Foreground focus: unsupervised learning from partially matching images. *International Journal of Computer Vision*, *85*(2), 143–166.

Loosli, G., Canu, S., & Bottou, L. (2007). Training invariant support vector machines using selective sampling. In L. Bottou, O. Chapelle, D. DeCoste, & J. Weston (Eds.), *Large scale kernel machines* (pp. 301–320). Cambridge: MIT Press.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Luo, M., Zhang, L., Nie, F., Chang, X., Qian, B., & Zheng, Q. (2017). Adaptive semi-supervised learning with discriminative least squares regression. In: International joint conference on artificial intelligence, pp. 2421–2427.

Ma, Z., Yang, Y., Nie, F., Sebe, N., Yan, S., & Hauptmann, A. G. (2014). Harnessing lab knowledge for real-world action recognition. *International Journal of Computer Vision*, *109*(1–2), 60–73.

Mao, C.H., Lee, H.M., Parikh, D., Chen, T., & Huang, S.Y. (2009). Semi-supervised co-training and active learning based approach for multi-view intrusion detection. In: ACM symposium on applied computing, pp. 2042–2048.

Nie, F., Cai, G., Li, J., & Li, X. (2018). Auto-weighted multi-view learning for image clustering and semi-supervised classification. *IEEE Transactions on Image Processing*, *27*(3), 1501–1511.

Nie, F., Li, J., & Li, X. (2016). Parameter-free auto-weighted multiple graph learning: a framework for multiview clustering and semi-supervised classification. In: International joint conference on artificial intelligence, pp. 1881–1887.

Nie, F., Tian, L., & Li, X. (2018). Multiview clustering via adaptively weighted procrustes. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery and data mining, pp. 2022–2030.

Nie, F., Wang, X., & Huang, H. (2017). Multiclass capped $\updownarrow_p$-norm svm for robust classifications. In: Thirty-first AAAI conference on artificial intelligence, pp. 2415–2421.

Nigam, K., & Ghani, R. (2000). Analyzing the effectiveness and applicability of co-training. In: International conference on information and knowledge management, pp. 86–93.

Oliva, A., & Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, *42*(3), 145–175.

Shu, Z., Wu, X., Fan, H., Huang, P., Wu, D., Hu, C., et al. (2017). Parameter-less auto-weighted multiple graph regularized nonnegative matrix factorization for data representation. *Knowledge-Based Systems*, *131*, 105–112.

Sindhwani, V., Niyogi, P., & Belkin, M. (2005). Beyond the point cloud: from transductive to semi-supervised learning. In: International conference on machine learning, pp. 824–831.

Sun, S., & Jin, F. (2011). Robust co-training. *International Journal of Pattern Recognition and Artificial Intelligence*, *25*(07), 1113–1126.

Tao, H., Hou, C., Nie, F., Zhu, J., & Yi, D. (2017). Scalable multi-view semi-supervised classification via adaptive regression. *IEEE Transactions on Image Processing*, *26*(9), 4283–4296.

Wang, D., Nie, F., & Huang, H. (2014). Large-scale adaptive semi-supervised learning via unified inductive and transductive model. In: ACM SIGKDD international conference on knowledge discovery and data mining, pp. 482–491.

Xu, C., Tao, D., & Xu, C. (2013). *A survey on multi-view learning*. : Computer Science.

Xu, Z., & King, I. (2014). *Introduction to semi-supervised learning*. San Rafael: Morgan and Claypool.

Yang, Y., Song, J., Huang, Z., Ma, Z., Sebe, N., & Hauptmann, A. G. (2013). Multi-feature fusion via hierarchical regression for multimedia analysis. *IEEE Transactions on Multimedia*, *15*(3), 572–581.

Yu, J., Wang, M., & Tao, D. (2012). Semisupervised multiview distance metric learning for cartoon synthesis. *IEEE Transactions on Image Processing*, *21*(11), 4636–4648.

Zhuge, W., Hou, C., Jiao, Y., Yue, J., Tao, H., & Yi, D. (2017). Robust auto-weighted multi-view subspace clustering with common subspace representation matrix. *Plos One*, *12*(5), e0176769.