

Weighted Multi-View Spectral Clustering Based on Spectral Perturbation

Linlin Zong, Xianchao Zhang, Xinyue Liu, Hong Yu

School of Software, Dalian University of Technology, Dalian, China

Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian, China

llzong@dlut.edu.cn, xczhang@dlut.edu.cn, xyliu@dlut.edu.cn, hongyu@dlut.edu.cn

Abstract

Considering the diversity of the views, assigning the multi-views with different weights is important to multi-view clustering. Several multi-view clustering algorithms have been proposed to assign different weights to the views. However, the existing weighting schemes do not simultaneously consider the characteristic of multi-view clustering and the characteristic of related single-view clustering. In this paper, based on the spectral perturbation theory of spectral clustering, we propose a weighted multi-view spectral clustering algorithm which employs the spectral perturbation to model the weights of the views. The proposed weighting scheme follows the two basic principles: 1) the clustering results on each view should be close to the consensus clustering result, and 2) views with similar clustering results should be assigned similar weights. According to spectral perturbation theory, the largest canonical angle is used to measure the difference between spectral clustering results. In this way, the weighting scheme can be formulated into a standard quadratic programming problem. Experimental results demonstrate the superiority of the proposed algorithm.

Introduction

Multi-view clustering (Zhou and Burges 2007) (Kumar, Rai, and Daume 2011) (Lee and Liu 2016), which exploits complementary information among multiple views, has been a hot topic since the past decade. Considering the diversity of the views, the ability of different views characterizing the data is different. It is challenging to find a suitable way of simultaneously exploiting the complementary information of all the views in order to derive a satisfactory partition (Tzortzis and Likas 2012).

Recently, several clustering algorithms combined the multiple views through weighted combination and used the weights to characterize the importance of the views. The work in (Tzortzis and Likas 2012) (Cai, Nie, and Huang 2013) (Guo et al. 2014) (Zhao, Ding, and Fu 2017) used algebraic method to minimize the objective function of all the views, however, the minimum objective function value does not indicate the optimal clustering result. The work in (Huang, Chuang, and Chen 2012a) (Li et al. 2016) (Liu et al. 2016) (Zong et al. 2016) aggregated multiple affinity

matrices into a single one. These algorithms are under the framework of multi-kernel learning which is different from (though related with) multi-view clustering. Anyway, the existing weighting schemes do not simultaneously consider the characteristic of multi-view clustering and the characteristic of related single-view clustering.

In this paper, we focus on one specific clustering task, i.e., multi-view spectral clustering which has a consensus Laplacian matrix among all the views. For spectral clustering, small perturbations in the entries of the Laplacian matrix may lead to large perturbations in the eigenvectors. The Euclidean distance between two sets of eigenvectors can be large while the subspaces spanned by the eigenvectors is preserved (Hunter and Strohmer 2010). In this case, the difference in clustering results is not captured by the Euclidean distance between the eigenvectors but by the closeness of the subspaces spanned by the eigenvectors. According to the spectral perturbation theory, the closeness of the subspaces spanned by the eigenvectors is defined by the largest canonical angle between these subspaces.

Considering the spectral perturbation of spectral clustering, we propose a Weighted Multi-view Spectral Clustering algorithm (WMSC) based on spectral perturbation theory. We model the differences among multiple views following two basic principles: 1) the clustering results on each view should be close to the consensus clustering result, and 2) views with similar clustering results should be assigned similar weights. When realizing the two principles, the Laplacian matrix of each view is regarded as a perturbation of the consensus Laplacian matrix and the consensus Laplacian matrix is approximated by the weighted combination of multiple Laplacian matrices, then we use the largest canonical angle to measure the difference between spectral clustering results. Thus, the first principle is realized by minimizing the largest canonical angle between the subspace spanned by each view's eigenvectors and the one spanned by consensus eigenvectors; the second principle is realized by using a smoothness function to make the difference in weight proportional to the largest canonical angle. In this way, the weighting scheme is formulated into a standard quadratic programming problem. Experimental results show that the proposed algorithm outperforms typical unweighted multi-view spectral clustering algorithms and weighted multi-view spectral clustering algorithms.

Insights on Spectral Clustering

Spectral Clustering

Spectral clustering partitions data points into groups according to their similarities. Different from traditional clustering methods such as k -means, spectral clustering does not require the data space to be linearly separable and can detect non-convex patterns.

Given a set of data points $\{x_1, x_2, \dots, x_n\}$ belonging to k clusters, Ng et al. (Ng, Jordan, and Weiss 2002) proposed a normalized multi-class spectral clustering algorithm. The brief flow of the algorithm is summarized as follows.

1. Construct the similarity matrix S ;
2. Compute the normalized Laplacian matrix $L = D^{-1/2}SD^{-1/2}$, where D is a diagonal matrix and $D_{ii} = \sum_{j=1}^n S_{ij}$;
3. Compute the first k eigenvectors of L and construct a matrix with the eigenvectors as column;
4. Normalize the rows of the matrix to norm 1;
5. Cluster the normalized matrix with k -means.

For binary classification, the clusters are separated according to the sign of the second eigenvector's elements. Specifically, let $l(x_i)$ be the label of x_i , e be the second eigenvector, $l(x_i) = \text{sign}(e_i)$, where $\text{sign}(e_i) = 1$ if $e_i > 0$ and $\text{sign}(e_i) = -1$ if $e_i \leq 0$.

The key issue of spectral clustering is the construction of similarity matrix. For a comprehensive discussion of theoretical and practical aspects of spectral clustering, see (Luxburg 2007).

Spectral Perturbation

Small perturbations in the entries of the Laplacian matrix may result in large perturbations in the eigenvectors. Taking binary-class for example, we show the influence of matrix perturbation on the eigenvectors. Given 30 points belonging to two clusters, Figure 1 (Hunter and Strohmer 2010) presents the ground truth Laplacian matrix, the perturbed Laplacian matrix and their eigen-structures. Both the Laplacian matrices are effective for clustering, however, it can be seen that the second eigenvectors vary greatly (the sign is completely opposite).

The result of spectral clustering is determined by the first k eigenvectors of Laplacian matrix. If two sets of eigenvectors have a small difference in cluster ability, their clustering results are similar. It is easy to find a case (e.g., the example in Figure 1) that the Euclidean distance between the eigenvectors is large, whereas the clustering results are nearly the same. Thus it is not appropriate to define the difference in clustering results using the Euclidean distance between the eigenvectors, i.e., the Euclidean distance cannot reflect the difference in cluster ability between two set of eigenvectors. Fortunately, the subspaces spanned by the eigenvectors is nearly the same if the clustering results are similar. The difference in clustering results can be captured by the closeness of the subspaces spanned by the eigenvectors. In (Hunter and Strohmer 2010), the closeness of the subspaces spanned by the eigenvectors is measured by the canonical angle between

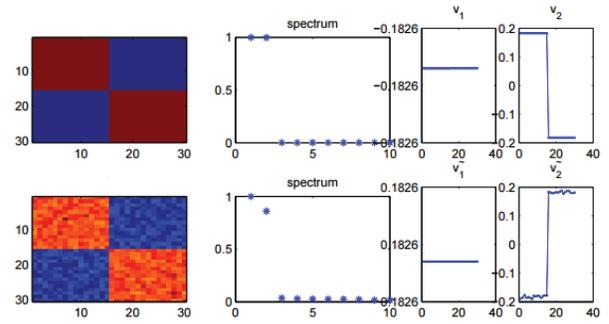


Figure 1: Example in (Hunter and Strohmer 2010). 30 data points with two underlying clusters. Top (from left to right): the ground truth Laplacian matrix L , the first ten eigenvalues, the first eigenvector v_1 and the second eigenvector v_2 . Bottom (from left to right): the perturbed Laplacian matrix \tilde{L} , the first ten eigenvalues, the first eigenvector \tilde{v}_1 and the second eigenvector \tilde{v}_2 .

these subspaces. Then, we could use the canonical angle to capture the difference in cluster ability between two set of eigenvectors.

Definition 1 (Hunter and Strohmer 2010) Let V_k and \tilde{V}_k be subspaces spanned by the orthogonal eigenvectors v_1, \dots, v_{i+k} and $\tilde{v}_1, \dots, \tilde{v}_{i+k}$. And let $\gamma_1 \leq \dots \leq \gamma_k$ be the singular values of $[v_1, \dots, v_{i+k}]^T [\tilde{v}_1, \dots, \tilde{v}_{i+k}]$. Then the values,

$$\theta_j = \arccos \gamma_j \quad (1)$$

are called the canonical angles between V_k and \tilde{V}_k .

Define V_k and \tilde{V}_k to be close if the largest canonical angle is small (Hunter and Strohmer 2010). That is, a smaller value of the largest canonical angle is, the more similar the cluster ability is. In Figure 1, the largest canonical angle between $\{v_2\}$ and $\{\tilde{v}_2\}$ is as small as 0.0199 which indicates similar cluster ability of $\{v_2\}$ and $\{\tilde{v}_2\}$.

For a detailed introduction of canonical angle, one can see (Davis 1970) (Stewart and Sun 1990).

Weighted Multi-view Spectral Clustering

This paper aims to quantify the importance of each view based on the characteristic of multi-view spectral clustering. In the following, we firstly introduce a multi-view spectral clustering method. Then, we propose a weighting scheme based on spectral perturbation theory.

Multi-view Spectral Clustering

In multi-view clustering, the same object represented in different views is expected to be in the same cluster. Thus, for the set of completely mapped multi-view data points, the ground truth similarity matrix in each view should be the same. In other words, there is a consensus ground truth Laplacian matrix among all the views. Usually, the consensus Laplacian matrix is unknown but can be approximated by the weighted combination of each view's Laplacian matrix.

Given the dataset $X = \{X^{(1)}, X^{(2)}, \dots, X^{(n_v)}\}$ with n_v views, where $X^{(a)} = \{x_1^{(a)}, x_2^{(a)}, \dots, x_n^{(a)}\}$, $a \in \{1, 2, \dots, n_v\}$, n is the number of data points, $L^{(a)} \in R^{n \times n}$ is the Laplacian matrix of the a -th view, $L^* \in R^{n \times n}$ is the consensus Laplacian matrix. The objective function of calculating L^* is shown as follows:

$$L^* = \sum_{a=1}^{n_v} \mu_a L^{(a)} \quad s.t. \sum_{a=1}^{n_v} \mu_a = 1, \mu_a \geq 0 \quad (2)$$

where μ_a is the weight of the a -th view. Then, L^* can be applied to spectral clustering.

In previous work such as (Zhou and Burges 2007), all the views are equally treated, i.e., $\mu_a = 1/n_v, a = \{1, 2, \dots, n_v\}$. However, the multiple views may have different effects on the final clustering result since they are complementary and diverse. It is rational to discriminate different views, and the weighting scheme is of great importance.

Weighting Scheme

In an ideal multi-view clustering, the clustering result on each view is the same with the consensus clustering result. In practical settings, views are distinguished but the views with similar clustering results are of similar importance. Thus, the weighting scheme should satisfy the following two principles: 1) the clustering results on each view should be close to the consensus clustering result, and 2) views with similar clustering results should be assigned similar weights. The first principle presents the basic assumption of multi-view clustering and the second principle presents the basic rule of weighting. The two principles describe the basic rules of weighted multi-view clustering.

Difference Minimization In spectral clustering, the clustering result is determined by the first k eigenvectors of the Laplacian matrix. Let $\{v_1^{(a)}, \dots, v_k^{(a)}\}$ and $\{v_1^*, \dots, v_k^*\}$ be the first k eigenvectors of $L^{(a)}$ and L^* respectively, to make the clustering result on the a -th view close to the consensus clustering result, the difference in cluster ability between $\{v_1^{(a)}, \dots, v_k^{(a)}\}$ and $\{v_1^*, \dots, v_k^*\}$ should be minimized.

Now the key issue is how to measure the difference in cluster ability between two set of eigenvectors. For all the views, the practical Laplacian matrices may be different but indicate the closeness of each pair of points, thus, the Laplacian matrix of each view is a perturbation of the consensus Laplacian matrix. According to spectral perturbation theory, the largest canonical angle indicates the similarity of cluster ability. To make the clustering results on each view close to the consensus clustering result, the largest canonical angle between the subspaces spanned by $\{v_1^{(a)}, \dots, v_k^{(a)}\}$ and $\{v_1^*, \dots, v_k^*\}$ should be minimized.

Here, we use a theorem of canonical angle.

Theorem 1 (Hunter and Strohmer 2010) Let $\lambda_i^{(a)}, v_i^{(a)}, \lambda_i^*, v_i^*$ be the i -th eigenvalue and eigenvector of $L^{(a)}$ and L^* respectively, $\lambda_1^{(a)} \geq \lambda_2^{(a)} \geq \dots \geq \lambda_n^{(a)}, \lambda_1^* \geq \lambda_2^* \geq \dots \geq \lambda_n^*$,

and let $\Theta = \text{diag}(\theta_1, \dots, \theta_k)$ be the canonical angles between the column space of $V_k^{(a)} = [v_1^{(a)}, \dots, v_k^{(a)}]$ and $V_k^* = [v_1^*, \dots, v_k^*]$. If there is a gap $\delta > 0$, such that

$$|\lambda_k^{(a)} - \lambda_{k+1}^*| \geq \delta; \quad \lambda_k^{(a)} \geq \delta$$

Then

$$\|\sin \Theta\|_F \ll 1/\delta \|L^* V_k^{(a)} - V_k^{(a)} \Sigma_k^{(a)}\|_F$$

Where $\Sigma_k^{(a)} = \text{diag}(\lambda_1^{(a)}, \dots, \lambda_k^{(a)})$ and $\sin \Theta$ is taken entrywise.

Inspired by Theorem 1, to minimize the largest canonical angle between two subspaces, the upper bound of $\sin \Theta$ should be minimized. Considering the fact that the rank of the semi-definite ground truth Laplacian matrix is equal to the number of clusters k , the ground truth Laplacian matrix has k positive eigenvalues and $n - k$ zero eigenvalues. Since L^* approximates the ground truth Laplacian matrix and the rank of L^* is expected to be k , $\lambda_{k+1}^* \approx 0$, $|\lambda_k^{(a)} - \lambda_{k+1}^*| \approx \lambda_k^{(a)}$. Given the Laplacian matrix of each view, δ can be seen as a constant. Therefore, the formulation to make the clustering result on the a -th view close to the consensus clustering result can be reduced to Eq.(3), where μ_a is the weight of the a -th view.

$$\begin{aligned} \min_{\mu} \sum_{a=1}^{n_v} \|L^* V_k^{(a)} - V_k^{(a)} \Sigma_k^{(a)}\|_F^2 \\ s.t. L^* = \sum_{a=1}^{n_v} \mu_a L^{(a)}, \sum_{a=1}^{n_v} \mu_a = 1, \mu_a \geq 0 \end{aligned} \quad (3)$$

Cluster Ability Smoothness If the spectral clustering results on two views are almost the same, i.e., the cluster ability of corresponding eigenvectors are almost the same, the weights of the two views should have little difference. As shown in the previous section, the largest canonical angle between subspaces spanned by the eigenvectors indicates the similarity of the cluster ability, i.e., the smaller the largest canonical angle is, the more similar the cluster ability is. Thus, the difference in weight between the views should be small if the largest canonical angle between corresponding subspaces is small.

Mathematically, let $C_{ab} \in [0, \pi]$ be the largest canonical angle between subspaces of the a -th view and the b -th view, $R_{ab} = \pi - C_{ab}$, we try to optimize Eq.(4),

$$\min_{\mu} \frac{1}{2} \sum_{ab} R_{ab} (\mu_a - \mu_b)^2 = \min_{\mu} \mu^T Q \mu \quad (4)$$

where $\mu = [\mu_1, \mu_2, \dots, \mu_{n_v}]^T$, μ_a is the weight of the a -th view, $Q = P - R$, $P = \text{diag}(p_1, \dots, p_{n_v})$, $p_a = \sum_{b=1}^{n_v} R_{ab}$.

Overall Objective Combining the above two aspects, the overall objective function of the proposed weighting scheme is formulated as Eq.(5),

$$\begin{aligned} \min_{\mu} \sum_{a=1}^{n_v} \|L^* V_k^{(a)} - V_k^{(a)} \Sigma_k^{(a)}\|_F^2 + \eta \mu^T Q \mu + \beta \mu^2 \\ s.t. L^* = \sum_{a=1}^{n_v} \mu_a L^{(a)}, \sum_{a=1}^{n_v} \mu_a = 1, \mu_a \geq 0 \end{aligned} \quad (5)$$

The last part of Eq.(5) regularizes the weight of each view, which ensures that each view contributes to the consensus Laplacian matrix. η and β respectively show the importance of the last two parts. Given η and β , in Eq.(5), the only variable is the weight of each view. Furthermore, Eq.(5) is turned to Eq.(6) by dropping the terms irrelevant to μ ,

$$\begin{aligned} & \mu^T \left(\sum_{a=1}^{n_v} T^{(a)} + \beta I + \eta Q \right) \mu - 2\mu^T \left(\sum_{a=1}^{n_v} Y^{(a)} \right) \\ \text{s.t. } & T_{ij}^{(a)} = \text{tr}(L^{(i)} V_k^{(a)} V_k^{(a)T} L^{(j)T}), i, j \in \{1, \dots, n_v\} \\ & Y_i^{(a)} = \text{tr}(L^{(i)} V_k^{(a)} \Sigma_k^{(a)T} V_k^{(a)T}), i \in \{1, \dots, n_v\} \\ & \sum_{a=1}^{n_v} \mu_a = 1, \mu_a \geq 0 \end{aligned} \quad (6)$$

where $Y^{(a)}$ is n_v dimensional and $T^{(a)} \in R^{n_v \times n_v}$.

It is easy to see that Eq.(6) is a standard quadratic programming problem with respect to μ and can be solved by classical techniques, e.g. the tool *quadprog* in Matlab.

The Algorithm

Summarizing the former analysis, we give the algorithm framework in Algorithm 1. Compared with single view spectral clustering which clusters each view separately, WMSC adds the step of calculating n_v -dimensional μ using Eq.(6). The computational complexity of constructing Eq.(6) (including construct $T^{(a)}$, $Y^{(a)}$ and Q) is $O(n_v^3 n^2 k)$. The computational complexity of solving Eq.(6) is $O(n_v^3)$. Generally, since $n_v \ll n$ and $k \ll n$, calculating μ does not increase the computational complexity of spectral clustering ($O(n^3)$). For large-scale data, the large-scale spectral clustering methods, e.g. sampling based methods (Zhang et al. 2016) (Li et al. 2015), can be applied to speed up WMSC.

Algorithm 1 WMSC algorithm

Require: $X = \{X^{(1)}, X^{(2)}, \dots, X^{(n_v)}\}$: the multi-view dataset

Ensure: *label*: the labels of each data

- 1: Calculate the normalized Laplacian matrix $L^{(a)}$ of the a -th view, $a \in \{1, 2, \dots, n_v\}$;
 - 2: Calculate the eigendecomposition of $L^{(a)}$ to obtain $V^{(a)}$ and $\Sigma_k^{(a)}$, $a \in \{1, 2, \dots, n_v\}$;
 - 3: Calculate μ according to Eq.(6);
 - 4: Calculate L^* according to Eq.(2);
 - 5: Run spectral clustering on L^* and get *label* of X .
-

Experiment

Datasets

We experiment on four benchmark datasets: ThreeSources¹, UCI Handwritten digits data², Flickr (Liu et al. 2015) and

¹<http://mlg.ucd.ie/datasets/3sources.html>

²<https://archive.ics.uci.edu/ml/datasets/Multiple+Features>

Cornell³. ThreeSources contains 169 news in six topics. The news are collected from three well-known online news sources. Each source is treated as one view. UCI Handwritten digits data consists of 2000 examples represented in three views: 76 Fourier coefficients of the character shapes, 240 pixel averages in 2 x 3 windows and 47 Zernike moments. Flickr contains 1028 images represented in two views: image-tag view and image-user view. The images are divided into eight categories. Cornell is composed of 226 web pages collected from Cornell University. A web page is made of two views: the text on it and the anchor text on the hyperlinks pointing to it.

The statistics of the datasets are summarized in Table 1.

Table 1: Statistics of the datasets.

Dataset	# instance	# view	# cluster
ThreeSources	169	3	6
Handwritten	2000	3	10
Flickr	1028	2	8
Cornell	226	2	4

Baselines

We compare the performance of the following methods.

- Single view spectral clustering: SC (Ng, Jordan, and Weiss 2002). We run spectral clustering and report the clustering result on each view. The clustering result on the i -th view is referred to as SC(i).
- Spectral clustering on data with concatenated features of all the views: ConcatSC.
- Unweighted multi-view spectral clustering: Coregspectral (Kumar, Rai, and Daume 2011), RMSC (Xia et al. 2014) and MVSC (Zhou and Burges 2007).
- Weighted multi-view spectral clustering (or kernel k-means): MVSpec (Tzortzis and Likas 2012), MKSC (Guo et al. 2014), AASC (Huang, Chuang, and Chen 2012a), LMKC (Li et al. 2016) and the proposed method WMSC.

Settings

Two common evaluation metrics, accuracy (ACC) and normalized mutual information (NMI) (Xu, Liu, and Gong 2003), are used to evaluate the performance of the proposed algorithm and the baselines. To avoid the randomness, we run all the algorithms 30 times and report their average values.

All methods use Gaussian kernel to calculate the similarity matrix where the scale parameter is set as the median of the pairwise Euclidean distances between the data points (Kumar, Rai, and Daume 2011).

For the baselines, we set the parameters by the grids suggested in the original papers. In our weighting scheme, two parameters β and η need to be set. To balance every part, we

³<http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/>

Table 2: ACC on all the datasets.

Algorithm\Dataset	ThreeSources	Handwritten	Flickr	Cornell
SC(1)	0.562±0.000	0.711±0.000	0.404±0.020	0.575±0.000
SC(2)	0.576±0.005	0.668±0.009	0.375±0.010	0.677±0.000
SC(3)	0.514±0.020	0.566±0.000	–	–
SC-cat	0.562±0.000	0.558±0.003	0.424±0.040	0.575±0.000
Coregspectral	0.596±0.031	0.822±0.000	0.451±0.009	0.734±0.073
RMSC	0.485±0.000	0.798±0.002	0.447±0.005	0.597±0.049
MVSC	0.525±0.042	0.776±0.015	0.415±0.014	0.586±0.035
MVSpec	0.592±0.000	0.768±0.000	0.398±0.000	0.743±0.000
MKSC	0.601±0.008	0.696±0.000	0.426±0.009	0.747±0.002
AASC	0.580±0.010	0.720±0.050	0.408±0.013	0.649±0.066
LMKC	0.568±0.000	0.868±0.000	0.456±0.000	0.615±0.000
WMSC	0.630±0.014	0.871±0.000	0.462±0.012	0.779±0.000

initially set the parameters by Eq.(7). Then, $\tilde{\beta}$ and $\tilde{\eta}$ are set by the grid $\{0.02, 0.04, \dots, 0.18, 0.2\}$.

$$\beta = \tilde{\beta}_* \frac{\|\sum_v T^{(v)} + Q\|_F}{\|I\|_F}; \eta = \tilde{\eta}_* \frac{\|\sum_v T^{(v)} + I\|_F}{\|Q\|_F} \quad (7)$$

Clustering Results

The clustering results in terms of ACC and NMI are reported in Table 2 ~ Table 3. In each column of the two tables, the best result is highlighted in boldface. From the results, the following points can be observed.

Firstly, on each multi-view dataset, the results of single view spectral clustering vary with the views. For example, on the ThreeSources dataset, the difference in ACC between the best view and the worst view is 6.2%, and the difference in NMI between the best view and the worst view is 6.9%. In particular, on the Cornell dataset, the difference in ACC between the best view and the worst view is 10.2%, but the difference in NMI between the best view and the worst view is 37.6%. The possible reason why ACC and NMI have such a big difference is that the cluster structure of Cornell is unbalanced. According to the definition of ACC and NMI, ACC is big as long as most points in the same cluster are assigned to the large cluster, however, NMI will be low if the other points are assigned arbitrarily. The above phenomenon conforms our observation that the views are diverse. Moreover, WMSC performs better than the single view results, which indicates that the multiple views are complementary. It is reasonable to ensemble multiple views.

Secondly, WMSC outperforms the unweighted multi-view spectral clustering algorithms. WMSC outperforms the unweighted methods in terms of both ACC and NMI on each dataset. Taking ACC for example, the ACC of WMSC raises 3% on the ThreeSources dataset, 5% on the Handwritten dataset, 1% on the Flickr dataset, 4% on the Cornell dataset. It can be concluded that it is rational to discriminate the views using spectral perturbation theory.

Finally, the other weighted methods do not show apparent superiorities over the unweighted ones. For instance, on average, the NMI of Coregspectral is greater than that of LMKC. It is because that these weighting schemes do

not consider the spectral perturbation of spectral clustering. This indicates that the other weighting schemes are not very effective. WMSC not only performs better than the unweighted methods, but also performs better than other weighted methods in terms of both ACC and NMI on all the datasets.

In summary, it can be concluded that WMSC generally performs the best among the competitive methods for clustering multi-view data.

Weight Analysis

For each weighted multi-view clustering algorithm, we report the weight of each view in Table 4.

For the algorithms MVSpec, MKSC and AASC, we find that the weight of each view is nearly the same on most datasets and the weights are inconsistent with the clustering results on the other datasets. For example, for the Flickr dataset, the first view performs better than the second view, but the weight of the first view is smaller than that of the second view. What is more, the weights of MKSC and AASC are negative on the Cornell dataset, which does not accord with the common sense of weighting. For the algorithm LMKC, the weight of each view is consistent with the clustering result on each dataset, i.e., the better the clustering result is, the greater the weight is. However, the weight differences of LMKC are not as apparent as those of WMSC.

For the algorithm WMSC, on the Handwritten, Flickr and Cornell datasets, the better the clustering result is, the greater the weight is. On the ThreeSources dataset, the second view performs the best among the three views and has the greatest weight. The first view performs better than the third view, however the weight of the first view is less than that of the third view by 0.0213. Generally, the weight of each view is almost proportional to its clustering result, which conforms the fact that a view is more important if it has better clustering performance.

In summary, the proposed weighting scheme is the best one. WMSC simultaneously exploits the complementary information of all the views and derives a satisfactory clustering result.

Table 3: NMI on all the datasets.

Algorithm\Dataset	ThreeSources	Handwritten	Flickr	Cornell
SC(1)	0.468±0.000	0.647±0.001	0.414±0.010	0.093±0.000
SC(2)	0.500±0.009	0.625±0.001	0.362±0.015	0.469±0.000
SC(3)	0.431±0.008	0.497±0.000	–	–
SC-cat	0.514±0.000	0.487±0.001	0.414±0.019	0.093±0.000
Coregspectral	0.525±0.007	0.750±0.000	0.436±0.012	0.520±0.058
RMSC	0.441±0.000	0.735±0.002	0.471±0.004	0.418±0.013
MVSC	0.476±0.036	0.717±0.011	0.436±0.027	0.256±0.171
MVSpec	0.501±0.000	0.715±0.000	0.376±0.000	0.501±0.000
MKSC	0.530±0.006	0.649±0.000	0.456±0.012	0.505±0.008
AASC	0.505±0.020	0.663±0.057	0.410±0.029	0.247±0.170
LMKC	0.505±0.000	0.786±0.000	0.420±0.000	0.434±0.000
WMSC	0.553±0.007	0.800±0.001	0.481±0.005	0.564±0.000

Table 4: Weight of each view.

	ThreeSources			Handwritten			Flickr		Cornell	
	SC(1)	SC(2)	SC(3)	SC(1)	SC(2)	SC(3)	SC(1)	SC(2)	SC(1)	SC(2)
MVSpec	0.3382	0.3227	0.3391	0.3333	0.3333	0.3333	0.3676	0.6324	0.5000	0.5000
MKSC	0.3286	0.3458	0.3256	0.2552	0.2611	0.4837	0.4246	0.5754	-1.5946	2.5946
AASC	0.3333	0.3340	0.3327	0.3316	0.3345	0.3339	0.5116	0.4884	1.0639	-0.0639
LMKC	0.2978	0.3906	0.3116	0.4097	0.4806	0.1097	1.0000	0.0000	0.4433	0.5567
WMSC	0.2047	0.5666	0.2260	0.4806	0.3905	0.1289	0.6534	0.3466	0.2766	0.7234

Parameter Study

WMSC introduces two parameters $\tilde{\beta}$ and $\tilde{\eta}$. In Figure 2, we experimentally show the effect of each parameter on the performance of WMSC. Figure 2(a) shows the NMI of WMSC on each dataset by varying $\tilde{\beta}$ in $\{0.02, 0.04, \dots, 0.18, 0.2\}$ with $\tilde{\eta} = 0.02$. Figure 2(b) shows the NMI of WMSC on each dataset by varying $\tilde{\eta}$ in $\{0.02, 0.04, \dots, 0.18, 0.2\}$ with $\tilde{\beta} = 0.02$.

From this figure, we find that WMSC has better performance on all the datasets when $\tilde{\beta} \in [0.08, 0.14]$ and $\tilde{\eta} \in [0.08, 0.16]$. These results indicate that the performance of WMSC is stable across a wide range of parameters.

Running Time Study

The computational complexity of WMSC has been analyzed in previous section (please refer to the original papers for the computational complexities of the baselines). In Figure 3, we compare the running time of the proposed algorithm and the baselines. WMSC consumes less time than RMSC and LMKC on all the datasets, and consumes more time than MVSC and MVSpec on all the datasets. Compared with other algorithms, sometimes WMSC consumes more time and sometimes WMSC consumes less time, e.g., WMSC consumes more time than Coregspectral on the Handwritten dataset but consumes less time than Coregspectral on the ThreeSources dataset. Therefore, we can conclude that the time needed by WMSC is medium among all the methods.

Related Work

Multi-view clustering algorithms integrate multiple compatible and complementary features to improve the clustering

performance. It is challenging to find a suitable way of simultaneously exploiting the complementary information of all the views in order to derive a satisfactory partition. Spectral clustering (Luxburg 2007) based multi-view methods have attracted much attention in the past decades.

From a weighted perspective, existing multi-view spectral clustering algorithms can roughly be grouped into two categories. The first one equally treats each view. The work in (de Sa 2005) created a bipartite graph based on the nodes co-occurring in both views and found a cut that crosses the fewest lines by using spectral clustering. The work in (Zhou and Burges 2007) generalized the normalized cut from a single view to multiple views via a random walk. Kumar et al. (Kumar, Rai, and Daume 2011) integrated multiple information by co-regularizing the clustering hypotheses. The work in (Kumar and Daumé 2011) reconstructed the similarity matrix of one view by using the eigenvectors of the Laplacian in other views. Xia et al. (Xia et al. 2014) conducted Markov chain based spectral clustering and learned the consensus transition probability matrix via low-rank and sparse decomposition. Li et al. (Li et al. 2015) approximated the similarity graphs using bipartite graphs. (Lu, Yan, and Lin 2016) boosted the clustering performance by using the multi-view information and sparse regularization. The work in (Lee and Liu 2016) learned an augmented view and constructed the corresponding affinity matrix from a spectral decomposition of an information-rich matrix. Feng et al. (Feng et al. 2017) proposed a novel multi-view clustering algorithm via robust local representation. The second one assigns the views different weights. The work in (Guo et al. 2014) learned a linear combination of multiple kernels and determined the coefficient by minimizing the kernel alignment

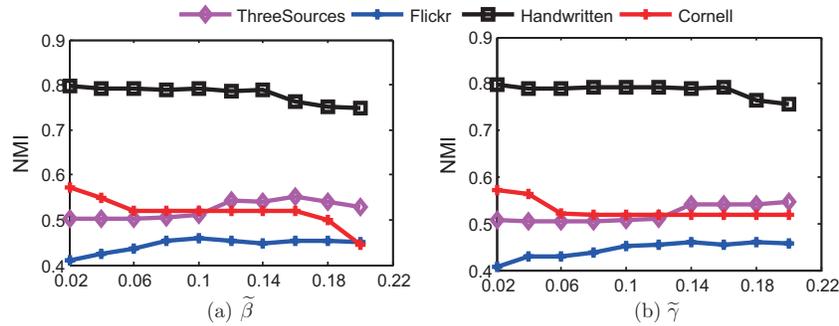


Figure 2: Parameter Study

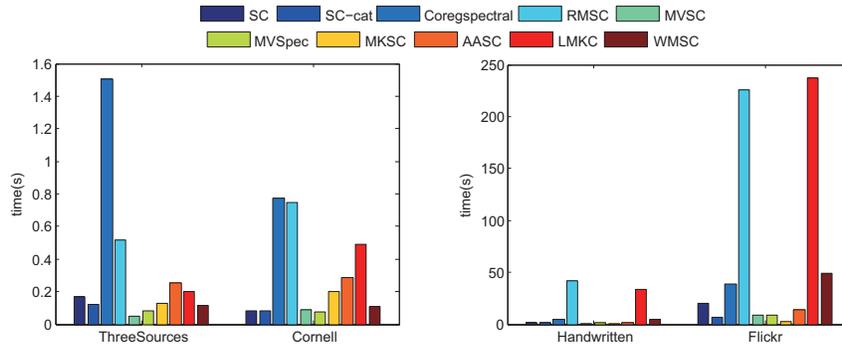


Figure 3: Running Time Study

between the learned kernel and the kernel of each view. The work in (Tzortzis and Likas 2012) learned a weighted combination of the kernels in parallel to clustering. Huang et al. (Huang, Chuang, and Chen 2012a) sought for an optimal combination of affinity matrices so that it is more immune to ineffective affinities and irrelevant features. Zong et al. (Zong et al. 2016) learned a unified similarity matrix from multiple locally linear neighborhood.

Multi-view spectral clustering is related to multiple kernel k-means in terms of kernel combination. Multiple kernel k-means integrates a group of pre-specified kernels to improve performance of k-means. Yu et al. (Yu et al. 2011) presented a novel optimized kernel k-means algorithm to combine multiple data sources for clustering analysis. The work in (Huang, Chuang, and Chen 2012b) incorporated multiple kernels and automatically adjusted the kernel weights. The work in (Gönen and Margolin 2014) combined kernels calculated on the views in a localized way to better capture sample-specific characteristics of the data. The work in (Lu et al. 2014) employed an effective kernel evaluation measure to unify two tasks of clustering and MKL into a single optimization framework. Du et al. (Du et al. 2015) simultaneously found the best clustering label, the cluster membership and the optimal combination of multiple kernels. The work in (Li et al. 2016) learned a weighted combination of the kernels with a local kernel alignment. The work in (Liu et al. 2016) proposed to learn a unified kernel by reducing

the redundancy and enhanced the diversity of the selected kernels.

To the best of our knowledge, the proposed algorithm is the first one to use spectral perturbation theory for view weighting.

Conclusion

Most multi-view clustering algorithms treat the views equally, however, it is rational to weight the views according to their contributions to the consensus result. In this paper, we have proposed a principled weighted multi-view spectral clustering algorithm which weights the views based on spectral perturbation theory. The weighting scheme tries to make the clustering result on each view close to the consensus clustering result, and smooth the weights with similar cluster ability. It is formulated into a standard quadratic programming problem. Experimental results show that the proposed algorithm outperforms typical single-view spectral clustering algorithms, unweighted multi-view spectral clustering algorithms and existing weighted multi-view spectral clustering algorithms, thus is effective for clustering multi-view data.

Acknowledgments

This work was partially supported by National High Technology Research and Development Program (863 Program)

of China (No. 2015AA015403) and National Science Foundation of China (No. 61632019).

References

- Cai, X.; Nie, F.; and Huang, H. 2013. Multi-view k-means clustering on big data. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2598–2604. AAAI Press.
- Davis, C. 1970. The rotation of eigenvectors by a perturbation- iii. *Siam Journal on Numerical Analysis* 7(1):1–46.
- de Sa, V. R. 2005. Spectral clustering with two views. In *ICML workshop on learning with multiple views*.
- Du, L.; Zhou, P.; Shi, L.; Wang, H.; Fan, M.; Wang, W.; and Shen, Y. 2015. Robust multiple kernel k-means using l_{21} -norm. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, 3476–3482.
- Feng, L.; Cai, L.; Liu, Y.; and Liu, S. 2017. Multi-view spectral clustering via robust local subspace learning. *Soft Computing* 21(8):1–12.
- Gönen, M., and Margolin, A. A. 2014. Localized data fusion for kernel k-means clustering with application to cancer biology. In *Advances in Neural Information Processing Systems*, 1305–1313.
- Guo, D.; Zhang, J.; Liu, X.; Cui, Y.; and Zhao, C. 2014. Multiple kernel learning based multi-view spectral clustering. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, 3774–3779. IEEE.
- Huang, H.-C.; Chuang, Y.-Y.; and Chen, C.-S. 2012a. Affinity aggregation for spectral clustering. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 773–780. IEEE.
- Huang, H. C.; Chuang, Y. Y.; and Chen, C. S. 2012b. Multiple kernel fuzzy clustering. *IEEE Transactions on Fuzzy Systems* 20(1):120–134.
- Hunter, B., and Strohmer, T. 2010. Performance analysis of spectral clustering on compressed, incomplete and inaccurate measurements. *Mathematics*.
- Kumar, A., and Daumé, H. 2011. A co-training approach for multi-view spectral clustering. In *Proceedings of the 28th International Conference on Machine Learning*, 393–400.
- Kumar, A.; Rai, P.; and Daume, H. 2011. Co-regularized multi-view spectral clustering. In *Advances in Neural Information Processing Systems*, 1413–1421.
- Lee, C. K., and Liu, T. L. 2016. Guided co-training for multi-view spectral clustering. In *IEEE International Conference on Image Processing*, 4042–4046.
- Li, Y.; Nie, F.; Huang, H.; and Huang, J. 2015. Large-scale multi-view spectral clustering via bipartite graph. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*.
- Li, M.; Liu, X.; Wang, L.; Dou, Y.; Yin, J.; and Zhu, E. 2016. Multiple kernel clustering with local kernel alignment maximization. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 4042–4046.
- Liu, J.; Wang, C.; Gao, J.; Gu, Q.; Aggarwal, C.; Kaplan, L.; and Han, J. 2015. Gin: A clustering model for capturing dual heterogeneity in networked data. In *Proceedings of 2015 SIAM International Conference on Data Mining*.
- Liu, X.; Dou, Y.; Yin, J.; Wang, L.; and Zhu, E. 2016. Multiple kernel k -means clustering with matrix-induced regularization. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 1888–1894.
- Lu, Y.; Wang, L.; Lu, J.; Yang, J.; and Shen, C. 2014. Multiple kernel clustering based on centered kernel alignment. *Pattern Recognition* 47(11):3656–3664.
- Lu, C.; Yan, S.; and Lin, Z. 2016. Convex sparse spectral clustering: Single-view to multi-view. *IEEE Transactions on Image Processing* 25(6):2833–2843.
- Luxburg, U. V. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17(17):395–416.
- Ng, A. Y.; Jordan, M. I.; and Weiss, Y. 2002. On spectral clustering: Analysis and an algorithm. In *Proceedings of Advances in Neural Information Processing Systems*, volume 14, 849–856.
- Stewart, G. W., and Sun, J. G. 1990. *Matrix Perturbation Theory*. Academic Press, Boston.
- Tzortzis, G., and Likas, A. 2012. Kernel-based weighted multi-view clustering. In *12th IEEE International Conference on Data Mining*, 675–684.
- Xia, R.; Pan, Y.; Du, L.; and Yin, J. 2014. Robust multi-view spectral clustering via low-rank and sparse decomposition. In *AAAI*, 2149–2155.
- Xu, W.; Liu, X.; and Gong, Y. 2003. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 267–273.
- Yu, S.; Tranchevent, L.; Moor, B. D.; and Moreau, Y. 2011. Optimized data fusion for kernel k-means clustering. *IEEE Trans Pattern Anal Mach Intell* 34(5):1031–1039.
- Zhang, X.; Zong, L.; You, Q.; and Yong, X. 2016. Sampling for nyström extension-based spectral clustering: Incremental perspective and novel analysis. *Acm Transactions on Knowledge Discovery from Data* 11(1).
- Zhao, H.; Ding, Z.; and Fu, Y. 2017. Multi-view clustering via deep matrix factorization. In *AAAI*, 2921–2927.
- Zhou, D., and Burges, C. J. 2007. Spectral clustering and transductive learning with multiple views. In *Proceedings of the 24th International conference on Machine learning (ICML)*, 1159–1166.
- Zong, L.; Zhang, X.; Yu, H.; Zhao, Q.; and Ding, F. 2016. Local linear neighbor reconstruction for multi-view data. *Pattern Recognition Letters* 84:56–62.