

Multi-view Discriminative Learning via Joint Non-negative Matrix Factorization

Zhong Zhang, Zhili Qin, Peiyan Li, Qinli Yang, and Junming Shao

Big Data Reserach Center, School of Computer Science and Engineering,
University of Electronic Science and Technology of China, Chengdu 611731, China

{zhongzhang,zhiliqin,peiyanli}@std.uestc.edu.cn,

{qinli.yang,junmshao}@uestc.edu.cn

<http://dm.uestc.edu.cn>

Abstract. Multi-view learning attempts to generate a classifier with a better performance by exploiting relationship among multiple views. Existing approaches often focus on learning the consistency and/or complementarity among different views. However, not all consistent or complementary information is useful for learning, instead, only class-specific discriminative information is essential. In this paper, we propose a new robust multi-view learning algorithm, called DICS, by exploring the Discriminative and non-discriminative Information existing in Common and view-Specific parts among different views via joint non-negative matrix factorization. The basic idea is to learn a latent common subspace and view-specific subspaces, and more importantly, discriminative and non-discriminative information from all subspaces are further extracted to support a better classification. Empirical extensive experiments on seven real-world data sets have demonstrated the effectiveness of DICS, and show its superiority over many state-of-the-art algorithms.

Keywords: Multi-view learning, Matrix factorization, Classification

1 Introduction

Many real-world entities are often represented with different views such as web pages [1, 33], multi-lingual news [2, 8, 16] and neuroimaging [22–24]. Consistency and complementarity, as the bridges to link all views together, are the two main assumptions in current multi-view learning [30]. The consistency assumption suggests that there is consistent information shared by all views [3, 18, 31]. Apparently, it is insufficient to exploit multi-view data using only consistent information since each view also contains complementary knowledge that other views do not have [1, 9, 19]. Therefore, investigating the complementarity of views is another important paradigm to learn multi-view data.

However, a question comes to our mind: whether the derived consistent and (or) complementary information really always support a better classification performance? Our answer is: *no*, since empirical pre-experiments indicate that prediction performance on multi-view data can be even worse than using single-view data in some real-world data sets. The main reason is that the consistent or

complementary information does not learn discriminative information directly. The classifier constructed by multi-view data may give an even worse classification performance if the learned consistent and (or) complementary information contains no clear discriminative information.

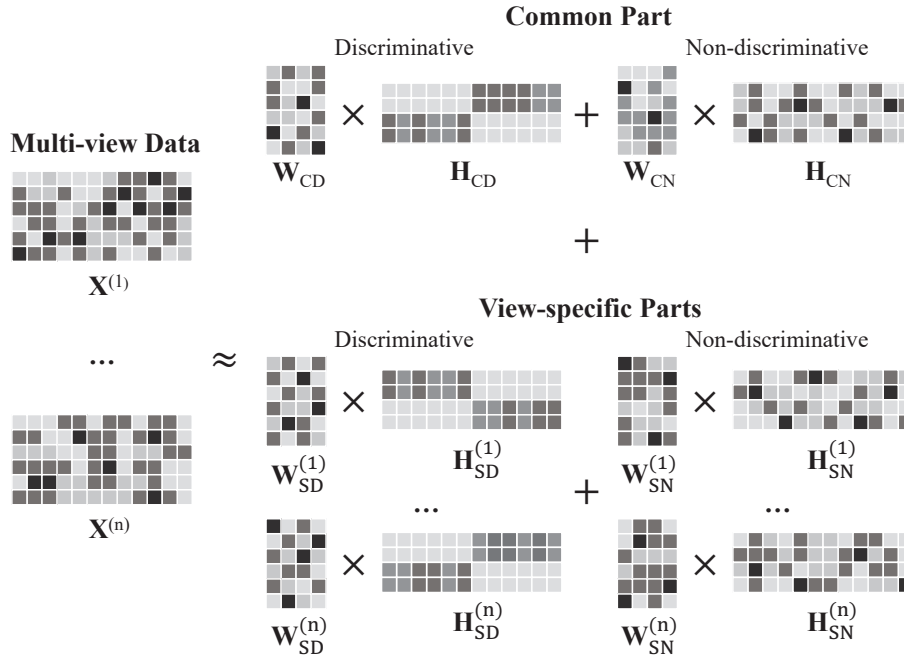


Fig. 1: Illustration of extracting discriminative information from multi-view data via joint non-negative matrix factorization. Each view of the data matrix is a superposition of four different parts: common discriminative part, common non-discriminative part, specific discriminative part and specific non-discriminative part.

In this paper, towards robust multi-view learning, we examine both discriminative and non-discriminative information existing in the consistent and complementary parts, and use only discriminative information for learning. Following this idea, we propose a new multi-view learning algorithm, called **DICS**, by exploring the **D**iscriminative and non-discriminative **I**nformation existing in **C**ommon and view-**S**pecific parts among different views via joint non-negative matrix factorization (NMF). Specifically, as usual, multi-view data is factorized into common part shared across views and view-specific parts existing within each view. Beyond, for both common part and each view-specific part, they are further factorized into two parts (discriminative part and non-discriminative part). To better obtain the discriminative parts, a supervised constraint is added to guide the joint NMF factorization. For illustration, Fig. 1 gives a simple exam-

ple to illustrate the decomposition. Here, each view of data is factorized into four parts: the common discriminative, common non-discriminative, specific discriminative and specific non-discriminative part, respectively. To find the optimal decomposition, we follow the block coordinate descent (BCD) framework [14] to solve the objective function of DICS. Finally, only the derived discriminative parts from common part and view-specific parts are used to construct a classifier. Experimental results show that DICS allows extracting discriminative information as well as discarding non-discriminative information effectively, and supports a gained classification performance, which outperforms many state-of-the-art algorithms on seven real-world data sets.

2 Related Work

The most simplest way to deal with multi-view data is to concatenate all feature vectors of different views into one single long feature vector. However, such method ignores the relationships among multiple views and may suffer from the curse of dimensionality. To present, many advanced multi-view learning algorithms have been proposed, which can be broadly categorized into two types: The first category aims to exploit the consistency, and the second one focuses on exploiting the complementarity among multiple views.

Studies in exploiting consistency generally seek a common representation on which all views have minimum disagreement. For instance, Canonical Correlation Analysis (CCA) related algorithms [3, 6, 11, 12, 26] project two or more views into latent subspaces by maximizing the correlations among projected views. Spectral methods [5, 16, 20, 29, 33] use weighted summation to merge graph Laplacian matrices from different views into one optimal graph for further clustering or embedding. Matrix factorization based methods [8, 18, 27] jointly factorize multi-view data into one common centroid representation by minimizing the overall reconstruction loss of different views. In addition, multiple kernel learning (MKL) [7] can also be considered as exploiting the consistency across different views, where each view is mapped into a new space (e.g. kernel Hilbert space) using kernel trick, and then combines all kernel matrices into one unified kernel by minimizing a pre-defined objective function.

Another paradigm of multi-view learning is to explicitly preserve complementary information of different views. Co-training style algorithms [1, 15, 28, 32] treat each view as complementarity. Generally speaking, it iteratively trains two classifiers on two different views, and each classifier generates its complementary information to help the other classifier to train in the next iteration. Beyond, the Co-EM algorithm [21] can be considered as a probabilistic version of co-training. Subspace related methods are also adopted to learn the complementarity. For instance, [9, 10, 13, 19, 25] learn one shared latent factor and view-specific latent factors to simultaneously capture the consistency and complementarity.

In summary, most existing multi-view learning algorithms mainly focus on learning consistency and complementarity from multi-view data. However, discriminative information existing in consistency and complementarity is not fully

investigated, which is actually the direct factor to dominate the learning performance.

3 The Proposed Method

3.1 Preliminaries

Given a non-negative matrix $\mathbf{X} \in \mathbb{R}_+^{m \times n}$, where each column represents a data point. NMF aims to approximately factorize the data matrix into two non-negative matrix $\mathbf{W} \in \mathbb{R}_+^{m \times k}$ and $\mathbf{H} \in \mathbb{R}_+^{n \times k}$, so that,

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{H}} & \|\mathbf{X} - \mathbf{W}\mathbf{H}^T\|_F^2 \\ \text{s.t. } & \mathbf{W}, \mathbf{H} \geq 0 \end{aligned} \quad (1)$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Note that the original data matrix is a linear combination of all column vectors in \mathbf{W} with weights of corresponding column vectors in \mathbf{H} . Therefore, \mathbf{W} and \mathbf{H} are often called the basis matrix and the coefficient matrix respectively.

For multi-view data, NMF-based approaches often take either \mathbf{W} or \mathbf{H} as a common factor. One of the representative formulation is as follows.

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{H}} & \sum_{v=1}^{n_v} \|\mathbf{X}^{(v)} - \mathbf{W}^{(v)}\mathbf{H}^T\|_F^2 + \Phi(\mathbf{W}, \mathbf{H}) \\ \text{s.t. } & \mathbf{W}, \mathbf{H} \geq 0 \end{aligned} \quad (2)$$

where n_v denotes the number of views, and $\mathbf{W}^{(v)}$ denotes the basis matrices corresponding to different views. \mathbf{H} denotes the common coefficient matrix shared across views, and $\Phi(\cdot)$ are some regularization terms on \mathbf{W} and \mathbf{H} . It assumes that different views of one identical object are generated from distinct subspaces, and all views share with one centroid latent representation. This paradigm considers the consistency shared by all views, however, it ignores the complementary knowledge existing in each view.

3.2 Discriminant Learning on Multi-view Data

As multiple views have their commonality and distinctiveness, we first decompose the multi-view data into two parts: common part and view-specific parts, like many existing approaches [9, 10, 13, 19]. Formally, let \mathbf{W}_C represents the common subspace shared by all views and $\mathbf{W}_S^{(v)}$ represents the distinct subspace corresponding to each specific view. Therefore, each view of data matrix can be written as $\mathbf{X}^{(v)} = \mathbf{W}_C\mathbf{H}_C^T + \mathbf{W}_S^{(v)}\mathbf{H}_S^{(v)T}$. To derive the common and view-specific information, we thus can formulate our objective function as follows.

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{H}} & \sum_{v=1}^{n_v} \left\| \mathbf{X}^{(v)} - \begin{bmatrix} \mathbf{W}_C & \mathbf{W}_S^{(v)} \end{bmatrix} \begin{bmatrix} \mathbf{H}_C^T \\ \mathbf{H}_S^{(v)T} \end{bmatrix} \right\|_F^2 + \Phi(\mathbf{W}, \mathbf{H}) \\ \text{s.t. } & \mathbf{W}, \mathbf{H} \geq 0 \end{aligned} \quad (3)$$

To learn the discriminative information existing in multi-view data, we further leverage the available label information to guide joint matrix factorization in a supervised way. Specifically, we first divide the common part and each view-specific part into the discriminative part and the non-discriminative part, respectively. Namely,

$$\widetilde{\mathbf{W}} = \begin{bmatrix} \mathbf{W}_{\text{CD}} & \mathbf{W}_{\text{CN}} & \mathbf{W}_{\text{SD}}^{(v)} & \mathbf{W}_{\text{SN}}^{(v)} \end{bmatrix} \quad (4)$$

$$\widetilde{\mathbf{H}} = \begin{bmatrix} \mathbf{H}_{\text{CD}} & \mathbf{H}_{\text{CN}} & \mathbf{H}_{\text{SD}}^{(v)} & \mathbf{H}_{\text{SN}}^{(v)} \end{bmatrix} \quad (5)$$

where \mathbf{W}_{CD} and \mathbf{W}_{CN} indicate the common discriminative as well as the non-discriminative part of matrix $\widetilde{\mathbf{W}}$, respectively. Similarly, $\mathbf{W}_{\text{SD}}^{(v)}$ and $\mathbf{W}_{\text{SN}}^{(v)}$ indicate the view-specific parts. It is the same for $\widetilde{\mathbf{H}}$.

Afterwards, we impose the supervised constraint on the latent coefficient matrix \mathbf{H} . Here, it is worth noting that we only add the constraint on the discriminative part of \mathbf{H} to derive discriminability. In addition, we should notice that the discriminative information not only exists in the common part, but also in each view-specific part. Therefore, the objective function is further reformulated as follows.

$$\begin{aligned} & \min_{\mathbf{W}, \mathbf{H}, \mathbf{B}} \sum_{v=1}^{n_v} \|\mathbf{X}^{(v)} - \widetilde{\mathbf{W}}\widetilde{\mathbf{H}}^T\|_F^2 + \Phi(\mathbf{W}, \mathbf{H}) \\ & + \gamma \left\| \mathbf{Y} - \begin{bmatrix} \mathbf{B}_{\text{CD}} & \mathbf{B}_{\text{SD}}^{(v)} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{\text{CD}}^T \\ \mathbf{H}_{\text{SD}}^{(v)T} \end{bmatrix} \right\|_F^2 \\ & \text{s.t. } \mathbf{W}, \mathbf{H} \geq 0, \|(\mathbf{W})_{:,i}\|_2 = 1 \end{aligned} \quad (6)$$

where $\mathbf{Y} \in \mathbb{R}^{c \times n}$ is the label matrix, c is the number of classes, and n is the number of data instances. $y_{i,j} = 1$ if the instance j belong to class i and 0 otherwise. $\mathbf{B} = [\mathbf{B}_{\text{CD}} \ \mathbf{B}_{\text{SD}}^{(v)}] \in \mathbb{R}^{c \times (k_1 + k_3)}$ is a linear projection matrix which maps the latent representation into label space. Subscript ‘‘C’’ and ‘‘S’’ represent ‘‘common’’ and ‘‘specific’’ respectively. ‘‘D’’ and ‘‘N’’ represent ‘‘discriminative’’ and ‘‘non-discriminative’’ respectively. For example, \mathbf{W}_{CD} denotes the common discriminative subspace. We normalize each column vector of \mathbf{W} to ensure a unique solution. The supervised regularization term is imposed on $\mathbf{H}_{\text{D}} = [\mathbf{H}_{\text{CD}} \ \mathbf{H}_{\text{SD}}^{(v)}]$ to make the derived patterns discriminative.

3.3 Regularization Terms

To further enhance the discriminative power of latent subspaces, we impose a $\ell_{1,1}$ norm constraint on \mathbf{W}_{D} as $\|\mathbf{W}_{\text{D}}^T \mathbf{W}_{\text{D}}\|_{1,1}$, where $\mathbf{W}_{\text{D}} = [\mathbf{W}_{\text{CD}} \ \mathbf{W}_{\text{SD}}^{(v)}]$. This term can be factorized into two parts: $\|\mathbf{W}_{\text{D}}^T \mathbf{W}_{\text{D}}\|_{1,1} = \sum_i \mathbf{w}_{\text{D}i}^T \mathbf{w}_{\text{D}i} + \sum_{i \neq j} \mathbf{w}_{\text{D}i}^T \mathbf{w}_{\text{D}j}$. The first term is used to prevent overfitting. The second term encourages basis vectors to be as orthogonal as possible, which reduces the redundancy of discriminative bases. At last, we impose a $\ell_{1,1}$ norm constraint on \mathbf{H}_{D} , which

encourages the discriminative coefficients to be sparse. The reason is that data points of different classes should not possess identical latent concepts (i.e. basis vectors). It is reasonable that a latent concept only appears in a certain class but not in the others. With such intuition, a discriminative latent representation vector \mathbf{h}_{D_i} should be sparse in the elements which are corresponding to the latent concepts that \mathbf{h}_{D_i} doesn't possess. Finally, putting all terms together, the objective function of DICS is formulated as follows.

$$\begin{aligned} & \min_{\mathbf{W}, \mathbf{H}, \mathbf{B}} \sum_{v=1}^{n_v} \left\| \mathbf{X}^{(v)} - \widetilde{\mathbf{W}} \widetilde{\mathbf{H}}^T \right\|_F^2 + \alpha \left\| \mathbf{W}_D^T \mathbf{W}_D \right\|_{1,1} \\ & + \beta \left\| \mathbf{H}_D \right\|_{1,1} + \gamma \left\| \mathbf{Y} - \begin{bmatrix} \mathbf{B}_{CD} & \mathbf{B}_{SD}^{(v)} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{CD}^T \\ \mathbf{H}_{SD}^{(v)T} \end{bmatrix} \right\|_F^2 \\ & \text{s.t. } \mathbf{W}, \mathbf{H} \geq 0, \|(\mathbf{W})_{\cdot, i}\|_2 = 1 \end{aligned} \quad (7)$$

where α, β, γ are non-negative parameters to balance the regularization terms.

3.4 Optimization

The objective function Eq. (7) is not convex over both variables \mathbf{W} and \mathbf{H} . Therefore, it is impractical to find the global optimum. We follow the general BCD framework to divide the objective function Eq. (7) into several convex subproblems corresponding to each column of \mathbf{W} and \mathbf{H} , then solve each subproblem successively by fixing the others. In this way, the global convergence and local minimum solutions can be obtained [4].

Firstly, we represent $\mathbf{W}\mathbf{H}^T$ as the sum of rank-1 outer products. We can equivalently reformulate the objective function Eq. (7) as follows.

$$\begin{aligned} f(\mathbf{W}, \mathbf{H}, \mathbf{B}) = & \sum_{v=1}^{n_v} \left\| \mathbf{X}^{(v)} - \sum_{i=1}^{k1} \mathbf{w}_{CDi} \mathbf{h}_{CDi}^T - \sum_{i=1}^{k2} \mathbf{w}_{CNi} \mathbf{h}_{CNi}^T - \right. \\ & \left. \sum_{i=1}^{k3} \mathbf{w}_{SDi} \mathbf{h}_{SDi}^{(v)T} - \sum_{i=1}^{k4} \mathbf{w}_{SNi} \mathbf{h}_{SNi}^{(v)T} \right\|_F^2 + \\ & \alpha \left(\sum_{i=1}^{k1} \sum_{j=1}^{k1} \mathbf{w}_{CDi}^T \mathbf{w}_{CDj} + \sum_{i=1}^{k3} \sum_{j=1}^{k3} \mathbf{w}_{SDi}^{(v)T} \mathbf{w}_{SDj}^{(v)} + \right. \\ & \left. 2 \sum_{i=1}^{k1} \sum_{j=1}^{k3} \mathbf{w}_{CDi}^T \mathbf{w}_{SDj}^{(v)} \right) + \beta \mathbf{1}_{1 \times n} \left(\sum_{i=1}^{k1} \mathbf{h}_{CDi} + \sum_{i=1}^{k3} \mathbf{h}_{SDi}^{(v)} \right) \\ & \gamma \left\| \mathbf{Y} - \sum_{i=1}^{k1} \mathbf{b}_{CDi} \mathbf{h}_{CDi}^T - \sum_{i=1}^{k3} \mathbf{b}_{SDi}^{(v)} \mathbf{h}_{SDi}^{(v)T} \right\|_F^2 \end{aligned} \quad (8)$$

where \mathbf{w}_{CDi} , \mathbf{w}_{CNi} , $\mathbf{w}_{SDi}^{(v)}$, $\mathbf{w}_{SNi}^{(v)}$, \mathbf{h}_{CDi} , \mathbf{h}_{CNi} , $\mathbf{h}_{SDi}^{(v)}$, $\mathbf{h}_{SNi}^{(v)}$ are the i -th column vectors of \mathbf{W}_{CD} , \mathbf{W}_{CN} , $\mathbf{W}_{SD}^{(v)}$, $\mathbf{W}_{SN}^{(v)}$, \mathbf{H}_{CD} , \mathbf{H}_{CN} , $\mathbf{H}_{SD}^{(v)}$, $\mathbf{H}_{SN}^{(v)}$ respectively. $\mathbf{1}_{1 \times n}$ is a row vector of length n with all elements 1.

By fixing all column vectors except the one we want to update, we can obtain the convex subproblem respect to it, then solve it based on the BCD framework. Note that we use $[\cdot]_+$ to denote $\max(0, \cdot)$, which projects the negative value to the boundary of feasible region of zero. Finally, we give the update rules as follows.

$$\mathbf{w}_{CDi} = \mathbf{w}_{CDi} + \left[\frac{\sum_{v=1}^{n_v} (\mathbf{R}^{(v)} \mathbf{h}_{CDi} - \alpha (\mathbf{W}_{CD} \mathbf{1}_{k1 \times 1} + \mathbf{W}_{SD}^{(v)} \mathbf{1}_{k3 \times 1}))}{n_v (\mathbf{h}_{CDi}^T \mathbf{h}_{CDi} + \alpha)} \right]_+ \quad (9)$$

$$\mathbf{w}_{CNi} = \mathbf{w}_{CNi} + \left[\frac{\sum_{v=1}^{n_v} \mathbf{R}^{(v)} \mathbf{h}_{CNi}}{n_v (\mathbf{h}_{CNi}^T \mathbf{h}_{CNi})} \right]_+ \quad (10)$$

$$\mathbf{w}_{SDi}^{(v)} = \mathbf{w}_{SDi}^{(v)} + \left[\frac{\mathbf{R}^{(v)} \mathbf{h}_{SDi}^{(v)} - \alpha (\mathbf{W}_{CD} \mathbf{1}_{k1 \times 1} + \mathbf{W}_{SD}^{(v)} \mathbf{1}_{k3 \times 1})}{\mathbf{h}_{SDi}^{(v)T} \mathbf{h}_{SDi}^{(v)} + \alpha} \right]_+ \quad (11)$$

$$\mathbf{w}_{SNi}^{(v)} = \mathbf{w}_{SNi}^{(v)} + \left[\frac{\mathbf{R}^{(v)} \mathbf{h}_{SNi}^{(v)}}{\mathbf{h}_{SNi}^{(v)T} \mathbf{h}_{SNi}^{(v)}} \right]_+ \quad (12)$$

$$\mathbf{h}_{CDi} = \mathbf{h}_{CDi} + \left[\frac{\sum_{v=1}^{n_v} (\mathbf{R}^{(v)T} \mathbf{w}_{CDi} - \frac{\beta}{2} \mathbf{1}_{n \times 1} + \gamma \mathbf{Q}^{(v)T} \mathbf{b}_{CDi})}{n_v (\mathbf{w}_{CDi}^T \mathbf{w}_{CDi} + \gamma \mathbf{b}_{CDi}^T \mathbf{b}_{CDi})} \right]_+ \quad (13)$$

$$\mathbf{h}_{CNi} = \mathbf{h}_{CNi} + \left[\frac{\sum_{v=1}^{n_v} \mathbf{R}^{(v)T} \mathbf{w}_{CNi}}{n_v (\mathbf{w}_{CNi}^T \mathbf{w}_{CNi})} \right]_+ \quad (14)$$

$$\mathbf{h}_{SDi}^{(v)} = \mathbf{h}_{SDi}^{(v)} + \left[\frac{\mathbf{R}^{(v)T} \mathbf{w}_{SDi}^{(v)} - \frac{\beta}{2} \mathbf{1}_{n \times 1} + \gamma \mathbf{Q}^{(v)T} \mathbf{b}_{SDi}^{(v)}}{\mathbf{w}_{SDi}^{(v)T} \mathbf{w}_{SDi}^{(v)} + \gamma \mathbf{b}_{SDi}^{(v)T} \mathbf{b}_{SDi}^{(v)}} \right]_+ \quad (15)$$

$$\mathbf{h}_{SNi}^{(v)} = \mathbf{h}_{SNi}^{(v)} + \left[\frac{\mathbf{R}^{(v)T} \mathbf{w}_{SNi}^{(v)}}{\mathbf{w}_{SNi}^{(v)T} \mathbf{w}_{SNi}^{(v)}} \right]_+ \quad (16)$$

where $\mathbf{R}^{(v)}$ and $\mathbf{Q}^{(v)}$ are

$$\mathbf{R}^{(v)} = \mathbf{X}^{(v)} - \mathbf{W}_{CD} \mathbf{H}_{CD}^T - \mathbf{W}_{CN} \mathbf{H}_{CN}^T - \mathbf{W}_{SD}^{(v)} \mathbf{H}_{SD}^{(v)T} - \mathbf{W}_{SN}^{(v)} \mathbf{H}_{SN}^{(v)T} \quad (17)$$

$$\mathbf{Q}^{(v)} = \mathbf{Y} - \mathbf{B}_{CD} \mathbf{H}_{CD}^T - \mathbf{B}_{SD}^{(v)} \mathbf{H}_{SD}^{(v)T} \quad (18)$$

Note that we extract the common factors $\mathbf{R}^{(v)}$ and $\mathbf{Q}^{(v)}$ from the equations just for saving the writing space. However, it is not efficient for implementation, since the computation orders i.e. $(\mathbf{W}\mathbf{H}^T)\mathbf{h}_i$ and $\mathbf{W}(\mathbf{H}^T\mathbf{h}_i)$ largely affect the computational complexity. The former takes $mn(k+1)$ multiply operations, the later takes $(m+n)k$ multiply operations. Obviously the later form is much more efficient in implementation.

Algorithm 1 DICS Algorithm

Input:

Multi-view data matrices $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(n_v)}$, label matrix \mathbf{Y} , parameters α, β, γ , number of latent factors $k1, k2, k3, k4$.

Output:

Basis matrices $\mathbf{W} = \{\mathbf{W}_{CD}, \mathbf{W}_{CN}, \mathbf{W}_{SD}^{(v)}, \mathbf{W}_{SN}^{(v)}\}$,
 Coefficient matrices $\mathbf{H} = \{\mathbf{H}_{CD}, \mathbf{H}_{CN}, \mathbf{H}_{SD}^{(v)}, \mathbf{H}_{SN}^{(v)}\}$,
 Projection matrices $\mathbf{B} = \{\mathbf{B}_{CD}, \mathbf{B}_{SD}^{(v)}\}$.

- 1: Initialize \mathbf{W} , \mathbf{H} , and \mathbf{B} .
 - 2: **repeat**
 - 3: Update each column of \mathbf{W}_{CD} using Eq. (9)
 - 4: Update each column of \mathbf{W}_{CN} using Eq. (10)
 - 5: **for** $v = 1$ to n_v **do**
 - 6: Update each column of $\mathbf{W}_{SD}^{(v)}$ using Eq. (11)
 - 7: **end for**
 - 8: **for** $v = 1$ to n_v **do**
 - 9: Update each column of $\mathbf{W}_{SN}^{(v)}$ using Eq. (12)
 - 10: **end for**
 - 11: Update each column of \mathbf{H}_{CD} using Eq. (13)
 - 12: Update each column of \mathbf{H}_{CN} using Eq. (14)
 - 13: **for** $v = 1$ to n_v **do**
 - 14: Update each column of $\mathbf{H}_{SD}^{(v)}$ using Eq. (15)
 - 15: **end for**
 - 16: **for** $v = 1$ to n_v **do**
 - 17: Update each column of $\mathbf{H}_{SN}^{(v)}$ using Eq. (16)
 - 18: **end for**
 - 19: Update \mathbf{B}_{CD} using Eq. (19)
 - 20: **for** $v = 1$ to n_v **do**
 - 21: Update $\mathbf{B}_{SD}^{(v)}$ using Eq. (20)
 - 22: **end for**
 - 23: **until** convergence or max no. iterations reached;
-

In addition, when the other variables are fixed, the projection matrices \mathbf{B}_{CD} and $\mathbf{B}_{SD}^{(v)}$ can be solved in a closed form as follows.

$$\mathbf{B}_{CD} = \frac{\sum_{v=1}^{n_v} (\mathbf{Y} - \mathbf{B}_{SD}^{(v)} \mathbf{H}_{SD}^{(v)T})}{n_v} \mathbf{H}_{CD} (\mathbf{H}_{CD}^T \mathbf{H}_{CD} + \lambda \mathbf{I})^{-1} \quad (19)$$

$$\mathbf{B}_{SD}^{(v)} = (\mathbf{Y} - \mathbf{B}_{CD} \mathbf{H}_{CD}^T) \mathbf{H}_{SD}^{(v)} (\mathbf{H}_{SD}^{(v)T} \mathbf{H}_{SD}^{(v)} + \lambda \mathbf{I})^{-1} \quad (20)$$

where \mathbf{I} is the identity matrix, λ is a small positive number.

Initialization. Since the NMF objective function is non-convex and has many local minima, a proper initialization is beneficial to improve learning performance. We develop a heuristic approach to initialize the basis matrix. DICS encourages the discriminative bases to achieve a degree of orthogonality, thus we try to initialize them as orthogonal as possible. To initialize \mathbf{W}_C , we first

calculate the mean of multi-view data, i.e. $\bar{\mathbf{X}} = \frac{1}{n_v} \sum_v \mathbf{X}^{(v)}$. Afterwards, we clustering $\bar{\mathbf{X}}$ into $k_1 + k_2$ clusters and obtain the corresponding centroids. Then we compute the pairwise linear correlation coefficients between each pair of centroids, and sort them in an ascending order. At last, we select k_1 centroids corresponding to the top k_1 correlation coefficients to initialize \mathbf{W}_{CD} , and use the rest k_2 centroids to initialize \mathbf{W}_{CN} . It is same to initialize each $\mathbf{W}_{\text{S}}^{(v)}$ by replacing $\bar{\mathbf{X}}$ with $\mathbf{X}^{(v)}$.

Time Complexity. The computational complexity of DICS is the same as solving standard NMF problem via hierarchical alternating least squares (HALS) algorithm under the BCD framework [14]. It is $O(\sum_v m_v nk)$ in the multi-view case, where m_v is the dimension of the v -view feature. Finally, the pseudocode of DICS is given in Algorithm 1.

4 Experiment

In this section, we first experimentally evaluate the proposed algorithm DICS in classification task on seven real world multi-view data sets. Then we empirically investigate that whether the extracted discriminative information from the common and the view-specific parts are really helpful for improving the learning performance. At last, the sensitivity of parameters and the convergence of DICS are analyzed.

4.1 Data Sets

Four popular real-world multi-view data sets are used in the experiment, including WebKB, Reuters, YaleFace and BBC, where the WebKB data set can be further divided into four sub data sets, namely Cornell, Texas, Washington, Wisconsin. Therefore, finally seven data sets are used to evaluate the performance of the proposed algorithm in this study. The statistics of data sets are summarized in Table 1.

Table 1: Statistics of the Data Sets

Data sets	Data size	# of views	# of classes	# of dimensions
Reuters ¹	1200	5	6	2000 for all
Cornell ¹	195	2	5	1703/585
Texas ¹	187	2	5	1703/561
Washington ¹	230	2	5	1703/690
Wisconsin ¹	265	2	5	1703/795
YaleFace ²	256	2	8	2016 for all
BBC ³	685	4	5	4659/4633/ 4665/4684

4.2 Selection of Comparison Algorithms

We compare DICS algorithm with several single-view and multi-view algorithms to demonstrate its effectiveness. For fair comparison, the source codes of all comparing algorithms are directly downloaded from the author’s website or requested from the author by email. The parameters of all algorithms are selected within the range that the author suggested, which are listed in the following. Also, the source code of our proposed DICS algorithm can be acquired from Dropbox⁴.

- *KNN*. We use the KNN algorithm (Set $k = 1$) as the baseline algorithm since all NMF-based algorithms can be regarded as a preprocessing before KNN. We apply KNN on all single views and report the best performance on the view. Also we apply the KNN algorithm on the concatenated feature vector (i.e. KNNcat).
- *NMF*. We apply the standard NMF algorithm on each of the single view data and the concatenated feature vector (i.e. NMFcat), as another baseline algorithm.
- *SSNMF*. This is a supervised NMF variant proposed in [17], which incorporates a linear classifier to encode the supervised information. We select the regularization parameter λ within the range of [0.5:0.5:3].
- *GNMF*⁵. This is a manifold regularized version of NMF [2], which preserves the local similarity by imposing a graph Laplacian regularization. We use the normalized dot product (cosine similarity) to construct the affinity graph, and select the regularization parameter λ within the set of $\{10^0, 10^1, 10^2, 10^3, 10^4\}$.
- *multiNMF*⁶. This is a well-known multi-view NMF algorithm proposed in [18]. We select the regularization parameter λ within the set of $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0\}$.
- *MVCC*⁷. MVCC incorporates the local manifold regularization for multi-view learning [27]. We set parameter α to 100, and select β and γ within the set of $\{50, 100, 200, 500, 1000\}$.
- *MCL*. This is a semi-supervised multi-view NMF variant with graph regularized constraint [8]. We select parameter α within the range of [100:50:250], β within the set of $\{0.01, 0.02, 0.03\}$, and set gamma to 0.005 as author suggested.
- *DICS*. This is the proposed algorithm. We select parameters: α , β and γ within the set of $\{10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$.

¹ <http://lig-membres.imag.fr/grimal/data.html>

² <http://vision.ucsd.edu/~iskwak/ExtYaleDatabase/ExtYaleB.html>

³ <http://mlg.ucd.ie/datasets/segment.html>

⁴ <https://www.dropbox.com/s/guohn1zhq073x9f/DICS.zip?dl=0>

⁵ <http://www.cad.zju.edu.cn/home/dengcai/Data/GNMF.html>

⁶ <http://jialu.cs.illinois.edu>

⁷ <https://github.com/vast-wang/Clustering.git>

4.3 Classification on Real-world Data sets

For DICS and all comparing algorithms, we first perform a five-folds cross validation to select the parameters, then we run ten times 10-folds cross validation with the selected parameters to obtain the final average classification accuracy and standard deviation. For all comparing NMF-based methods, we don't fix the number of latent factors k a global constant number, considering different algorithms may prefer different k s. Thus, we select k within the range of [5:5:100] for each algorithm. As for DICS, we need to set the number of four latent factors k_1, k_2, k_3, k_4 respectively. To avoid searching too large parameter space, we first select $k_i (i = 1, 2, 3, 4)$ within the range of [5:5:20], then we select the regularization parameters by fixing all k_i .

For classification, we first obtain latent representations from different NMF-based approaches, then we use KNN($k = 1$) for classification. Specifically, for unsupervised algorithms including NMF, GNMF, multiNMF, MCL and MVCC, we first apply algorithms on the data sets to obtain the latent representations \mathbf{H} , then we use \mathbf{H} for further training and testing. For supervised method like DICS, we first obtain the discriminative basis \mathbf{W}_D on training data, then we use the Moore-Penrose Pseudoinverse of \mathbf{W}_D as projection matrix to obtain new data representation, namely $\tilde{\mathbf{X}}^{(v)} = (\mathbf{W}_D^T \mathbf{W}_D)^{-1} \mathbf{W}_D^T \mathbf{X}^{(v)}$. Then we concatenate $\tilde{\mathbf{X}}^{(v)}$ as the input for KNN.

Table 2 summarizes the classification results of different multi-view learning algorithms, where the numbers in the parentheses of the table denote the standard deviation. The best result on each data set is highlighted in boldface. As we can see from the results, the proposed DICS outperforms the other comparison algorithms on all seven data sets. DICS is slightly better than other algorithms on Reuters, YaleFace and BBC. But it achieves remarkably promising performance on four WebKB sub data sets, where it outperforms the second best algorithm up to 9.01% on Texas especially. The amazing result may result from twofold: (a) DICS not only explores the common and the view-specific information, but more importantly, the discriminative information existing in these parts is further extracted, which thus supports a gained prediction performance. (b) By filtering out the non-discriminative information from common part and view-specific parts, and adding the supervised constraints on encoding coefficients, the extracted discriminative information is much more effective for classification.

4.4 Empirical Study of DICS Algorithm

DICS assumes that multi-view data can be decomposed into the common part and the view-specific parts, and only the discriminative information in them is essential. To verify this assumption, we first construct the following subspaces: $\mathbf{W}_D = [\mathbf{W}_{CD} \ \mathbf{W}_{SD}^{(v)}]$, $\mathbf{W}_N = [\mathbf{W}_{CN} \ \mathbf{W}_{SN}^{(v)}]$, $\mathbf{W}_C = [\mathbf{W}_{CD} \ \mathbf{W}_{CN}]$ and $\mathbf{W}_S = [\mathbf{W}_{SD}^{(v)} \ \mathbf{W}_{SN}^{(v)}]$, denoting as the ‘‘Discriminative’’, ‘‘Non-discriminative’’, ‘‘Common’’ and ‘‘Specific’’ subspace. Afterwards, we project the original data

Table 2: Multi-view Classification Performance on Real-world Data Sets

Method	ACC (%)						
	Reuters	Cornell	Texas	Washington	Wisconsin	YaleFace	BBC
KNN	43.3 (0.5)	61.6 (1.4)	66.4 (1.2)	74.4 (1.3)	60.2 (1.4)	89.1 (1.1)	36.7 (0.6)
KNNcat	37.3 (0.5)	62.5 (1.5)	65.5 (1.1)	70.4 (0.7)	61.7 (1.4)	74.9 (1.1)	22.7 (1.0)
NMF	59.2 (1.1)	64.0 (2.9)	72.8 (3.5)	74.5 (1.0)	74.9 (2.1)	93.3 (0.7)	69.8 (1.9)
NMFcat	59.6 (0.9)	65.9 (1.9)	73.7 (2.1)	75.3 (2.5)	77.1 (1.7)	92.5 (1.4)	87.9 (1.4)
SSNMF	64.0 (0.7)	66.5 (2.5)	69.3 (2.6)	73.4 (1.1)	73.9 (2.6)	93.8 (0.8)	81.6 (0.7)
GNMF	50.0 (1.0)	49.0 (2.0)	59.8 (1.8)	58.2 (2.0)	67.0 (1.8)	14.9 (1.3)	46.6 (1.7)
multiNMF	61.1 (0.8)	54.7 (1.4)	67.4 (2.8)	59.0 (2.8)	61.5 (2.8)	90.7 (1.2)	89.3 (1.4)
MCL	64.4 (0.8)	69.9 (2.2)	70.0 (2.5)	74.7 (2.1)	79.6 (2.4)	90.0 (0.2)	90.0 (0.8)
MVCC	55.0 (1.5)	64.9 (1.7)	71.0 (3.4)	70.8 (2.8)	76.4 (2.7)	29.5 (3.6)	70.2 (9.6)
DICS	66.9 (1.6)	75.5 (2.3)	82.7 (2.0)	78.7 (1.3)	84.3 (1.1)	94.1 (1.0)	91.9 (0.7)

onto these subspaces to obtain the corresponding components of data. We perform classification on each component, and the results are given in Fig. 2. The classification performance of the “Common” part is much worse than the “Specific” part, which suggests that only using the consistent information of multi-view data is not enough to capture the whole discriminative information. Also, performance on the “Discriminative” part is better than all other parts in all data sets except Reuters. It suggests that extracting the discriminative information from the common as well as the view-specific parts, and discarding the non-discriminative parts do help improve the learning performance.

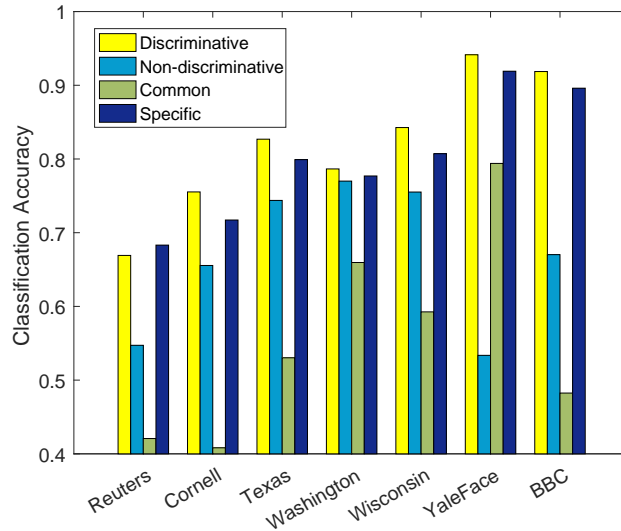


Fig. 2: Classification accuracy of DICS on different extracted components of multi-view data.

4.5 Parameter Study

There are three regularization parameters in DICS, i.e. α , β and γ . α controls the orthogonality degree of discriminative bases \mathbf{W}_D , β controls the degree of sparsity of discriminative latent representation \mathbf{H}_D , and γ balances the importance of supervised regularization term. To investigate how these parameters affect the final classification accuracy, we vary one parameter at a time within the set of $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$, and fix the others to 10^{-3} . Fig. 3 shows the variation trend of classification accuracy over different parameters on four typical data sets. The classification accuracy is relatively stable when α and β are less than 1, then drops sharply after α and β are increasing. As for parameter γ , the classification accuracy on BBC largely increases after γ is greater than 10^{-2} , and has become steady after γ is greater than 1. It is similar to other data sets except YaleFace, classification accuracy on YaleFace starts to decrease after γ is greater than 1. Based on the observation, we suggest selecting parameters α and β within a small range of $[0, 1]$, and simply set the parameter $\gamma = 1$ for practical use.

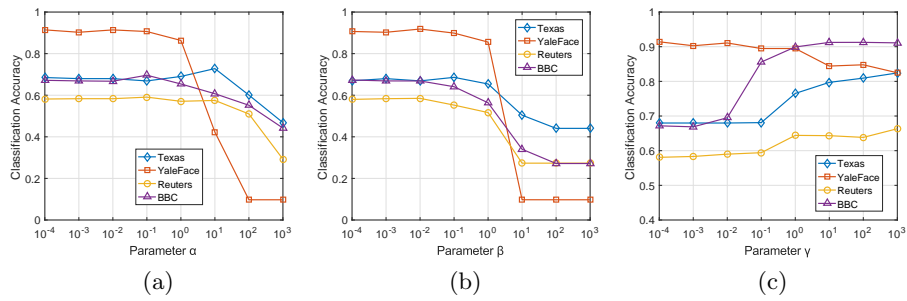


Fig. 3: Classification accuracy curve w.r.t. parameters α , β and γ .

4.6 Convergence Analysis

Though the original problem Eq. (7) is non-convex, the derived updating rules can achieve optimal minimum for each subproblem, the original problem Eq. (7) will eventually converge to a local minimum solution. In order to empirically investigate the convergence property of DICS, we plot the convergence curve and the corresponding classification accuracy curve on four typical data sets (see Fig. 4). From all four plots, we can observe that the objective values drop sharply and meanwhile the classification accuracies increase rapidly within about the first 10 iterations. After that, convergence curves and the accuracy curves begin to grow/decrease mildly, then it converges eventually. Usually, DICS will converge in no more than 50 iterations, while the corresponding classification accuracy becomes stable.

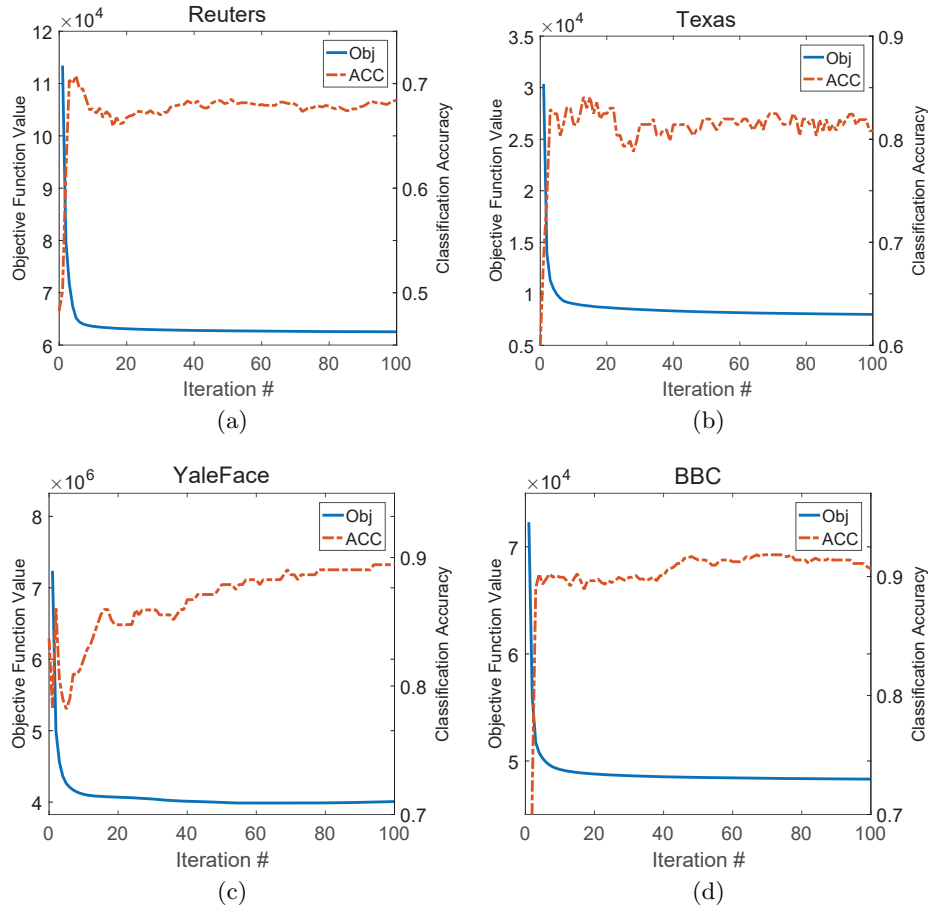


Fig. 4: Convergence and the corresponding classification accuracy curve of DICS on four typical data sets.

5 Conclusion

In this paper, we propose a novel multi-view learning algorithm, called DICS, by exploiting the discriminative information existing in multi-view data. To this end, a joint non-negative matrix factorization is employed to factorize multi-view data into a common part and view-specific parts. Beyond, the discriminative and non-discriminative information in these parts are further extracted in a supervised way. In contrast to existing multi-view learning approaches focusing on consistent and (or) complementary information, our new approach, offers an intuitive and effective way to improve classification performance based on the direct discriminative information. The high discriminative power of derived distinct patterns, further demonstrates the effectiveness of DICS on seven multi-view real-world data sets. Although DICS has several desirable properties, it has

its own drawbacks. One limitation is that tuning k_i in DICS is quite troublesome, since inferring the subspace dimensionality is still an open problem for all NMF-based algorithms. We simply tune k_i via model selection with traditional strategy. However, once we set proper k_i for each subspace, the promising results can be obtained as we have demonstrated.

6 Acknowledgments

This work is supported by the National Natural Science Foundation of China (61403062, 41601025, 61433014.), Science-Technology Foundation for Young Scientist of SiChuan Province (2016JQ0007), State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering (2017490211), National key research and development program (2016YFB0502300).

References

1. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: COLT. pp. 92–100 (1998)
2. Cai, D., He, X., Han, J., Huang, T.S.: Graph regularized nonnegative matrix factorization for data representation. TPAMI 33(8), 1548–1560 (2011)
3. Chaudhuri, K., Kakade, S.M., Livescu, K., Sridharan, K.: Multi-view clustering via canonical correlation analysis. In: ICML. pp. 129–136 (2009)
4. Chu, M., Diele, F., Plemmons, R., Ragni, S.: Optimality, computation, and interpretation of nonnegative matrix factorizations. SIMAX (2004)
5. De Sa, V.R.: Spectral clustering with two views. In: ICML workshop on learning with multiple views. pp. 20–27 (2005)
6. Farquhar, J.D., Hardoon, D.R., Meng, H., Shawe-Taylor, J., Szedmak, S.: Two view learning: Svm-2k, theory and practice. In: NIPS. pp. 355–362 (2005)
7. Gönen, M., Alpaydm, E.: Multiple kernel learning algorithms. JMLR 12(Jul), 2211–2268 (2011)
8. Guan, Z., Zhang, L., Peng, J., Fan, J.: Multi-view concept learning for data representation. TKDE 27(11), 3016–3028 (2015)
9. Gupta, S.K., Phung, D., Adams, B., Tran, T., Venkatesh, S.: Nonnegative shared subspace learning and its application to social media retrieval. In: KDD. pp. 1169–1178 (2010)
10. Gupta, S.K., Phung, D., Adams, B., Venkatesh, S.: Regularized nonnegative shared subspace learning. DMKD 26(1), 57–97 (2013)
11. Hardoon, D.R., Szedmak, S., Shawe-Taylor, J.: Canonical correlation analysis: An overview with application to learning methods. Neural Computation 16(12), 2639–2664 (2004)
12. Kan, M., Shan, S., Zhang, H., Lao, S., Chen, X.: Multi-view discriminant analysis. TPAMI 38(1), 188–194 (2016)
13. Kim, H., Choo, J., Kim, J., Reddy, C.K., Park, H.: Simultaneous discovery of common and discriminative topics via joint nonnegative matrix factorization. In: KDD. pp. 567–576 (2015)
14. Kim, J., He, Y., Park, H.: Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework. JGO 58(2), 285–319 (2014)

15. Kumar, A., Daumé, H.: A co-training approach for multi-view spectral clustering. In: ICML. pp. 393–400 (2011)
16. Kumar, A., Rai, P., Daume, H.: Co-regularized multi-view spectral clustering. In: NIPS. pp. 1413–1421 (2011)
17. Lee, H., Yoo, J., Choi, S.: Semi-supervised nonnegative matrix factorization. *IEEE Signal Processing Letters* 17(1), 4–7 (2010)
18. Liu, J., Wang, C., Gao, J., Han, J.: Multi-view clustering via joint nonnegative matrix factorization. In: SDM. pp. 252–260 (2013)
19. Liu, J., Jiang, Y., Li, Z., Zhou, Z.H., Lu, H.: Partially shared latent factor learning with multiview data. *TNNLS* 26(6), 1233–1246 (2015)
20. Nie, F., Li, J., Li, X.: Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification. In: IJCAI (2016)
21. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM. *Machine Learning* 39(2), 103–134 (2000)
22. Shao, J., Meng, C., Tahmasian, M., Brandl, F., Yang, Q., Luo, G., Luo, C., Yao, D., Gao, L., Riedl, V., et al.: Common and distinct changes of default mode and salience network in schizophrenia and major depression. *Brain Imaging and Behavior* pp. 1–12 (2018)
23. Shao, J., Myers, N., Yang, Q., Feng, J., Plant, C., Böhm, C., Förstl, H., Kurz, A., Zimmer, C., Meng, C., et al.: Prediction of alzheimer’s disease using individual structural connectivity networks. *Neurobiology of Aging* 33(12), 2756–2765 (2012)
24. Shao, J., Yang, Q., Wohlschlaeger, A., Sorg, C.: Discovering aberrant patterns of human connectome in alzheimer’s disease via subgraph mining. In: ICDMW. pp. 86–93 (2012)
25. Shao, J., Yu, Z., Li, P., Han, W., Sorg, C., Yang, Q.: Exploring common and distinct structural connectivity patterns between schizophrenia and major depression via cluster-driven nonnegative matrix factorization. In: ICDM (2017)
26. Sharma, A., Kumar, A., Daume, H., Jacobs, D.W.: Generalized multi-view analysis: A discriminative latent space. In: CVPR. pp. 2160–2167 (2012)
27. Wang, H., Yang, Y., Li, T.: Multi-view clustering via concept factorization with local manifold regularization. In: ICDM. pp. 1245–1250 (2016)
28. Wang, W., Zhou, Z.H.: A new analysis of co-training. In: ICML. pp. 1135–1142 (2010)
29. Xia, T., Tao, D., Mei, T., Zhang, Y.: Multiview spectral embedding. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 40(6), 1438–1446 (2010)
30. Xu, C., Tao, D., Xu, C.: A survey on multi-view learning. *arXiv preprint arXiv:1304.5634* (2013)
31. Ye, H.J., Zhan, D.C., Miao, Y., Jiang, Y., Zhou, Z.H.: Rank consistency based multi-view learning: a privacy-preserving approach. In: CIKM. pp. 991–1000 (2015)
32. Zhang, M.L., Zhou, Z.H.: Cotrade: confident co-training with data editing. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41(6), 1612–1626 (2011)
33. Zhou, D., Burges, C.J.: Spectral clustering and transductive learning with multiple views. In: ICML. pp. 1159–1166 (2007)