Neurocomputing 318 (2018) 227–235

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Spectral clustering based on iterative optimization for large-scale and high-dimensional data

Yang Zhao[a,b], Yuan Yuan[a], Feiping Nie[c], Qi Wang[c,d,*]

[a] Center for OPTical IMagery Analysis and Learning (OPTIMAL), Xi'an Institute of Optics and Precision Mechanics of CAS, Xi'an 710119, China
[b] University of Chinese Academy of Sciences, Beijing 100049, China
[c] School of Computer Science, and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China
[d] Unmanned System Research Institute, Northwestern Polytechnical University, Xi'an 710072, China

ARTICLE INFO

Article history:
Received 2 February 2018
Revised 19 June 2018
Accepted 14 August 2018
Available online 1 September 2018

Communicated by Dr. Chenping Hou

Keywords:
Manifold learning
Spectral clustering
Multi-task learning

ABSTRACT

Spectral graph theoretic methods have been a fundamental and important topic in the field of manifold learning and it has become a vital tool in data clustering. However, spectral clustering approaches are limited by their computational demands. It would be too expensive to provide an optimal approximation for spectral decomposition in dealing with large-scale and high-dimensional data sets. On the other hand, the rapid development of data on the Web has posed many rising challenges to the traditional single-task clustering, while the multi-task clustering provides many new thoughts for real-world applications such as video segmentation. In this paper, we will study a Spectral Clustering based on Iterative Optimization (SCIO), which solves the spectral decomposition problem of large-scale and high-dimensional data sets and it well performs on multi-task clustering. Extensive experiments on various synthetic data sets and real-world data sets demonstrate that the proposed method provides an efficient solution for spectral clustering.

1. Introduction

Clustering is a fundamental technique for data mining and it has been widely used in various fields such as image segmentation [1,2], feature selection [3–5] and dimension reduction [6,7]. In the past decade, many clustering methods have been proposed and they have achieved a great success including hierarchical clustering methods [8,9], central grouping methods [10–12] and graph (theoretic) clustering [13–15].

In the literature, central grouping methods such as k-means and Fuzzy c-means have been successfully used in many fields for their simpleness. It aims to learn c cluster centroids that minimize the within cluster data distances. In hierarchical clustering methods, clusters are formed by iteratively dividing the patterns using top-down or bottom up approach [9]. However, these methods have difficulty dealing with irregularly-shaped clusters and gradual variation within groups [16]. On the other hand, some approaches based on grouping pairwise data such as spectral clustering have gradually become a significant clustering technique because of their empirical performance advantages when compared with the central grouping methods and the hierarchical clustering methods. The spectral clustering methods [17] do a low-dimension embedding of the affinity matrix and followed by a k-means clustering in the low dimensional space [18]. The utilization of data graph and manifold information makes it possible to process the data with complicated structure [19,20]. Accordingly, spectral clustering has been widely applied and shown its effectiveness in the real-world applications such as image and video segmentation [1,21]. For instance, Normalized cuts (Ncut) [22], a representative work of spectral clustering, considers image segmentation as a graph cutting problem that can be solved by spectral decomposition.

Although the graph-based clustering methods have performed well, It would be too expensive to calculate the pairwise distance of enormous samples and difficult to provide an optimal approximation for spectral decomposition in dealing with a large affinity matrix [23]. In the clustering process, the storage complexity of the affinity matrix is $O(n^2)$ and the time complexity of eigen decomposition of Laplacian matrix is $O(n^3)$, where n is the number of samples. Therefore, the expense of handling large-scale and high-dimension datasets is unaffordable for the traditional

[*] Corresponding author at: School of Computer Science, Center for OPTical IMagery Analysis and Learning (OPTIMAL), and Unmanned System Research Institute, Northwestern Polytechnical University, Xi'an 710072, China.

E-mail addresses: zhaoyang.opt@gmail.com (Y. Zhao), y.yuan1.ieee@gmail.com (Y. Yuan), feipingnie@gmail.com (F. Nie), crabwq@gmail.com (Q. Wang).

https://doi.org/10.1016/j.neucom.2018.08.059
0925-2312/© 2018 Elsevier B.V. All rights reserved.

spectral clustering algorithms and several recent studies have been carried out on this problem. An efficient way to get low-rank matrix approximation based on Nyström extension has been widely used in many kernel based learning tasks [24]. Different from the traditional spectral clustering technique, it has a fixed complexity that is dependent only on a few local edges instead of a large affinity matrix. Liu *et al.* [25] propose an efficient clustering algorithm for large-scale graph data by clustering the bipartite graph using spectral methods. Fast approximate spectral clustering proposed by Yan et al. [26] is based on a theoretical analysis that provides a statistical characterization of the effect of local distortion on the mis-clustering rate. The Nonnegative Matrix Factorization (NMF) [27,28] has been proposed as the relaxation technique for clustering with excellent performance. Zhu et al. [29] construct anchor-based similarity graph with Balanced K-means based Hierarchical K-means algorithm, and then performs spectral analysis on the graph. Therefore, spectral clustering for large-scale data is still a hot topic in the field of clustering technique.

On the other hand due to the rapid development of massive storage, fast networks and media sharing sites, single-task clustering approaches have limited abilities to process enormous data. Multi-task clustering [30–32] has received increasing attention since it can handle large-scale and high-dimensional datasets by exploiting the knowledge shared by multiple tasks. For instance, Quanquan et al. [33] proposed a novel clustering paradigm by learning the shared subspace for multi-task clustering and transduction transfer classification. Yang et al. [34] proposed a multi-task spectral clustering model by exploring both inter-task clustering correlation and intra-task learning correlation. Note that, the existing approaches only consider the distribution constraint of multiple related data sets, but ignore the coherence of samples among consecutive tasks. For instance, in the task of video segmentation, a single voxel is gradually changing along with successive frames while the whole voxels in each frame are also following a global distribution. Accordingly, we study the multi-task clustering from a new thought which concentrates on the coherence sample in multiple related data sets.

Motivated by the existing approaches, we propose an improved Spectral Clustering based on Iterative Optimization, namely SCIO, for large-scale and high-dimensional data. We use a small subset of samples to approximate the whole affinity matrix by Nyström extension and solve the eigenvalue problem by an efficient iterative optimization. Furthermore, we extend SCIO to multi-task clustering, namely mSCIO, which falls into the fields of multi-task learning and preforms multiple related tasks together for achieving a better performance. However, it's quite different from the existing multi-task learning approaches since we consider both the coherent samples among multiple related data sets and the individual distribution of each data set. The basic idea of our work is to extend the spectral clustering technique to the multi-task learning and mSCIO also provides an efficient solution for multi-task clustering by Nyström extension and iterative optimization. Therefore, this study makes a major contribution to propose a new insight to spectral clustering for large-scale data by a novel iterative optimization. The experiments illustrate that SCIO provides an efficient and effective technique to solve the eigenvalue problem and mSCIO extends the proposed SCIO to multi-task clustering.

Notations: Throughout the paper, all the matrices are written as uppercase such as matrix M. The (i, j)-th element of M are denoted by M_{ij} and $M \geq 0$ means all the elements of M are equal to or larger than zero. The trace of matrix M is denoted by $Tr(M)$ and the Frobenius norm of M is denoted by $||M||_F$. I denotes an identity matrix and $\mathbf{1}$ denotes a column vector with all the elements are one.

2. Spectral clustering revisit

Given a set of data points $\mathbb{X} = \{x_1, x_2, \ldots, x_n\}$ and their affinity (or similarity) of data points x_i and x_j. We can represent these data points by an undirected graph $G = \{V, E\}$. Each vertex represents a data point x_i and the edge is aligned by their similarity. The pairwise affinity matrix W of the graph G can be denoted as

$$W_{ij} = e^{\frac{-||x_i - x_j||_2^2}{2\sigma^2}}, \tag{1}$$

where σ is a parameter controlling the width of the neighbors. In this way the original clustering of the data set $\mathbb{X}$ has been transformed to a graph partition problem of the graph G. Considering a graph $G = \{V, E\}$, and A and B represent a bipartition of V, where $A \bigcup B = V$ and $A \bigcap B = \emptyset$, we can measure the quality of the partition according to Normalized Cut (Ncut) algorithm as

$$\text{Ncut}(A, B) = \frac{\text{cut}(A, B)}{\text{vol}(A, V)} + \frac{\text{cut}(B, A)}{\text{vol}(B, V)} = \frac{2\dot{\text{cut}}(A, B)}{\text{vol}(A)||\text{vol}(B)}, \tag{2}$$

where $\text{cut}(A, B) = \sum_{i \in A, j \in B} W_{ij}$, $\text{vol}(A, V) = \sum_{i \in A, j \in V} W_{ij}$ and $||$ denotes the harmonic mean. Appealing to spectral graph theory [35], the optimal partition can be found by computing:

$$\begin{aligned} y &= \arg\min \text{Ncut}(A, B) \\ &= \arg\min_y \text{Ncut}(y) \\ &= \arg\min_y \frac{y^T(D - W)y}{y^T D y}. \end{aligned} \tag{3}$$

Eq. (3) is in form of the standard Rayleigh-Ritz theorem if we relax y to take on real values instead of two discrete values. Shi and Malik [22] pointed out that an approximate solution may be obtained by thresholding the eigenvector corresponding to second smallest eigenvalue of the normalized Laplacian L, which is defined as:

$$L = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}}WD^{-\frac{1}{2}}. \tag{4}$$

Note that, the normalized Laplacian L is positive semidefinite even when the W is indefinite. And its second smallest eigenvalue λ_2 lie on the interval [0,2] so the corresponding eigenvalues of $D^{-\frac{1}{2}}WD^{-\frac{1}{2}} = I - L$ are confined to lie inside $[-1, 1]$.

For the case of multiple group clustering where $k > 2$, the above equation can be rewritten as follows:

$$\text{Ncut} = Tr(Y^T L Y), \tag{5}$$

where $Y^T Y = I$ and Y contains the k top eigenvectors of the normalized Laplacian L. And this can be solved by the standard trace minimization problem according to the normalized spectral clustering proposed in [22]. The solution Y consists of the first k eigenvectors of the normalized Laplacian L as columns.

However, the traditional spectral clustering technique is struggling with large-scale and high-dimensional data sets since the affinity matrix W grows as the square of the number of the samples in the grouping problem, it quickly becomes infeasible to fit W in memory and a large amount of time and memory is also required to calculate and store the first k eigenvectors of a Laplacian matrix. One approach to solving this problem is to only calculate the neighbor samples to make the Laplacian L sparse and permits the use of an efficient eigensolver. However, it is still a big problem to large-scale datasets and this discourages the long-rage connections. Other researchers attempt to address this problem and make spectral clustering algorithms more applicable to large scale datasets and an alternative approach based on Nyström extension has been demonstrated to be more effective in many kernel-based learning tasks, and we will present the details in the following part.

The spectral clustering with Nyström extension only choose m samples at random from the full set of N samples where $m \ll N$.

For complicity in notion, the chosen samples so that m come first and the sub-matrix A is the affinities among them. The remaining $n = N - m$ samples come next and the sub-matrix B presents the affinities between the chosen samples and the remaining samples. The affinity matrix W can be rewritten as

$$W = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}, \tag{6}$$

where $A \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{m\times n}$ and $C \in \mathbb{R}^{n\times n}$. A is the sub-matrix of affinities among the chosen samples, B represents the affinities from the chosen samples to the remaining samples, and C is the affinities among all of the remaining samples. According to the Nyström extension, C can be approximated by $B^T A^{-1} B$ and the full affinity matrix W can be approximated by

$$\hat{W} = \begin{bmatrix} A & B \\ B^T & B^T A^{-1} B \end{bmatrix} = \begin{bmatrix} A \\ B^T \end{bmatrix} A^{-1} \begin{bmatrix} A & B \end{bmatrix}. \tag{7}$$

Note that, the affinity matrix $\hat{W}$ is a Gram matrix that is positive definite. Using the diagonalization $A = U\Lambda U^T$, where $U^T U = I$, the Eq. (7) is rewritten as

$$\hat{W} = \begin{bmatrix} U \\ B^T U \Lambda^{-1} \end{bmatrix} \Lambda \begin{bmatrix} U^T & \Lambda^{-1} U^T B \end{bmatrix}. \tag{8}$$

To apply the matrix form of the Nyström extension to Ncuts, it is necessary to compute the row sum of $\hat{W}$ and this is possible without explicitly evaluating the $B^T A^{-1} B$ block as

$$\hat{D}\mathbf{1} = \hat{W}\mathbf{1} = \begin{bmatrix} A\mathbf{1}_m + B\mathbf{1}_n \\ B^T\mathbf{1}_m + B^T A^{-1} B\mathbf{1}_n \end{bmatrix}, \tag{9}$$

where $A\mathbf{1}_m$ and $B\mathbf{1}_n$ are the row sum of A and B, respectively, and $B^T\mathbf{1}_m$ is the column sum of B. The block A and B are given as

$$\begin{aligned} A_{ij} &\leftarrow \frac{A_{ij}}{\sqrt{\hat{D}_{ii}\hat{D}_{jj}}}, & i, j &= 1, 2, \ldots, m, \\ B_{ij} &\leftarrow \frac{B_{ij}}{\sqrt{\hat{D}_{ii}\hat{D}_{(j+m)(j+m)}}}, & i &= 1, \ldots, m, j = 1, \ldots, n, \end{aligned} \tag{10}$$

and $\hat{D}^{-1/2}\hat{W}\hat{D}^{-1/2}$ can be approximated by Eq. (8) directly. However, there is one problem that the columns of $[U^T \quad \Lambda^{-1}U^T B]^T$ are not necessarily orthogonal and this can be solved by introducing $Q = A + A^{-1/2}BB^T A^{-1/2}$ and diagonalizing Q as $Q = R\Lambda_R R$. Defining the matrix $\hat{V}$ as

$$\hat{V} = \begin{bmatrix} A \\ B^T \end{bmatrix} A^{-1/2} R \hat{\Lambda}_R^{-1/2}. \tag{11}$$

We can find that $\hat{V}$ and $\hat{\Lambda}$ diagonalize $\hat{D}^{-1/2}\hat{W}\hat{D}^{-1/2}$ as $\hat{D}^{-1/2}\hat{W}\hat{D}^{-1/2} = \hat{V}\hat{\Lambda}_R\hat{V}^T$ and $\hat{V}^T\hat{V} = I$. The details of proof can be found in [16].

3. Spectral clustering based on iterative optimization

Motivated by the spectral clustering and the Nyström extension, we attempt to propose a spectral clustering based on iterative optimization for multiple related data sets. The proposed SCIO is not only efficient for the traditional single-task clustering in the task of large-scale and high-dimensional data sets but also available for multiple related data sets with coherent samples as SCIO could be considered as a special case of mSCIO. The details are presented in the following parts.

3.1. Formulation and motivation

Given multiple related datasets $\{\mathbb{X}_1, \mathbb{X}_2, \ldots, \mathbb{X}_N\}$, the data points of $\mathbb{X}_n$ not only follow their own distribution but also constrained by their related datasets $\mathbb{X}_{n-1}$. According to the traditional spectral clustering, $\mathbb{X}_n$ can be represented as a graph $G_n = \{V_n, E_n\}$. And the pairwise affinity matrix W_n of the graph G_n can be represented as

$$[W_n]_{ij} = e^{\frac{-||[x_n]_i - [x_n]_j||_2^2}{2\sigma^2}}, \tag{12}$$

where $[x_n]_i$ and $[x_n]_j$ are the samples in dataset X_n and the parameter σ controls the width of the neighborhoods. However, σ is sensitive to different data sets and difficult to represent the affinity matrix M with a fixed σ. To overcome this problem, we propose an adaptive parameter σ as

$$\sigma^2 = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} ||[x_n]_i - [x_n]_j||_2^2. \tag{13}$$

It's an easy but effective way to make sure that the elements of affinity matrix W_n can be around $e^{-1/2}$. In our experiments, the adaptive parameter σ is good enough for most data sets.

To extend single-task spectral clustering to multiple learning tasks, we introduce an additional constraint $||Y_n - Y_{n-1}||_F^2$. Y_n is the indicator matrix of graph G_n where $Y_n^T Y_n = I$ and $Y_n \geq 0$. According to [36], the indicator matrix Y_n is orthonormal and nonnegative simultaneously, there are only one element is positive and others are zeros in each row of Y_n. Moreover, this constraint can help us to solve the spectral clustering in a more efficient way and make it easy to implement. In this case, Eq. (5) can be rewritten as

$$Ncut_n = \sum_{n=1}^{N} Tr(Y_n^T L_n Y_n) + \gamma ||Y_n - Y_{n-1}||_F^2, \tag{14}$$

with the condition $Y_n^T Y_n = I$, and $\gamma \geq 0$ is the hyperparameter to control the coherence of multiple related data sets. According to the Lagrangian function, Eq. (14) can be rewritten as:

$$\begin{aligned} Ncut_n = &\sum_{n=1}^{N} Tr(Y_n^T L_n Y_n) + \gamma ||Y_n - Y_{n-1}||_F^2 \\ &+ \lambda ||Y_n^T Y_n - I||_F^2, \end{aligned} \tag{15}$$

where $\lambda > 0$ is the Lagrangian multiplier and we consider $Y_n^T Y_n = I$ as a new constraint that is rewritten as $||Y_n^T Y_n - I||_F^2$. Eq. (15) can be divided into three parts, $Tr(Y_n^T L_n Y_n)$ is the item for spectral clustering, $||Y_n^T Y_n - I||_F^2$ is the item for orthonormal constraint and $||Y_n - Y_{n-1}||_F^2$ is the item for multi-task constraint. In this work, we only consider the relation between two adjacent samples (x_{n-1} and x_n) and the multi-task constraint $||Y_n - Y_{n-1}||_F^2$ must be a convex function. However, complex multi-task constraints that can be presented as convex functions are also allowed but it is difficult to calculate the derivatives of them. Note that, if $\gamma = 0$, it's a single-task spectral clustering. However, Eq. (15) is still a non-smooth objective function and difficult to be solved efficiently. Motivated by NMF which has excellent performance in dealing with clustering by relaxation technique [37]. We relax the discreteness condition and introduce an iterative optimization to solve the eigenvalue problem, and the details will be illustrated in the next section.

3.2. Iterative optimization

Because of the additional multi-task constraint $||Y_n - Y_{n-1}||_F^2$, it's difficult to solve the eigenvalue problem by traditional Rayleigh quotient. As a result, we relax the discreteness condition and transform the eigenvalue problem to a minimization problem as

$$\begin{aligned}
\min \sum_{n=1}^{N} & Tr(Y_n^T L_n Y_n) + \gamma ||Y_n - Y_{n-1}||_F^2 \\
& + \lambda ||Y_n^T Y_n - I||_F^2 \\
= & \sum_{n=1}^{N} \min Tr(Y_n^T L_n Y_n) + \gamma ||Y_n - Y_{n-1}||_F^2 \\
& + \lambda ||Y_n^T Y_n - I||_F^2 \\
= & \sum_{n=1}^{N} \min E(Y_n). \qquad (16)
\end{aligned}$$

Note that, Y_n is only relevant to L_n and Y_{n-1}, and Y_{n-1} is given by the previous item E_{n-1}. We can separate the global minimization problem into several sub-problems and get each Y_n according to their recurrence relation. The traditional spectral relaxation approach relaxes the indicator matrix Y_n to orthonormal constraints, where $Y_n^T Y_n = I$. According to [36], the indicator matrix Y_n is orthonormal and nonnegative simultaneously, there are only one element is positive and others are zeros in each row of Y_n. Moreover, this constraint can help us to solve the traditional spectral clustering in a more efficient way and make it more easy to implement. $E(Y_n)$ in Eq. (16) can be rewritten as follows:

$$\begin{aligned}
E(Y_n) = & Tr(Y_n^T L_n Y_n) \\
& + \gamma Tr((Y_n - Y_{n-1})^T (Y_n - Y_{n-1})) \\
& + \lambda Tr((Y_n^T Y_n - I)^T (Y_n^T Y_n - I)), \qquad (17)
\end{aligned}$$

where $Y_n \geq 0$. The derivative of $E(Y_n)$ is

$$\frac{1}{2}\frac{\partial O}{\partial Y_n} = L_n Y_n + \gamma Y_n - \gamma Y_{n-1} + 2\lambda Y_n Y_n^T Y_n - 2\lambda Y_n, \qquad (18)$$

where $L_n = I - D_n^{-\frac{1}{2}} W_n D_n^{-\frac{1}{2}}$, W_n is the pairwise affinity matrix of graph G_n and D_n is the diagonal matrix with the i-th diagonal element as $D_n^{(ii)} = \sum_j W_n^{(ij)}$. D_n and W_n are symmetric positive semidefinite matrix and nonnegative matrix. Eq. (18) can be rewritten as

$$\begin{aligned}
\frac{1}{2}\frac{\partial O}{\partial Y_n} = & (I - D_n^{-\frac{1}{2}} W_n D_n^{-\frac{1}{2}}) Y_n \\
& + \gamma Y_n - \gamma Y_{n-1} + 2\lambda Y_n Y_n^T Y_n - 2\lambda Y_n \\
= & (Y_n + \gamma Y_n + 2\lambda Y_n Y_n^T Y_n) \\
& - (D_n^{-\frac{1}{2}} W_n D_n^{-\frac{1}{2}} Y_n + \gamma Y_{n-1} + 2\lambda Y_n) \\
= & Q_n - P_n. \qquad (19)
\end{aligned}$$

In this way, the derivative of $O(Y_n)$ is divided into two matrix $P_n \geq 0$ and $Q_n > 0$ as we define Y_n is a positive matrix and $D_n^{-\frac{1}{2}} W_n D_n^{-\frac{1}{2}}$ is a nonnegative matrix.

According to [36], it leads to the following multiplicative update formula to update the elements $[Y_n]_{ij}$ of the indicator matrix Y_n as

$$[Y_n]_{ij} \leftarrow [Y_n]_{ij} \sqrt{\frac{[P_n]_{ij}}{[Q_n]_{ij}}}, \qquad (20)$$

where $P_n = (D_n^{-\frac{1}{2}} W_n D_n^{-\frac{1}{2}} Y_n + \gamma Y_{n-1} + 2\lambda Y_n)$ and $Q_n = (Y_n + \gamma Y_n + 2\lambda Y_n Y_n^T Y_n)$. We update the indicator matrix Y_n according to Eq. (20) until Eq. (15) is convergent and the implement details are presented in Algorithm 1 . Note that, there are only one element is positive and others approximate zero in each row of the indicator matrix Y_n according to [36]. And Y_n can be considered as a nearly perfect indicator matrix to represent the clustering results.

```
Algorithm 1: Algorithm to solve the problem (16).
Input: Multiple datasets $\{\mathbb{X}_1, \mathbb{X}_2, \ldots, \mathbb{X}_N\}$ and cluster number $k$.
Output: Indicator matrices $\{Y_1, Y_2, \ldots, Y_N\}$.
Initialize $\gamma$, $\lambda$ and each indicator matrix $Y_n \in \mathbb{R}^{n \times k}$ randomly such that $Y_n > 0$;
Choose m samples in each data set $\mathbb{X}_n$ and calculate the affinity matrix $A_n$ and $B_n$ according to Eq. (12) and Eq. (13);
Calculate the diagonal matrix set $\mathbb{D} = \{\hat{D}_1, \hat{D}_2, \ldots, \hat{D}_N\}$ according to Eq. 9;
Update the affinity matrix $A_n$ and $B_n$ according to and Eq. 10;
Calculate the approximated matrix $\hat{D}_n^{-\frac{1}{2}} \hat{W}_n \hat{D}_n^{-\frac{1}{2}}$ according to Eq. 21;
for $X_1, \ldots, X_N$ do
    if $X_1$ then
        $Y_{n-1} = Y_1$
    end
    while Eq. (15) not converge do
        1. Calculate numerator function:
           $P(Y_n) = \hat{D}_n^{-\frac{1}{2}} \hat{W}_n \hat{D}_n^{-\frac{1}{2}} + \gamma Y_{n-1} + 2\lambda Y_n$;
        2. Calculate denominator function:
           $Q(Y_n) = Y_n + \gamma Y_n + 2\lambda Y_n Y_n^T Y_n$;
        3. Update each element in indicator matrix $Y_n$ according to Eq. (20):
           $Y_n^{(ij)} = Y_n^{(ij)} \sqrt{\frac{P(Y_n)^{(ij)}}{Q(Y_n)^{(ij)}}}$;
    end
end
Output the indicator matrix set $\mathbb{Y} = \{Y_1, Y_2, \ldots, Y_N\}$.
```

To further improve the computing efficiency of affinity matrix to make the proposed algorithm available for comparing high-dimension pairwise samples, we use Nyström extension to approximate the full affinity matrix by a few chosen samples as

$$\hat{D}_n^{-\frac{1}{2}} \hat{W}_n D_n^{-\frac{1}{2}} = \begin{bmatrix} A_n \\ B_n^T \end{bmatrix} A_n^{-1} \begin{bmatrix} A_n & B_n \end{bmatrix}, \qquad (21)$$

Note that, $D_n^{-\frac{1}{2}} \hat{W}_n D_n^{-\frac{1}{2}}$ can be approximated by A_n and B_n according to Eqs. (9) and (10) but the elements in $\hat{D}_n^{-\frac{1}{2}} \hat{W}_n \hat{D}_n^{-\frac{1}{2}}$ may be negative as A_n^{-1} could be negative. We have to keep the matrix P_n is nonnegative to update the indicate matrix Y_n and we propose two solutions to solve this problem. **(1)** the $D_n^{-\frac{1}{2}} W_n D_n^{-\frac{1}{2}}$ is approximated by $\hat{D}_n^{-\frac{1}{2}} \hat{W}_n \hat{D}_n^{-\frac{1}{2}}$, and most elements in $\hat{D}_n^{-\frac{1}{2}} \hat{W}_n \hat{D}_n^{-\frac{1}{2}}$ should be positive as $D_n^{-\frac{1}{2}} W_n D_n^{-\frac{1}{2}}$ is a positive matrix. We could consider the negative elements in $\hat{D}_n^{-\frac{1}{2}} \hat{W}_n \hat{D}_n^{-\frac{1}{2}}$ are noises and set the negative elements to zero. **(2)** A_n and B_n are nonnegative, we define $A_n^{-1} = A_n^{\dagger} - A_n^{-}$ where $A_n^{\dagger}$ is the positive part of A_n^{-1} and A_n^{-} is the negative part of A_n^{-1}. Note that, both $A_n^{\dagger}$ and A_n^{-} are nonnegative matrix so we can rewrite P_n and Q_n as

$$\begin{aligned}
P_n = & \begin{bmatrix} A_n \\ B_n^T \end{bmatrix} A^{\dagger} \begin{bmatrix} A_n & B_n \end{bmatrix} Y_n + \gamma Y_{n-1} + 2\lambda Y_n, \\
Q_n = & \begin{bmatrix} A_n \\ B_n^T \end{bmatrix} A^{-} \begin{bmatrix} A_n & B_n \end{bmatrix} Y_n + Y_n + \gamma Y_n + 2\lambda Y_n Y_n^T Y_n. \qquad (22)
\end{aligned}$$

In this case, the P_n and Q_n are nonnegative matrix and we can follow Eq. 20 to update Y_{ij}.

Table 1
Datasets description.

	Dimension	Class	Number
Mnist	784	10	60,000
TDT2	36,771	30	9394
WebKB	4,029 ~ 4,189	7	814 ~ 1,210
RCV 1	29,992	4	1,925
Reuters21578	18,933	65	1658

4. Experiments

In this section, we evaluate the effectiveness of the proposed algorithm for graph clustering on large-scale and high-dimension data sets including both synthetic datasets and real-world datasets with respect to single-task learning and multi-task learning. For single-task algorithms, the clustering approaches including *k*-means (KM) clustering, Fuzzy *c*-means (FCM) clustering, hierarchical clustering (HC) and Ncut are used for comparison. We also compare the proposed SCIO with Nyström Ncut (NysNcut) with respect to video segmentation since it is closely related to SCIO. The algorithm was implemented using Matlab and experiments were conducted on Intel Core i5 PC with 32G RAM.

4.1. Data description

We verify the clustering performance on five real-world data sets including Mnist, TDT2, WebKB, RCV1, and Reuters21578 as follows:

- The Mnist [38] dataset that is a handwritten digits dataset containing 60,000 examples that are categorized into 10 groups. In the experiments, we use 7 subsets of the Mnist data sets that contain 5000, 7500, 10,000, 12,500, 15,000, 17,500, and 20,000 samples.
- The TDT2 corpus [39,40] consists of data collected during the first half of 1998 and taken from 6 sources, including 2 newswires (APW, NYT), 2 radio programs (VOA, PRI) and 2 television programs (CNN, ABC). It consists of 11201 on-topic documents that are classified into 96 semantic categories. In our experiments, we evaluate the algorithms with 3 subsets with 5, 10, and 15 classes.
- The samples in the WebKB data set are webpages downloaded from 4 universities (Cornell, Texas, Washington, and Wisconsin) data set homepage, manually classified into seven different classes: student, faculty, staff, department, course, project, and others.
- Reuters-21,578 [41] is a collection of documents that appeared on Reuters newswire in 1987 and it contains 21,578 documents in 135 categories, we use a subset of Reuters-21,578 and it has 1659 samples with 18,933 documents.
- RCV1 [42,43] is a text categorization test collection that is distributed as a set of on-line appendices to a JMLR journal article. We use a subset of RCV1 which has 1925 documents with 29,992 distinct words, including categories “C15”, “ECAT”, “GCAT”, and “MCAT”. Table 1 summarizes the details of all data sets used in the experiments.

We also perform an video segmentation evaluation to further prove the effectiveness and efficiency of our approach on the multi-task clustering. The Video Segmentation Benchmark (VSB100) provides ground truth annotations for the Berkeley Video Dataset, which consists of 100 HD quality videos. In the experiments, we use the videos with half resolution and segment each frame into 10 regions for the tradeoff between over-segmentation and segmentation accuracy.

4.2. Evaluation metrics

We adopt the performance measures used in [44] which are Normalized Mutual Information (NMI) and Computational Time (CT) in the experiment of single task. The CT is the computational time of the clustering and the NMI can be estimated by

$$NMI = \frac{\sum_{i=1}^{c}\sum_{j=1}^{c} n_{i,j} log\frac{n_{i,j}}{n_i\hat{n}_j}}{\sqrt{\left(\sum_{i=1}^{c} n_i log\frac{n_i}{n}\right)\left(\sum_{i=1}^{c} \hat{n}_i log\frac{\hat{n}_i}{n}\right)}}, \tag{23}$$

where n_i denotes the number of data contained in the cluster $\mathcal{C}_i (1 \le i \le c)$, $\hat{n}_j$ is the number of data belonging to the $\mathcal{L}_j (1 \le j \le c)$, and $n_{i,\,j}$ denotes the number of data that are in the intersection between the cluster $\mathcal{C}_i$ and the class $\mathcal{L}_j$. The larger the NMI is, the better the clustering result will be.

We use two standard evaluation metrics including Boundary Precision-Recall (BPR) and Volume Precision-Recall (VPR) to evaluate the performance for video segmentation. The boundary metric is most popular in the BSDS benchmark for image segmentation. It casts the boundary detection problem as one of classifying boundary from non-boundary pixels and measures the quality of a segmentation boundary map in the precision-recall framework:

$$\begin{aligned} P &= \frac{|S \cap (\bigcup_{i=1}^{M} G_i)|}{|S|}, \\ R &= \frac{\sum_{i=1}^{M} |S \cap G_i|}{\sum_{i=1}^{M} |G_i|}, \\ F &= \frac{2PR}{R + P}, \end{aligned} \tag{24}$$

where S is the set of machine generated segmentation boundaries and $G_{i i=1}^{M}$ are the M sets of human annotation boundaries. F-measure is used to evaluate aggregate performance.

VPR optimally assigns spatiotemporal volumes between the computer generated segmentation $\mathbb{S}$ and the M human annotated segmentation $\{\mathbb{G}\}_{i=1}^{M}$ and measures their overlap as:

$$\begin{aligned} P &= \frac{1}{M}\sum_{i=1}^{M} \frac{\sum_{s\in\mathbb{S}} \max_{g\in\mathbb{G}_i} |s \cap g|}{|\mathbb{S}|}, \\ R &= \sum_{i=1}^{M} \frac{\sum_{g\in\mathbb{G}_i} \max_{s\in\mathbb{S}} |s \cap g|}{\sum_{i=1}^{M} |\mathbb{G}_i|}, \end{aligned} \tag{25}$$

where the volume overlap is expressed by the intersection operator ∩ and |.| denotes the number of pixels in the volume.

4.3. Toy example

In this section, we will provide a toy experiment to verify the effectiveness of the proposed approach on five single-task synthetic data sets including Double Spirals (DS), Cluster in Cluster (CC), Four Corners (FC), MooN (MN) and OutLier (OL). Each data set contains 10,000 samples that are divided into two or four groups. Fig. 1 lists the different data sets in details. These data sets are challenging as they are manifold and difficult to cluster only by the distance, especially for DS and CC. The remaining data sets including CC, MN and OL are unbalanced and it is also a challenging problem in the task of clustering.

According to the observations in Table 2, KM and FCM are much faster than others but they do not perform well because they only consider the local distance but ignore the global distribution. HC and Ncut achieve better performance than KM and FCM, but Ncut spends too much time calculating the affinity matrix and solving the eigenvalue problem. HC is perfect for the synthetic data sets with clear boundaries, but it is highly sensitive to noises and unclear boundaries. And the detailed discussion will be presented in

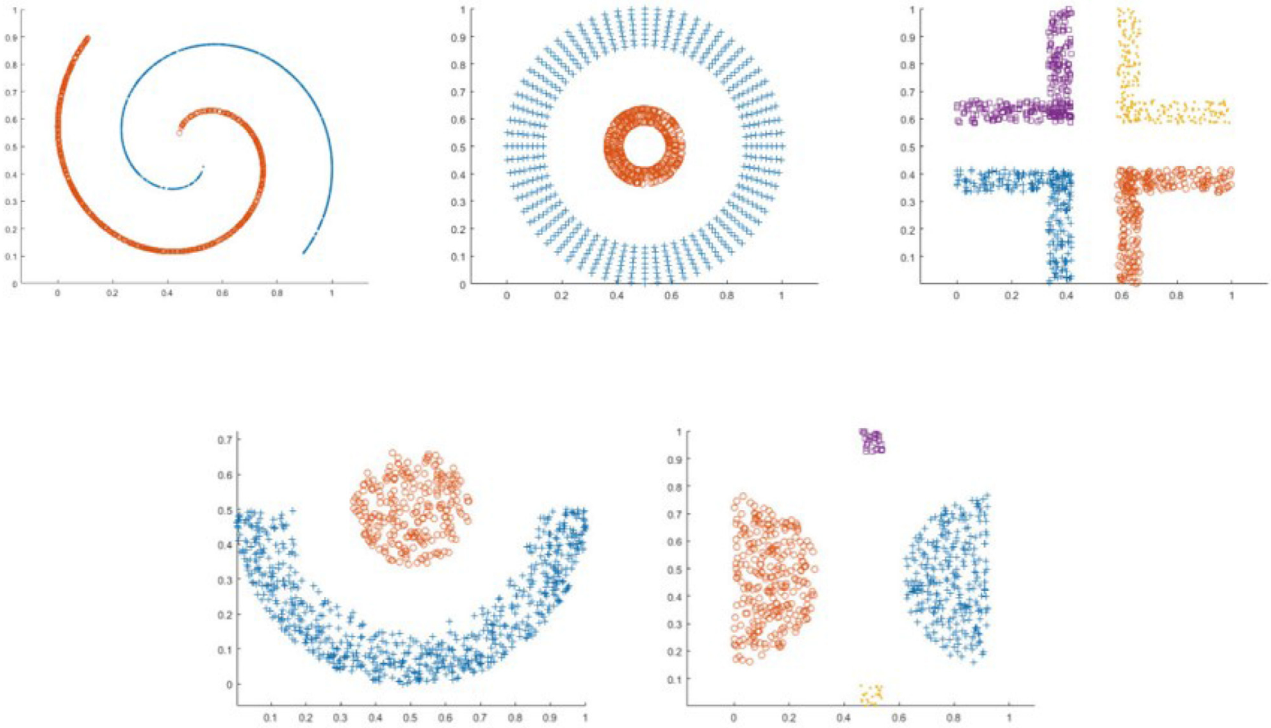


Fig. 1. Five synthetic datasets including Double Spirals (DS), Cluster in Cluster (CC), Four Corners (FC), MooN (MN) and OutLier (OL).

Table 2
Clustering results on five synthetic data sets.

	CC		DS		FC		MN		OL		Average	
	NMI	CT(s)	NMI	CT(s)	NMI	CT(s)	NMI	CT(s)	NMI	CT(s)	NMI	CT(s)
KM	0.01	0.05	0.17	0.03	0.75	0.02	0.35	0.02	0.73	0.02	0.40	0.03
FCM	0.01	0.27	0.11	0.09	0.90	0.34	0.37	0.23	0.65	0.14	0.43	0.17
HC	1.00	2.31	1.00	3.33	1.00	3.23	1.00	3.34	1.00	3.27	1.00	3.11
Ncut	1.00	69.25	1.00	69.32	1.00	69.58	1.00	69.84	1.00	69.48	1.00	69.37
MSCIO (iter.=100)	1.00	18.22	0.38	18.38	1.00	18.27	1.00	18.50	1.00	18.51	0.87	19.68
MSCIO (iter.=200)	1.00	27.36	0.65	28.51	1.00	30.18	1.00	28.44	1.00	36.31	0.93	29.80
MSCIO (iter.=300)	1.00	36.58	1.00	38.51	1.00	42.38	1.00	37.49	1.00	51.33	1.00	40.65

the experiments of the real-world data sets. The last three rows in Table 2 present our approach with different iteration times (100, 200, 300). As seen from the results, SCIO with 100 iterations is good enough for FC, MN, and OL, but the data sets such as DS and CC need more iterations to get a better result, because DS and CC follow the manifold distribution. Notwithstanding, SCIO is a more efficient algorithm than Ncut under the same accuracy.

4.4. Clustering accuracy comparison

In this part, we will study the results of different methods on five real-world data sets. By repeating each method for 20 independent runs, we get NMI and CT of the different methods. We directly apply the default functions and their parameters for KM, FCM and HC, which are provided by Matlab 2014. The parameter τ is set to 0 since we do not need the multi-task constraint $||Y_n - Y_{n-1}||_F^2$, and λ is set to 0.5. Comparing to the synthetic data sets, the real-world data sets have more noise and the boundary samples are ambiguous. Some data sets are resized, and Table 1 presents the data sets descriptions. We use Gaussian function to construct the affinity matrix W according to Eq. 1 and σ is adaptively calculated from Eq. (13). In the experiment of video segmentation, we follow the parameter setting in single-task clustering. However, the pixels in each frame are not only constrained by their locations but also related to the similarity of their intensity. Accordingly, the affinity matrix W is calculated as

$$W_{ij} = e^{-\frac{||X_i - X_j||_2^2}{4\sigma_X^2} - \frac{||I_i - I_j||_2^2}{4\sigma_I^2}}, \tag{26}$$

where X_i and I_i denote pixel location and intensity. σ_X and σ_I are calculated by Eq. (13), respectively. The parameter τ and λ is set to 0.5 as the current indicator matrix Y_n is constrained by its previous indicator matrix Y_{n-1} according to Eq. (14).

The single-task clustering performance of different methods, evaluated by NMI and CT, is presented in Table 3. The results of video segmentation on the VSB100 data set is shown in Table 4. There are several observations from the performance comparisons as follows:

- Based on an overall analysis of NMI and CT, the proposed SCIO performs best in single-task clustering as it achieves the competitive performance in most cases and it's also an efficient algorithm in dealing with large-scale and high-dimensional data sets. Especially in the experiment of TDT2 which contains extremely high-dimensional features, SCIO is much more efficient than Ncut and KM. This is mainly due to the fact that, SCIO reduces the computation of affinity matrix and provides a more efficient algorithm to solve the spectral decomposition problem.

Table 3
Clustering results on the real-world datasets.

	FCM		HC		KM		Ncut		SCIO	
	NMI	CT(s)	NMI	CT(s)	NMI	CT(s)	NMI	CT(s)	NMI	CT(s)
$Mnist_{No.=5000}$	0.14	1.81	0.02	9.35	0.47	6.96	0.47	17.36	**0.52**	15.63
$Mnist_{No.=7500}$	0.15	3.42	0.02	21.62	0.47	11.63	0.47	43.86	**0.48**	30.23
$Mnist_{No.=10,000}$	0.12	4.77	0.02	45.13	0.49	12.87	0.48	80.96	**0.52**	52.74
$Mnist_{No.=12,500}$	0.15	5.51	0.01	60.70	0.48	31.38	**0.49**	138.63	0.48	83.80
$Mnist_{No.=15,000}$	0.13	6.54	0.01	102.00	0.47	22.35	**0.48**	214.32	0.47	119.58
$Mnist_{No.=17,500}$	0.22	7.21	0.01	118.04	0.48	36.66	0.47	288.79	**0.49**	162.72
$Mnist_{No.=20,000}$	0.18	7.52	0.01	154.67	0.47	37.18	0.47	428.68	**0.50**	214.27
$TDT2_{Cls.=5}$	0.67	3.48	0.02	12.47	0.69	19.42	0.74	48.86	**0.76**	13.41
$TDT2_{Cls.=10}$	0.34	15.92	0.02	81.44	0.61	362.37	**0.70**	308.25	0.67	78.47
$TDT2_{Cls.=15}$	0.40	46.94	0.04	192.35	0.63	718.10	**0.65**	681.79	0.62	172.75
$WebKB_{cornell}$	0.01	2.89	0.02	1.50	0.12	6.59	0.18	5.15	**0.22**	2.03
$WebKB_{texas}$	0.02	1.26	0.05	1.12	0.21	6.70	**0.26**	4.82	0.22	2.05
$WebKB_{cornell}$	0.01	2.69	0.04	1.82	0.16	6.06	**0.27**	11.11	0.22	4.11
$WebKB_{wisconsin}$	0.01	2.89	0.03	1.50	0.13	10.21	**0.26**	12.17	0.24	4.59
$RCV1_{Cls.=4}$	0.19	5.56	0.01	53.04	0.12	168.34	0.27	199.97	**0.32**	47.28
*Reuters*21578	0.35	81.82	0.13	24.68	0.47	226.98	**0.49**	97.76	0.46	33.64
Average	0.19	18.57	0.03	52.84	0.40	117.46	**0.45**	158.85	**0.45**	61.70

Table 4
Video segmentation results on VBS100 datasets.

		NysNcut			MSCIO		
SI		200	150	100	200	150	100
BPR	R	0.61	0.67	–	0.81	0.82	**0.84**
	P	0.19	**0.21**	–	0.19	**0.21**	**0.21**
	F	0.29	0.32	–	0.31	**0.33**	**0.33**
VPR	R	**0.62**	0.61	–	0.54	0.50	0.52
	P	0.25	0.32	–	0.30	0.38	**0.42**
	F	0.35	0.42	–	0.39	0.43	**0.46**
CT		6.07	14.27	–	5.73	6.50	17.94

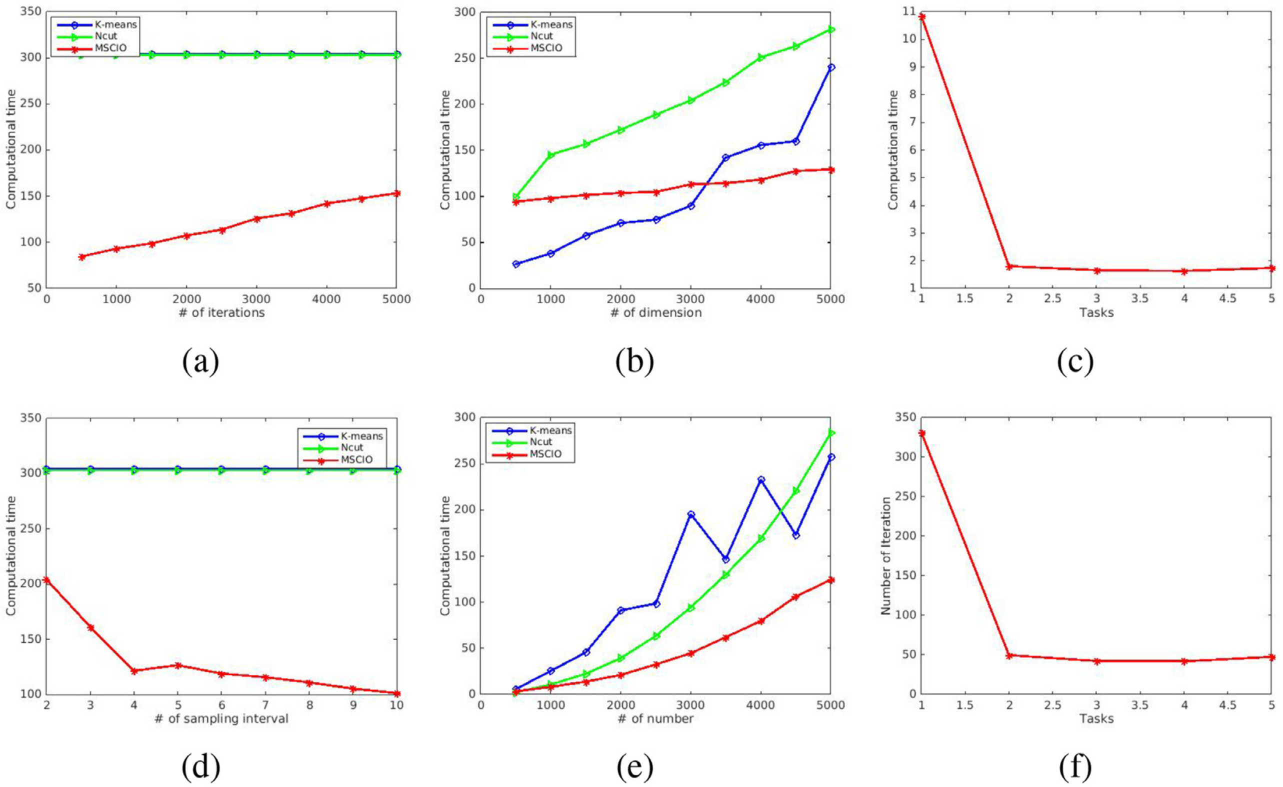
- Nonlinear manifold structure is also a significant cue for getting a good clustering result. Taking the results in Table 3 as an example, KM and FCM have limited ability for manifold data as they simply assign samples to its nearest cluster centroid, but ignore the global distribution. HC does well in dealing with the synthetic data sets, but it does not perform well in the task of the real-world data clustering, because HC is too much sensitive to the noises and the ambiguous samples. Ncut provides a clustering approach by making use of the spectrum of the affinity matrix and it remarkably improves the clustering effectiveness. SCIO achieves a similar performance as Ncut, as the utilization of affinity matrix and spectral graph theory.
- With the increase of data dimension, the running time of KM grows rapidly and Ncut also needs more time to calculate the affinity matrix of large-scale and high-dimensional data sets. KM achieves a good performance on Mnist data sets but it is not able to deal with high-dimensional data sets such as TDT2, RCV1, and Reuters21,578. Ncut is a bit better than KM with respect to CT on the experiments of the high-dimensional data sets, but it is not satisfactory. Comparing with the above approaches, SCIO provides a more efficient method. For the small-scale data sets, the running time of SCIO is similar to KM and Ncut, but it's much more efficient for large-scale and high-dimensional data.
- Compare with the single-task clustering, multi-task clustering provides a more efficient and effective technique for the large-scale data sets by dividing them into multiple related data sets and sharing the knowledge between related tasks. Taking the results in Table 4 as an example, mSCIO achieves a better performance than NysNcut because mSCIO divides the whole video into several consecutive frames. In this case, we only need to calculate the affinity matrix of the pixels in two consecutive frames but not the affinity matrix of the whole video. And this is the major reason that mSCIO is much more efficient than NysNcut. On the other hand, it allows us to take more pixels to approximate the affinity matrix because mSCIO does not need too much memory to store the affinity matrix of the whole video as NysNcut does, and NysNcut is out of memory when Sample Interval (SI) is less than 100.

4.5. Computational time comparison

We conduct experiments on a subset of Reuters21578 by controlling variables because the computational time is mainly influenced by four variables: iterations, sampling interval, sample dimension and sample size. Iterations is set to 5000, sampling interval is set to 5, and the subset of Reuters21,578 is 5000 samples with 5000 dimensions. We fix three variables to evaluate the remaining one. Fig. 2(a-b) and 2(d-e) present the comparison of k-means, Ncut and SCIO, and these methods are all implemented in their original formulation for impartiality. To illuminate the influence of multi-task constraint, the computational time and the iteration of five consecutive and related tasks are listed in Fig. 2(c) and 2(f).

After repeating each experiment for 20 runs, we make the following observations from the results in Fig. 2.

- SCIO takes much less time than k-means and Ncut under the same condition. As can be observed from the results in Fig. 2(a) and 2(d), SCIO needs only half of the time in dealing with large-scale and high-dimensional data set (5000 samples × 5000 dimensions), when comparing with k-means and Ncut. We further analyze the influence of sample dimension and sample size. According to the results in Fig. 2(b), the running time of k-means rapidly grows with the increase of sample dimension. Besides, Ncut is more sensitive to the increase of sample size because the operation of eigenvector computing has a computational complexity of $O(n^3)$ in general where n is the number of samples, as the observation in Fig. 2(c).
- In the experiment of multi-task clustering, it takes more than half of the time for convergence in the first task than the summery of the remaining tasks as can be observed from the results in Fig. 2(c) and 2(f). This is caused by the fact that the indicator matrix of the first task provides a coarse direction to optimize the following tasks by introduce an additional multi-task constraint as mentioned in Section 3.1.
- As shown in Fig. 3, the kernel width σ is sensitive to different data sets and the proposed adaptive σ (red point) provides a better performance. It was also suggested that $\sigma^2 =$



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 2. The runtime curves of multitask spectral clustering on three divergent dynamic datasets.

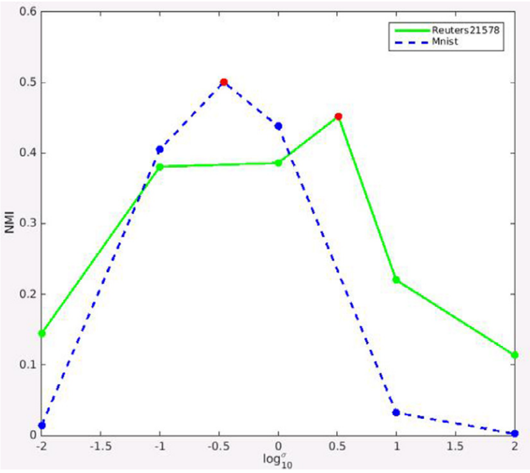


Fig. 3. Effect of parameter σ. We plot the change in NMI with different σ and the red points are the adaptive σ for different datasets. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

$\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}||[x_n]_i - [x_n]_j||_2^2$ which makes the values of the affinity matrix are around $e^{-1/2}$ provides a more robust data distribution than a fixed σ.

5. Conclusion

In this paper, we propose a multi-task spectral clustering, namely SCIO, based on iterative optimization. It provide an efficient and effective clustering technique in dealing with clustering on the large-scale and high-dimensional data sets. Moreover, It introduces a multi-task constraint to exploit the knowledge shared by multiple data sets and performs well on the multiple related tasks together. Several traditional methods including *k*-means and Ncut are compared and our algorithm outperform them in terms of normalized mutual information and computational time. Extensive experiments on both the synthetic data sets and the real-world data sets demonstrate that SCIO and mSCIO provide an efficient and effective solution for clustering on the large-scale and high-dimensional data sets and multi-task learning. Further research will include the extension of the proposed approaches to supervised case.

Acknowledgments

This work was supported by the State Key Program of National Natural Science Foundation of China under Grant 61632018, the National Key R\&D Program of China under Grant 2017YFB1002202, National Natural Science Foundation of China under Grant 61773316, Natural Science Foundation of Shaanxi Province under Grant 2018KJXX-024, Fundamental Research Funds for the Central Universities under Grant 3102017AX010, and the Open Research Fund of Key Laboratory of Spectral Imaging Technology, Chinese Academy of Sciences.

References

[1] P.F. Felzenszwalb, D.P. Huttenlocher, Efficient graph-based image segmentation, Int. J. Comput. Vision 59 (2) (2004) 167–181.

[2] L. Wang, M. Dong, Multi-level low-rank approximation-based spectral clustering for image segmentation, Pattern Recognit. Lett. 33 (16) (2012) 2206–2215.

[3] F. Nie, H. Huang, X. Cai, C.H.Q. Ding, Efficient and robust feature selection via joint l21-norms minimization, in: Neural Information Processing Systems, 2010, pp. 1813–1821.

[4] F. Nie, W. Zhu, X. Li, Unsupervised feature selection with structured graph optimization, in: Artificial Intelligence, 2016, pp. 1302–1308.

[5] Q. Wang, F. Zhang, X. Li, Optimal clustering framework for hyperspectral band selection, IEEE Trans. Geosci. Remote Sens. (2018), doi:10.1109/TGRS.2018.2828161.

[6] F. Nie, D. Xu, I.W. Tsang, C. Zhang, Flexible manifold embedding: a framework for semi-supervised and unsupervised dimension reduction, IEEE Trans. Image Process. 19 (7) (2010) 1921–1932.

[7] G. Gu, Z. Hou, C. Chen, Y. Zhao, A dimensionality reduction method based on structured sparse representation for face recognition, Artif. Intell. Rev. 46 (4) (2016) 431–443.

[8] J. Wu, H. Xiong, J. Chen, Towards understanding hierarchical clustering: a data distribution perspective, Neurocomputing 72 (10–12) (2009) 2319–2330.

[9] F. Murtagh, A survey of recent advances in hierarchical clustering algorithms, Comput. J. 26 (4) (1983) 354–359.

[10] A. Nagpal, A. Jatain, D. Gaur, Review based on data clustering algorithms, in: Proceedings of the IEEE Conference on Information & Communication Technologies (ICT), IEEE, 2013, pp. 298–303.

[11] J. Ye, Z. Zhao, M. Wu, Discriminative k-means for clustering, in: Advances in neural information processing systems, 2008, pp. 1649–1656.

[12] Q. Wang, J. Wan, Y. Yuan, Locality constraint distance metric learning for traffic congestion detection, Pattern Recognit. 75 (2018) 272–281.

[13] I.S. Dhillon, Y. Guan, B. Kulis, Weighted graph cuts without eigenvectors a multilevel approach, IEEE Trans. Pattern Anal. Mach. Intell. 29 (11) (2007) 1944–1957.

[14] Y. Chen, S. Sanghavi, H. Xu, Improved graph clustering, IEEE Trans. Inf. Theory 60 (10) (2014) 6440–6455.

[15] Q. Wang, J. Wan, Y. Yuan, Deep metric learning for crowdedness regression, in: Proceedings of the IEEE Transactions on Circuits and Systems for Video Technology, 2017, doi:10.1109/TCSVT.2017.2703920.

[16] C.C. Fowlkes, S.J. Belongie, F.R.K. Chung, J. Malik, Spectral grouping using the nyström method, IEEE Trans. Pattern Anal. Mach. Intell. 26 (2) (2004) 214–225.

[17] H. Liu, T. Liu, J. Wu, D. Tao, Y. Fu, Spectral ensemble clustering, in: Proceedings of the International Conference on Knowledge Discovery and Data Mining, 2015, pp. 715–724.

[18] R. Greenlaw, S. Kantabutra, Survey of clustering: algorithms and applications, Int. J. Inf. Retr. Res. 3 (2) (2013) 1–29.

[19] Q. Wang, M. Chen, X. Li, Quantifying and detecting collective motion by manifold learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2017, pp. 4292–4298.

[20] Q. Wang, J. Lin, Y. Yuan, Salient band selection for hyperspectral image classification via manifold ranking, IEEE Trans. Neural Netw. Learn. Syst. 27 (6) (2016) 1279–1289.

[21] A. Khoreva, F. Galasso, M. Hein, B. Schiele, Learning must-link constraints for video segmentation based on spectral clustering, in: Proceedings of the Pattern Recognition, 2014, pp. 701–712.

[22] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.

[23] A. Choromanska, T. Jebara, H. Kim, M. Mohan, C. Monteleoni, Fast spectral clustering via the nyström method, in: Proceedings of the International Conference on Algorithmic Learning Theory, 2013, pp. 367–381.

[24] S. Kumar, M. Mohri, A. Talwalkar, Sampling techniques for the nystrom method, in: Proceedings of the Artificial Intelligence and Statistics, 2009, pp. 304–311.

[25] J. Liu, C. Wang, M. Danilevsky, J. Han, Large-scale spectral clustering on graphs, in: Proceedings of the International Joint Conference on Artificial Intelligence, 2013, pp. 1486–1492.

[26] D. Yan, L. Huang, M.I. Jordan, Fast approximate spectral clustering, in: Proceedings of the International Conference on Knowledge Discovery and Data Mining, 2009, pp. 907–916.

[27] A.C. Türkmen, A review of nonnegative matrix factorization methods for clustering, CoRR (2015) abs/1507.03194.

[28] T. Liu, M. Gong, D. Tao, Large-cone nonnegative matrix factorization, IEEE Trans. Neural Netw. Learn. Syst. 28 (9) (2017) 2129–2142.

[29] W. Zhu, F. Nie, X. Li, Fast spectral clustering with efficient large graph construction, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 2017, pp. 2492–2496.

[30] X. Zhang, Convex discriminative multitask clustering, IEEE Trans. Pattern Anal. Mach. Intell. 37 (1) (2015) 28–40.

[31] T. Liu, D. Tao, M. Song, S.J. Maybank, Algorithm-dependent generalization bounds for multi-task learning, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2) (2017) 227–241.

[32] Y. Li, X. Tian, T. Liu, D. Tao, Multi-task model and feature joint learning, in: Proceedings of the International Joint Conference on Artificial Intelligence, 2015, pp. 3643–3649.

[33] Q. Gu, J. Zhou, Learning the shared subspace for multi-task clustering and transductive transfer classification, in: Proceedings of the IEEE International Conference on Data Mining, 2009, pp. 159–168.

[34] Y. Yang, Z. Ma, Y. Yang, F. Nie, H.T. Shen, Multitask spectral clustering by exploring intertask correlation, IEEE Trans. Cybernetics 45 (5) (2015) 1069–1080.

[35] J. Gallier, Spectral theory of unsigned and signed graphs. applications to graph clustering: a survey, CoRR (2016). abs/1601.04692.

[36] F. Nie, C.H.Q. Ding, D. Luo, H. Huang, Improved minmax cut graph clustering with nonnegative relaxation, in: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases, 2010, pp. 451–466.

[37] X. Li, G. Cui, Y. Dong, Graph regularized non-negative low-rank matrix factorization for image clustering, IEEE Trans. Cybernetics 47 (11) (2017) 3840–3853.

[38] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, in: Proceedings of the IEEE, 1998, pp. 2278–2324.

[39] D. Cai, X. Wang, X. He, Probabilistic dyadic data analysis with local and global consistency, in: Proceedings of the Machine Learning, 2009, pp. 105–112.

[40] D. Cai, Q. Mei, J. Han, C. Zhai, Modeling hidden topics on document manifold, in: Information and Knowledge Management, 2008, pp. 911–920.

[41] C. Apté, F. Damerau, S.M. Weiss, Automated learning of decision rules for text categorization, ACM Trans. Inf. Syst. 12 (3) (1994) 233–251.

[42] D. Cai, X. He, W.V. Zhang, J. Han, Regularized locality preserving indexing via spectral regression, in: Proceedings of the ACM Conference on Conference on Information and Knowledge Management (CIKM'07), 2007, pp. 741–750.

[43] D. Cai, X. He, J. Han, Document clustering using locality preserving indexing, IEEE Trans. Knowl. Data Eng. 17 (12) (2005) 1624–1637.

[44] Q. Zhan, Y. Mao, Improved spectral clustering based on nyström method, Multimedia Tools Appl. 76 (19) (2017) 20149–20165.

Yang Zhao is currently working toward the Ph.D. degree with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Xi'an Institute of Optics and Precision Mechanics of CAS, Xi'an, China and the University of Chinese Academy of Sciences, Beijing, China. His research interests include action recognition and video understanding.

Yuan Yuan (M'05-SM'09) is a Full Professor with the School of Computer Science and Center for OPTical IMagery Analysis and Learning(OPTIMAL), Northwestern Polytechnical University, Xi'an, China. She has authored over 150 papers, including about 100 in reputable journals such as the IEEE TRANSACTIONS and Pattern Recognition, and conference papers in the IEEE Conference on Computer Vision and Pattern Recognition, the British Machine Vision Conference, the IEEE International Conference on Image Processing, and the IEEE International Conference on Acoustics, Speech, and Signal Processing. Her research interests include visual information processing and image/video

Feiping Nie received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2009. He is currently a Full Professor with Northwestern Polytechnical University, Xi'an, China. He has authored over 100 papers in the prestigious journals and conferences, such as the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, the International Conference on Machine Learning, the Neural Information Processing Systems, and the Knowledge Discovery and Data Mining. His current research interests include machine learning and its application fields, such as pattern recognition, data mining, computer vision, image processing, and information retrieval. Dr. Nie is currently serving as an Associate Editor or a Program Committee Member for several prestigious journals and conferences in the related fields.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Computer Science, with the Unmanned System Research Institute, and with the Center for OPTical IMagery Analysis and Learning, Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.