Co-clustering on manifolds

Conference Paper · June 2009

DOI: 10.1145/1557019.1557063 · Source: DBLP

2 authors, including:

Quanquan Gu

University of Virginia

48 PUBLICATIONS **775** CITATIONS

Co-Clustering on Manifolds

Quanquan Gu
State Key Laboratory on Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology (TNList)
Department of Automation, Tsinghua University, Beijing, China, 100084
gqq03@mails.tsinghua.edu.cn

Jie Zhou
State Key Laboratory on Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology (TNList)
Department of Automation, Tsinghua University, Beijing, China, 100084
jzhou@tsinghua.edu.cn

ABSTRACT

Co-clustering is based on the duality between data points (e.g. documents) and features (e.g. words), i.e. data points can be grouped based on their distribution on features, while features can be grouped based on their distribution on the data points. In the past decade, several co-clustering algorithms have been proposed and shown to be superior to traditional one-side clustering. However, existing co-clustering algorithms fail to consider the geometric structure in the data, which is essential for clustering data on manifold. To address this problem, in this paper, we propose a Dual Regularized Co-Clustering (DRCC) method based on semi-nonnegative matrix tri-factorization. We deem that not only the data points, but also the features are sampled from some manifolds, namely data manifold and feature manifold respectively. As a result, we construct two graphs, i.e. data graph and feature graph, to explore the geometric structure of data manifold and feature manifold. Then our co-clustering method is formulated as semi-nonnegative matrix tri-factorization with two graph regularizers, requiring that the cluster labels of data points are smooth with respect to the data manifold, while the cluster labels of features are smooth with respect to the feature manifold. We will show that DRCC can be solved via alternating minimization, and its convergence is theoretically guaranteed. Experiments of clustering on many benchmark data sets demonstrate that the proposed method outperforms many state of the art clustering methods.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning; I.5.3 [**Pattern Recognition**]: Clustering

General Terms

Algorithms, Experimentations

KDD'09, June 28–July 1, 2009, Paris, France.

Keywords

Co-clustering, Data manifold, Feature manifold, Graph regularization, Semi-nonnegative matrix tri-factorization

1. INTRODUCTION

Clustering is one of the most fundamental topics in unsupervised machine learning and has been widely applied in data mining, computer vision, biology and so on. From a traditional view, clustering aims to divide the unlabeled data set into groups of similar data points. From a geometrical view, a data set can be seen as a set of discrete samplings on continuous manifold, and clustering aims at finding intrinsic structures of the manifold.

Many clustering methods have been proposed up to now, e.g. Kmeans [1], spectral clustering [21] [18] [15] and Nonnegative Matrix Factorization (NMF) [13] [23]. It is worth noting there is close connection between Kmeans, spectral clustering and NMF [24] [7] [10] [14].

However, the methods mentioned above focus on one-side clustering, i.e. clustering the data side based on the similarities along the feature side. Motivated by the duality between data points (e.g. documents) and features (e.g. words), i.e. data points can be grouped based on their distribution on features, while features can be grouped based on their distribution on the data points, several co-clustering algorithms have been proposed in the past decade and shown to be superior to traditional one-side clustering. For instance, [6] proposed a bipartite spectral graph partition approach to co-cluster words and documents. However, it requires that each document cluster is associated with a word cluster, which is a very tough restriction. [8] proposed an information theoretic co-clustering algorithm, which can be seen as the extension of information bottleneck method [20] to two-side clustering. [11] proposed an orthogonal nonnegative matrix tri-factorization (ONMTF) to co-cluster words and documents, which owns an elegant mathematical form and encouraging performance.

Recent studies show that many real world data are actually sampled from a nonlinear low dimensional manifold which is embedded in the high dimensional ambient space [17] [16]. Yet existing co-clustering algorithms [6] [8] [11] fail to consider the geometric structure in the data which is essential for clustering data on manifold. This greatly limits the application of co-clustering for the data lying on manifold.

To address this problem, in this paper, we propose a Dual Regularized Co-Clustering (DRCC) method based on semi-

nonnegative matrix tri-factorization, which inherits the advantages of ONMTF [11]. We deem that not only the data points but also the features are discrete samplings from some manifolds, namely data manifold and feature manifold respectively. Thus, we construct two graphs, i.e. data graph and feature graph, to explore the geometric structure of data manifold as well as feature manifold. We require that the cluster labels of data points are smooth with respect to the intrinsic data manifold, while the cluster labels of features are smooth with respect to the intrinsic feature manifold. This is achieved by graph regularization. Then DRCC is formulated as semi-nonnegative matrix tri-factorization with two graph regularizers. As a result, DRCC takes into account the geometric information of the data points and features, and is suitable for clustering data on manifold. We will show that DRCC can be optimized by iterative multiplicative updating algorithm and its convergence is theoretically guaranteed. Experiments of clustering on many benchmark data sets demonstrate that the proposed method outperforms many state of the art clustering methods.

The remainder of this paper is organized as follows. In Section 2 we will propose dual regularized co-clustering (DRCC) method, along with the optimization algorithm, followed with the proof of the convergence of the algorithm. In Section 3, we discuss several related works. The experiments on benchmark data sets are demonstrated in Section 4. Finally, we draw a conclusion and point out the future work in Section 5.

2. DUAL REGULARIZED CO-CLUSTERING

In this section, we first briefly introduce the formulation of co-clustering, and some notations frequently used in this paper. Then we present the graph regularization on both the data side and the feature side, followed which we present the dual regularized co-clustering (DRCC) method and its optimization algorithm. Finally, we prove the convergence of the algorithm.

2.1 Problem Formulation & Notations

In the setting of co-clustering, we are given a data set $\mathcal{X} = \{\mathbf{x}_{\cdot 1}, \ldots, \mathbf{x}_{\cdot n}\} \in \mathbb{R}^d$. The goal is to group the data points $\{\mathbf{x}_{\cdot 1}, \ldots, \mathbf{x}_{\cdot n}\}$ into c clusters $\{\mathcal{C}_j\}_{j=1}^c$, while group the features $\{\mathbf{x}_{1\cdot}, \ldots, \mathbf{x}_{d\cdot}\}$ into m clusters $\{\mathcal{W}_j\}_{j=1}^m$.

We use a partition matrix $\mathbf{F} \in \{0,1\}^{n\times c}$ to represent the clustering result of data points, such that $\mathbf{F}_{ij} = 1$ if $\mathbf{x}_{\cdot i}$ belongs to cluster $\mathcal{C}_j$ and $\mathbf{F}_{ij} = 0$ otherwise. This is also known as *hard* clustering, i.e. the cluster assignment is binary. Similarly, we use another partition matrix $\mathbf{G} \in \{0,1\}^{d\times m}$ to represent the clustering result of features.

For convenience, we present in Table 1 the important notations used in the rest of this paper.

2.2 Graph Regularization

As we have mentioned above, recent researches show that many real world data distribute on low-dimensional manifold embedded in the high-dimensional ambient space [17] [16]. However, existing co-clustering algorithms [6] [8] [11] fail to consider the geometric structure which is essential for clustering data on manifold. A natural treatment for the data sampled from a manifold is to construct a graph to discretely approximate the manifold, whose vertices correspond to the data samples, while the edge weight represents the affinity between the data points. One common assumption

Table 1: Important notations used in this paper.

Notation	Description
n	number of data points
d	number of features
c	number of data clusters
m	number of feature clusters
$\mathcal{X}$	data set
$\mathbf{X}$	data matrix of size $d \times n$
$\mathbf{x}_{i\cdot}$	ith row of $\mathbf{X}$
$\mathbf{x}_{\cdot i}$	ith column of $\mathbf{X}$
$\mathcal{N}(\cdot)$	k-nearest neighborhood
$\mathbf{F}$	data partition matrix of size $n \times c$
$\mathbf{f}_{i\cdot}$	ith row of $\mathbf{F}$
$\mathbf{f}_{\cdot i}$	ith column of $\mathbf{F}$
$\mathbf{G}$	feature partition matrix of size $d \times m$
$\mathbf{g}_{i\cdot}$	ith row of $\mathbf{G}$
$\mathbf{g}_{\cdot i}$	ith column of $\mathbf{G}$
$\mathcal{G}_F$	data graph
$\mathcal{G}_G$	feature graph
$\mathbf{W}^F$	data affinity matrix of size $n \times n$
$\mathbf{W}^G$	feature affinity matrix of size $d \times d$
$\mathbf{D}^F$	data degree matrix of size $n \times n$
$\mathbf{D}^G$	feature degree matrix of size $d \times d$
$\mathbf{L}_F$	data graph Laplacian of size $n \times n$
$\mathbf{L}_G$	feature graph Laplacian of size $d \times d$

about the affinity between data points is *Cluster Assumption* [4], which says if two samples are close to each other in the input space, then their labels (or embeddings) are also close to each other. This assumption has been widely used in spectral clustering [21] [18] [15], dimensionality reduction [16] [12] and semi-supervised learning [4] [25]. Furthermore, we deem that not only the data points are sampled from a manifold, namely data manifold, but also from the dual view, the features are discrete samplings from another manifold, namely feature manifold. As a result, we construct two graphs, i.e. data graph and feature graph, to explore the geometric structure of data manifold and feature manifold. In the following, we will introduce the construction of data graph and feature graph respectively.

2.2.1 Data Graph

We construct a data graph $\mathcal{G}_F$ whose vertices correspond to $\{\mathbf{x}_{\cdot 1}, \ldots, \mathbf{x}_{\cdot n}\}$. According to *Cluster Assumption*, if data points $\mathbf{x}_{\cdot i}$ and $\mathbf{x}_{\cdot j}$ are close to each other, then their cluster labels $\mathbf{f}_{i\cdot}$ and $\mathbf{f}_{j\cdot}$ should be close as well. This is formulated as follows,

$$\frac{1}{2}\sum_{ij} ||\mathbf{f}_{i\cdot} - \mathbf{f}_{j\cdot}||^2 W_{ij}^F \tag{1}$$

where W_{ij}^F is the affinity measuring how close $\mathbf{f}_{i\cdot}$ and $\mathbf{f}_{j\cdot}$ will be.

For simplicity, we define the data affinity matrix $\mathbf{W}^F$ as follows,

$$W_{ij}^F = \begin{cases} 1, & \text{if } \mathbf{x}_{\cdot j} \in \mathcal{N}(\mathbf{x}_{\cdot i}) \text{ or } \mathbf{x}_{\cdot i} \in \mathcal{N}(\mathbf{x}_{\cdot j}) \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

where $\mathcal{N}(\mathbf{x}_{\cdot i})$ denotes the k-nearest neighbor of $\mathbf{x}_{\cdot i}$. It has the advantage that there is no parameter to be tuned except the neighborhood size, i.e. k. Other kinds of affinity can also be adopted, e.g. heat kernel [12].

Eq.(1) can be further rewritten as

$$\begin{aligned}
&\frac{1}{2}\sum_{i,j}||\mathbf{f}_{i\cdot}-\mathbf{f}_{j\cdot}||^2W_{ij}^F \\
=&\sum_{i,j}\mathbf{f}_{i\cdot}W_{ij}^F\mathbf{f}_{i\cdot}^T-\sum_{i,j}\mathbf{f}_{i\cdot}W_{ij}^F\mathbf{f}_{j\cdot}^T \\
=&\sum_{i}\mathbf{f}_{i\cdot}D_{ii}^F\mathbf{f}_{i\cdot}^T-\sum_{i,j}\mathbf{f}_{i\cdot}W_{ij}^F\mathbf{f}_{j\cdot}^T \\
=&\mathrm{tr}(\mathbf{F}^T(\mathbf{D}^F-\mathbf{W}^F)\mathbf{F}) \\
=&\mathrm{tr}(\mathbf{F}^T\mathbf{L}_F\mathbf{F}) \qquad (3)
\end{aligned}$$

where $D_{ii}^F=\sum_j W_{ij}^F$ is the diagonal degree matrix, and $\mathbf{L}_F=\mathbf{D}^F-\mathbf{W}^F$ is the graph Laplacian [5] of the data graph $\mathcal{G}_F$. Eq.(3) reflects the label smoothness of the data points. The smoother the data labels are with respect to the underlying data manifold, the smaller the value of the data graph regularization in Eq.(3) will be.

2.2.2 Feature Graph

Similar with the construction of the data graph $\mathcal{G}_F$, we construct a feature graph $\mathcal{G}_G$ whose vertices correspond to $\{\mathbf{x}_{1\cdot},\ldots,\mathbf{x}_{d\cdot}\}$. According to *Cluster Assumption* again, if features $\mathbf{x}_{i\cdot}$ and $\mathbf{x}_{j\cdot}$ are near, then their cluster labels $\mathbf{g}_{i\cdot}$ and $\mathbf{g}_{j\cdot}$ should be near as well. This is formulated as follows

$$\frac{1}{2}\sum_{ij}||\mathbf{g}_{i\cdot}-\mathbf{g}_{j\cdot}||^2W_{ij}^G \qquad (4)$$

where W_{ij}^G is the affinity measuring how close $\mathbf{g}_{i\cdot}$ and $\mathbf{g}_{j\cdot}$ will be.

For simplicity, we also define the feature affinity matrix $\mathbf{W}^G$ as follows,

$$W_{ij}^G=\begin{cases}1, & \text{if } \mathbf{x}_{j\cdot}\in\mathcal{N}(\mathbf{x}_{i\cdot}) \text{ or } \mathbf{x}_{i\cdot}\in\mathcal{N}(\mathbf{x}_{j\cdot})\\ 0, & \text{otherwise.}\end{cases} \qquad (5)$$

where $\mathcal{N}(\mathbf{x}_{i\cdot})$ denotes the k-nearest neighbor of $\mathbf{x}_{i\cdot}$.

Eq.(4) can be further rewritten as

$$\begin{aligned}
&\frac{1}{2}\sum_{i,j}||\mathbf{g}_{i\cdot}-\mathbf{g}_{j\cdot}||^2W_{ij}^G \\
=&\mathrm{tr}(\mathbf{G}^T(\mathbf{D}^G-\mathbf{W}^G)\mathbf{G}) \\
=&\mathrm{tr}(\mathbf{G}^T\mathbf{L}_G\mathbf{G}) \qquad (6)
\end{aligned}$$

where $D_{ii}^G=\sum_j W_{ij}^G$ is the degree matrix, and $\mathbf{L}_G=\mathbf{D}^G-\mathbf{W}^G$ is the graph Laplacian of the feature graph $\mathcal{G}_G$. Eq.(6) reflects the label smoothness of the features. The smoother the feature labels are with respect to the underlying feature manifold, the smaller the value of the feature graph regularization in Eq.(6) will be.

2.3 Objective

Based on the two graph regularizers presented in Eq.(3) and Eq.(6), we propose a new co-clustering method, minimizing the following objective,

$$J_{DRCC} = ||\mathbf{X}-\mathbf{GSF}^T||_F^2+\lambda\mathrm{tr}(\mathbf{F}^T\mathbf{L}_F\mathbf{F})+\mu\mathrm{tr}(\mathbf{G}^T\mathbf{L}_G\mathbf{G}) \qquad (7)$$

where $\lambda,\mu\geq 0$ are regularization parameters balancing the reconstruction error of co-clustering in the first term and the label smoothness of the data points and features in the second and third terms. Since there are two graph regularizers in the objective, we call Eq.(7) *Dual Regularized Co-Clustering* (DRCC). When letting $\lambda=\mu=0$, DRCC degenerates to ordinary co-clustering method.

By its definition, the elements in $\mathbf{F}$ and $\mathbf{G}$ can only take binary values, which makes the minimization in Eq.(7) very difficult, therefore we relax $\mathbf{F}$ and $\mathbf{G}$ into continuous nonnegative domain. Then DRCC in Eq.(7) turns out to minimize,

$$\begin{aligned}
J_{DRCC} &= ||\mathbf{X}-\mathbf{GSF}^T||_F^2+\lambda\mathrm{tr}(\mathbf{F}^T\mathbf{L}_F\mathbf{F})+\mu\mathrm{tr}(\mathbf{G}^T\mathbf{L}_G\mathbf{G}) \\
&\text{s.t.} \quad \mathbf{G}\geq 0,\mathbf{F}\geq 0 \qquad (8)
\end{aligned}$$

where $\mathbf{S}$ is a matrix whose entries can take any signs. Note that Eq.(8) is a Dual Regularized Semi-Nonnegative Matrix Tri-Factorization (DRSNMTF). To make the objective in Eq.(8) lower bounded, we use L_2 normalization on columns of $\mathbf{F}$ and $\mathbf{G}$ in the optimization, and compensate the norms of $\mathbf{F}$ and $\mathbf{G}$ to $\mathbf{S}$.

2.4 Optimization

In the following, we will give the solution to Eq.(8). As we see, minimizing Eq.(8) is with respect to $\mathbf{S},\mathbf{F}$ and $\mathbf{G}$, and we cannot give a closed-form solution. We will present an alternating scheme to optimize the objective. In other words, we will optimize the objective with respect to one variable while fixing the other variables. This procedure repeats until convergence.

2.4.1 Computation of S

Optimizing Eq.(8) with respect to $\mathbf{S}$ is equivalent to optimizing

$$J_1=||\mathbf{X}-\mathbf{GSF}^T||_F^2 \qquad (9)$$

Setting $\frac{\partial J_1}{\partial\mathbf{S}}=0$ leads to the following updating formula

$$\mathbf{S}=(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{XF}(\mathbf{F}^T\mathbf{F})^{-1} \qquad (10)$$

2.4.2 Computation of F

Optimizing Eq.(8) with respect to $\mathbf{F}$ is equivalent to optimizing

$$\begin{aligned}
J_2 &= ||\mathbf{X}-\mathbf{GSF}^T||_F^2+\lambda\mathrm{tr}(\mathbf{F}^T\mathbf{L}_F\mathbf{F}) \\
&\text{s.t.} \quad \mathbf{F}\geq 0, \qquad (11)
\end{aligned}$$

For the constraint $\mathbf{F}\geq 0$, we cannot get a closed-form solution of $\mathbf{F}$. In the following, we will present an iterative multiplicative updating solution. We introduce the Lagrangian multiplier $\boldsymbol{\alpha}\in\mathbb{R}^{n\times c}$, thus the Lagrangian function is

$$L(\mathbf{F})=||\mathbf{X}-\mathbf{GSF}^T||_F^2+\lambda\mathrm{tr}(\mathbf{F}^T\mathbf{L}_F\mathbf{F})-\mathrm{tr}(\boldsymbol{\alpha}\mathbf{F}^T) \qquad (12)$$

Setting $\frac{\partial L(\mathbf{F})}{\partial\mathbf{F}}=0$, we obtain

$$\boldsymbol{\alpha}=2\lambda\mathbf{L}_F\mathbf{F}-2\mathbf{A}+2\mathbf{FB} \qquad (13)$$

where $\mathbf{A}=\mathbf{X}^T\mathbf{GS}$ and $\mathbf{B}=\mathbf{S}^T\mathbf{G}^T\mathbf{GS}$.

Using the Karush-Kuhn-Tucker condition [2] $\boldsymbol{\alpha}_{ij}\mathbf{F}_{ij}=0$, we get

$$[\lambda\mathbf{L}_F\mathbf{F}-\mathbf{A}+\mathbf{FB}]_{ij}\mathbf{F}_{ij}=0 \qquad (14)$$

Introduce $\mathbf{L}_F=\mathbf{L}_F^+-\mathbf{L}_F^-$, $\mathbf{A}=\mathbf{A}^+-\mathbf{A}^-$ and $\mathbf{B}=\mathbf{B}^+-\mathbf{B}^-$ where $\mathbf{A}_{ij}^+=(|\mathbf{A}_{ij}|+\mathbf{A}_{ij})/2$ and $\mathbf{A}_{ij}^-=(|\mathbf{A}_{ij}|-\mathbf{A}_{ij})/2$ [9], we obtain

$$[\lambda\mathbf{L}_F^+\mathbf{F}-\lambda\mathbf{L}_F^-\mathbf{F}-\mathbf{A}^++\mathbf{A}^-+\mathbf{FB}^+-\mathbf{FB}^-]_{ij}\mathbf{F}_{ij}=0 \qquad (15)$$

Eq.(15) leads to the following updating formula

$$\mathbf{F}_{ij} \leftarrow \mathbf{F}_{ij}\sqrt{\frac{[\lambda\mathbf{L}_F^-\mathbf{F}+\mathbf{A}^+ + \mathbf{F}\mathbf{B}^-]_{ij}}{[\lambda\mathbf{L}_F^+\mathbf{F}+\mathbf{A}^- + \mathbf{F}\mathbf{B}^+]_{ij}}} \tag{16}$$

2.4.3 Computation of G

Optimizing Eq.(8) with respect to $\mathbf{G}$ is equivalent to optimizing

$$\begin{aligned} J_3 &= ||\mathbf{X}-\mathbf{G}\mathbf{S}\mathbf{F}^T||_F^2 + \mu\text{tr}(\mathbf{G}^T\mathbf{L}_G\mathbf{G}) \\ &\text{s.t.} \quad \mathbf{G} \geq 0, \end{aligned} \tag{17}$$

Similar with the computation of $\mathbf{F}$, since $\mathbf{G} \geq 0$, we introduce the Lagrangian multiplier $\boldsymbol{\beta} \in \mathbb{R}^{d\times m}$, thus the Lagrangian function is

$$L(\mathbf{G}) = ||\mathbf{X}-\mathbf{G}\mathbf{S}\mathbf{F}^T||_F^2 + \mu\text{tr}(\mathbf{G}^T\mathbf{L}_G\mathbf{G}) - \text{tr}(\boldsymbol{\beta}\mathbf{G}^T) \tag{18}$$

Setting $\frac{\partial L(\mathbf{G})}{\partial \mathbf{G}} = 0$, we obtain

$$\boldsymbol{\beta} = 2\mu\mathbf{L}_G\mathbf{G} - 2\mathbf{P} + 2\mathbf{G}\mathbf{Q} \tag{19}$$

where $\mathbf{P} = \mathbf{X}\mathbf{F}\mathbf{S}^T$ and $\mathbf{Q} = \mathbf{S}\mathbf{F}^T\mathbf{F}\mathbf{S}^T$.

Using the Karush-Kuhn-Tucker complementarity condition [2] $\boldsymbol{\beta}_{ij}\mathbf{G}_{ij} = 0$, we get

$$[\mu\mathbf{L}_G\mathbf{G} - \mathbf{P} + \mathbf{G}\mathbf{Q}]_{ij}\mathbf{G}_{ij} = 0. \tag{20}$$

Introduce $\mathbf{L}_G = \mathbf{L}_G^+ - \mathbf{L}_G^-$, $\mathbf{P} = \mathbf{P}^+ - \mathbf{P}^-$ and $\mathbf{Q} = \mathbf{Q}^+ - \mathbf{Q}^-$, we obtain

$$[\mu\mathbf{L}_G^+\mathbf{G} - \mu\mathbf{L}_G^-\mathbf{G} - \mathbf{P}^+ + \mathbf{P}^- + \mathbf{G}\mathbf{Q}^+ - \mathbf{G}\mathbf{Q}^-]_{ij}\mathbf{G}_{ij} = 0. \tag{21}$$

Eq.(21) leads to the following updating formula

$$\mathbf{G}_{ij} \leftarrow \mathbf{G}_{ij}\sqrt{\frac{[\mu\mathbf{L}_G^-\mathbf{G} + \mathbf{P}^+ + \mathbf{G}\mathbf{Q}^-]_{ij}}{[\mu\mathbf{L}_G^+\mathbf{G} + \mathbf{P}^- + \mathbf{G}\mathbf{Q}^+]_{ij}}} \tag{22}$$

In summary, we present the iterative multiplicative updating algorithm of optimizing Eq.(8) in Algorithm 1.

Algorithm 1 Dual Regularized Co-Clustering

Input: $\mathbf{X}$, the number of data clusters c, the number of feature clusters m, regularization parameters λ, μ, maximum number of iterations T;
Output: Partitions $\mathbf{F} \in \mathbb{R}^{n\times c}$;
Initialize $\mathbf{F}$ and $\mathbf{G}$ using K-means;
while not convergent **and** $t \leq T$ **do**
  Compute $\mathbf{S} = (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}$;
  Update $\mathbf{F}_{ij} \leftarrow \mathbf{F}_{ij}\sqrt{\frac{[\lambda\mathbf{L}_F^-\mathbf{F}+\mathbf{A}^+ + \mathbf{F}\mathbf{B}^-]_{ij}}{[\lambda\mathbf{L}_F^+\mathbf{F}+\mathbf{A}^- + \mathbf{F}\mathbf{B}^+]_{ij}}}$;
  Update $\mathbf{G}_{ij} \leftarrow \mathbf{G}_{ij}\sqrt{\frac{[\mu\mathbf{L}_G^-\mathbf{G}+\mathbf{P}^+ + \mathbf{G}\mathbf{Q}^-]_{ij}}{[\mu\mathbf{L}_G^+\mathbf{G}+\mathbf{P}^- + \mathbf{G}\mathbf{Q}^+]_{ij}}}$;
end while

2.5 Convergence Analysis

In this section, we will investigate the convergence of Algorithm 1.

We use the auxiliary function approach [13] to prove the convergence of the algorithm. Here we first introduce the definition of auxiliary function [13].

Definition 2.1 *[13] $Z(h, h')$ is an auxiliary function for $F(h)$ if the conditions*

$$Z(h, h') \geq F(h), Z(h, h) = F(h),$$

are satisfied.

Lemma 2.2 *[13] If Z is an auxiliary function for F, then F is non-increasing under the update*

$$h^{(t+1)} = \arg\min_h Z(h, h^{(t)})$$

PROOF. $F(h^{(t+1)}) \leq Z(h^{(t+1)}, h^{(t)}) \leq Z(h^{(t)}, h^{(t)}) = F(h^{(t)})$ □

Lemma 2.3 *[9] For any nonnegative matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{k\times k}$, $\mathbf{S} \in \mathbb{R}^{n\times k}$,$\mathbf{S}' \in \mathbb{R}^{n\times k}$, and $\mathbf{A}$, $\mathbf{B}$ are symmetric, then the following inequality holds*

$$\sum_{i=1}^{n}\sum_{p=1}^{k}\frac{(\mathbf{A}\mathbf{S}'\mathbf{B})_{ip}\mathbf{S}_{ip}^2}{\mathbf{S}'_{ip}} \geq tr(\mathbf{S}^T\mathbf{A}\mathbf{S}\mathbf{B})$$

In the following, we will present 4 theorems, which guarantee the convergence of Algorithm 1.

Theorem 2.4 *Let*

$$J(\mathbf{F}) = tr(\lambda\mathbf{F}^T\mathbf{L}_F\mathbf{F} - 2\mathbf{A}\mathbf{F}^T + \mathbf{F}\mathbf{B}\mathbf{F}^T) \tag{23}$$

Then the following function

$$\begin{aligned} & Z(\mathbf{F}, \mathbf{F}') \\ = \;& \lambda\sum_{ij}\frac{(\mathbf{L}_F^+\mathbf{F}')_{ij}\mathbf{F}_{ij}^2}{\mathbf{F}'_{ij}} - \lambda\sum_{ijk}(\mathbf{L}_F^-)_{jk}\mathbf{F}'_{ji}\mathbf{F}'_{ki}(1+\log\frac{\mathbf{F}_{ji}\mathbf{F}_{ki}}{\mathbf{F}'_{ji}\mathbf{F}'_{ki}}) \\ - \;& 2\sum_{ij}\mathbf{A}_{ij}^+\mathbf{F}'_{ij}(1+\log\frac{\mathbf{F}_{ij}}{\mathbf{F}'_{ij}}) + 2\sum_{ij}\mathbf{A}_{ij}^-\frac{\mathbf{F}_{ij}^2+\mathbf{F}_{ij}'^2}{2\mathbf{F}'_{ij}} \\ + \;& \sum_{ij}\frac{(\mathbf{F}'\mathbf{B}^+)_{ij}\mathbf{F}_{ij}^2}{\mathbf{F}'_{ij}} - \sum_{ijk}\mathbf{B}_{jk}^-\mathbf{F}'_{ij}\mathbf{F}'_{ik}(1+\log\frac{\mathbf{F}_{ij}\mathbf{F}_{ik}}{\mathbf{F}'_{ij}\mathbf{F}'_{ik}}) \end{aligned}$$

is an auxiliary function for $J(\mathbf{F})$. Furthermore, it is a convex function in $\mathbf{F}$ and its global minimum is

$$\mathbf{F}_{ij} = \mathbf{F}_{ij}\sqrt{\frac{[\lambda\mathbf{L}_F^-\mathbf{F}+\mathbf{A}^+ + \mathbf{F}\mathbf{B}^-]_{ij}}{[\lambda\mathbf{L}_F^+\mathbf{F}+\mathbf{A}^- + \mathbf{F}\mathbf{B}^+]_{ij}}} \tag{24}$$

PROOF. See Appendix A. □

Theorem 2.5 *Updating $\mathbf{F}$ using Eq.(16) will monotonically decrease the value of the objective in Eq.(8), hence it converges.*

PROOF. By Lemma 2.2 and Theorem 2.4, we can get that $J(\mathbf{F}^0) = Z(\mathbf{F}^0, \mathbf{F}^0) \geq Z(\mathbf{F}^1, \mathbf{F}^0) \geq J(\mathbf{F}^1) \geq \ldots$ So $J(\mathbf{F})$ is monotonically decreasing. Since $J(\mathbf{F})$ is obviously bounded below, we prove this theorem. □

Theorem 2.6 *Let*

$$J(\mathbf{G}) = tr(\mu\mathbf{G}^T\mathbf{L}_G\mathbf{G} - 2\mathbf{G}^T\mathbf{P} + \mathbf{G}\mathbf{Q}\mathbf{G}^T) \tag{25}$$

Then the following function

$$\begin{aligned} & Z(\mathbf{G}, \mathbf{G}') \\ = \;& \mu\sum_{ij}\frac{(\mathbf{L}_G^+\mathbf{G}')_{ij}\mathbf{G}_{ij}^2}{\mathbf{G}'_{ij}} - \mu\sum_{ijk}(\mathbf{L}_G^-)_{jk}\mathbf{G}'_{ji}\mathbf{G}'_{ki}(1+\log\frac{\mathbf{G}_{ji}\mathbf{G}_{ki}}{\mathbf{G}'_{ji}\mathbf{G}'_{ki}}) \\ - \;& 2\sum_{ij}\mathbf{P}_{ij}^+\mathbf{G}'_{ij}(1+\log\frac{\mathbf{G}_{ij}}{\mathbf{G}'_{ij}}) + 2\sum_{ij}\mathbf{P}_{ij}^-\frac{\mathbf{G}_{ij}^2+\mathbf{G}_{ij}'^2}{2\mathbf{G}'_{ij}} \\ + \;& \sum_{ij}\frac{(\mathbf{G}'\mathbf{Q}^+)_{ij}\mathbf{G}_{ij}^2}{\mathbf{G}'_{ij}} - \sum_{ijk}\mathbf{Q}_{jk}^-\mathbf{G}'_{ij}\mathbf{G}'_{ik}(1+\log\frac{\mathbf{G}_{ij}\mathbf{G}_{ik}}{\mathbf{G}'_{ij}\mathbf{G}'_{ik}}) \end{aligned}$$

is an auxiliary function for $J(\mathbf{G})$. Furthermore, it is a convex function in $\mathbf{G}$ and its global minimum is

$$\mathbf{G}_{ij} = \mathbf{G}_{ij}\sqrt{\frac{[\mu\mathbf{L}_G^-\mathbf{G} + \mathbf{P}^+ + \mathbf{G}\mathbf{Q}^-]_{ij}}{[\mu\mathbf{L}_G^+\mathbf{G} + \mathbf{P}^- + \mathbf{G}\mathbf{Q}^+]_{ij}}} \tag{26}$$

PROOF. See Appendix B. □

Theorem 2.7 *Updating $\mathbf{G}$ using Eq.(22) will monotonically decrease the value of the objective in Eq.(8), hence it converges.*

PROOF. By Lemma 2.2 and Theorem 2.6, we can get that $J(\mathbf{G}^0) = Z(\mathbf{G}^0, \mathbf{G}^0) \geq Z(\mathbf{G}^1, \mathbf{G}^0) \geq J(\mathbf{G}^1) \geq \ldots$ So $J(\mathbf{G})$ is monotonically decreasing. Since $J(\mathbf{G})$ is obviously bounded below, we prove this theorem. □

According to Theorem 2.5 and Theorem 2.7, Algorithm 1 is guaranteed to converge. Note that there is no guarantee that Algorithm 1 will converge to global optimum.

3. RELATED WORKS

In this section, we will review several works related with ours, and compare our method with them.

Given a nonnegative data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}_+^{d\times n}$, NMF [13] aims to find two nonnegative matrices $\mathbf{S} \in \mathbb{R}_+^{d\times c}$ and $\mathbf{F} \in \mathbb{R}_+^{n\times c}$ which minimize the following objective

$$\begin{aligned} J_{NMF} &= ||\mathbf{X} - \mathbf{S}\mathbf{F}^T||_F^2, \\ &\text{s.t.} \quad \mathbf{S} \geq 0, \mathbf{F} \geq 0, \end{aligned} \tag{27}$$

Note that in Eq.(27) $\mathbf{X}$ is a nonnegative constant matrix. This limits the application of NMF for general data with mixed signs.

[9] proposed a Semi-NMF, which relaxes the nonnegative constraint $\mathbf{S} \geq 0$ in Eq.(27) and hence is suitable for general data. It minimizes the following objective

$$\begin{aligned} J_{SNMF} &= ||\mathbf{X} - \mathbf{S}\mathbf{F}^T||_F^2, \\ &\text{s.t.} \quad \mathbf{F} \geq 0, \end{aligned} \tag{28}$$

Note that in Eq.(28) $\mathbf{X}$ is a constant matrix whose entries can take any signs.

The most related works with ours is [3] and [11].

In [3], the authors proposed a graph regularized NMF (GNMF), which adds an additional graph regularizer on NMF, imposing *Cluster Assumption* on the data points. It minimizes the following objective

$$\begin{aligned} J_{GNMF} &= ||\mathbf{X} - \mathbf{S}\mathbf{F}^T||_F^2 + \lambda\text{tr}(\mathbf{F}^T\mathbf{L}_F\mathbf{F}), \\ &\text{s.t.} \quad \mathbf{S} \geq 0, \mathbf{F} \geq 0, \end{aligned} \tag{29}$$

Hence GNMF can take into account the geometric information of the data.

In [11], the authors proposed an Orthogonal Nonnegative Matrix Tri-Factorization (ONMTF) to co-cluster words and documents, aiming to find three nonnegative matrices $\mathbf{G} \in \mathbb{R}^{d\times m}$, $\mathbf{S} \in \mathbb{R}^{m\times c}$ and $\mathbf{F} \in \mathbb{R}^{n\times c}$ which minimizes the following objective

$$\begin{aligned} J_{ONMTF} &= ||\mathbf{X} - \mathbf{G}\mathbf{S}\mathbf{F}^T||_F^2, \\ &\text{s.t.} \quad \mathbf{G} \geq 0, \mathbf{S} \geq 0, \mathbf{F} \geq 0, \\ &\qquad \mathbf{G}^T\mathbf{G} = \mathbf{I}_m, \mathbf{F}^T\mathbf{F} = \mathbf{I}_c \end{aligned} \tag{30}$$

where $\mathbf{I}_m \in \mathbb{R}^{m\times m}$ and $\mathbf{I}_c \in \mathbb{R}^{c\times c}$ are identity matrices.

DRCC not only considers the geometric structure in the data points as in GNMF, but also takes into account the geometric information in the features. In addition, our method relaxes the nonnegative constraint on $\mathbf{S}$ which is imposed in GNMF and ONMTF. As a result, DRCC applies for general data, while both GNMF and ONMTF are restricted to nonnegative data. Furthermore, the orthogonality constraints on $\mathbf{F}$ and $\mathbf{G}$ which are imposed in ONMTF are omitted in our method, since we use L_2 normalization on columns of $\mathbf{F}$ and $\mathbf{G}$ in the optimization, and compensate the norms of $\mathbf{F}$ and $\mathbf{G}$ to $\mathbf{S}$.

4. EXPERIMENTS

In this section, we will evaluate the performance of the proposed method. We compare our method with Kmeans, Normalized Cut (NCut) [18], NMF [13], Semi-NMF (SNMF) [9], ONMTF [11] and GNMF [3]. In order to verify our assumption that features also lie on a manifold, we test a special case of the proposed method with $\mu = 0$, denoted by RCC, and compare it with DRCC.

4.1 Evaluation Metrics

To evaluate the clustering results, we adopt the performance measures used in [3]. These performance measures are the standard measures widely used for clustering.

Clustering Accuracy Clustering Accuracy discovers the one-to-one relationship between clusters and classes and measures the extent to which each cluster contained data points from the corresponding class. Clustering Accuracy is defined as follows:

$$Acc = \frac{\sum_{i=1}^n \delta(map(r_i), l_i)}{n}, \tag{31}$$

where r_i denotes the cluster label of $\mathbf{x}_i$, and l_i denotes the true class label, n is the total number of documents, $\delta(x, y)$ is the delta function that equals one if $x = y$ and equals zero otherwise, and $map(r_i)$ is the permutation mapping function that maps each cluster label r_i to the equivalent label from the data set.

Normalized Mutual Information The second measure is the Normalized Mutual Information (NMI), which is used for determining the quality of clusters. Given a clustering result, the NMI is estimated by

$$NMI = \frac{\sum_{i=1}^c \sum_{j=1}^c n_{i,j} \log \frac{n_{i,j}}{n_i \hat{n}_j}}{\sqrt{(\sum_{i=1}^c n_i \log \frac{n_i}{n})(\sum_{j=1}^c \hat{n}_j \log \frac{\hat{n}_j}{n})}}, \tag{32}$$

where n_i denotes the number of data contained in the cluster $\mathcal{C}_i (1 \leq i \leq c)$, $\hat{n}_j$ is the number of data belonging to the $\mathcal{L}_j (1 \leq j \leq c)$, 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.

4.2 Data Sets

In our experiment, we use 6 data sets which are widely used as benchmark data sets in clustering literature [3] [11].

Coil20[1] This data set contains 32×32 gray scale images of 20 3D objects viewed from varying angles. For each object there are 72 images.

[1] http://www1.cs.columbia.edu/CAVE/software/softlib/coil-20.php

PIE The CMU PIE face database [19] contains 68 individuals with 41368 face images as a whole. The face images were captured by 13 synchronized cameras and 21 flashes, under varying pose, illumination and expression. All the images were also resized to 32×32.

CSTR This is the data set of the abstracts of technical reports published in the Department of Computer Science at a university. The data set contained 476 abstracts, which were divided into four research areas: Natural Language Processing (NLP), Robotics/Vision, Systems and Theory.

Newsgroup4 The Newsgroup4 data set used in our experiments is selected from the famous 20-newsgroups data set[2]. The topic *rec* containing *autos, motorcycles, baseball* and *hockey* was selected from the version 20news-18828. The Newsgroup4 data set contains 3970 documents.

WebKB4 The WebKB dataset contains webpages gathered from university computer science departments. There are about 8280 documents and they are divided into 7 categories: student, faculty, staff, course, project, department and other, among which student, faculty, course and project are four most populous entity-representing categories.

WebACE The data set contains 2340 documents consisting of news articles from Reuters new service via the Web in October 1997. These documents are divided into 20 classes.

Table.2 summarizes the characteristics of the data sets used in this experiment.

Table 2: Description of the data sets

Data Sets	#samples	#features	#classes
Coil20	1440	1024	20
PIE	1428	1024	68
CSTR	476	1000	4
Newsgroup4	3970	1000	4
WebKB4	4199	1000	4
WebACE	2340	1000	20

4.3 Parameter Settings

Since each clustering algorithm has one or more parameters to be tuned, in order to compare these algorithms fairly, we run these algorithms under different parameter settings, and select the best average result to compare with each other. We set the number of clusters equal to the true number of classes for all the data sets and clustering algorithms.

For NCut [18], the scale parameter of Gaussian kernel for constructing adjacency matrix is set by the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}$.

For ONMTF, the number of word clusters is set to be the same as the number of document clusters, i.e. the true number of classes in our experiment, according to [11].

For GNMF, the neighborhood size to construct the graph is set by searching the grid $\{1, 2, 3, \ldots, 10\}$ according to [3], and the regularization parameter is set by the grid $\{0.1, 1, 10, 100, 500, 1000\}$.

For DRCC, the number of data clusters is set the same as the number of feature clusters, i.e. the true number of classes, as in ONMTF. And for simplicity, the neighborhood size of the data graph is set to be the same as that of the feature graph, i.e. k, which is tuned by searching the grid $\{1, 2, 3, \ldots, 10\}$. We also set $\lambda = \mu$ and λ is tuned by searching the grid $\{0.1, 1, 10, 100, 500, 1000\}$. So the parameters of DRCC is tuned roughly. Better parameter tuning would achieve better clustering performance than that reported in this paper.

The parameter setting of RCC is the same as DRCC, except keeping $\mu = 0$.

Note that no parameter selection is needed for Kmeans, NMF and Semi-NMF, given the number of clusters.

Under each parameter setting of each method mentioned above, we repeat clustering 20 times, and the average result is computed. And we report the best average result for each method.

4.4 Clustering Results

The best average results are shown in Table 3 and Table 4. Table 3 shows the clustering accuracy of all the algorithms on all the data sets, while Table 4 shows the normalized mutual information.

We can see that DRCC outperforms the other clustering methods on all the data sets. The superiority of DRCC arises in the following two aspects: (1) co-clustering the features and data points together, and the clustering of features can lead to improvement in the clustering of data points; (2) exploration of the geometric structure in the data points as well as in the features, which is essential for clustering data on manifold. In addition, DRCC outperforms RCC on all the data sets except CSTR. This indicates considering the geometric structure in the features can further improve the clustering results at most cases, and verifies our assumption that features also lie on a manifold. Besides, ONMTF and GNMF usually achieve encouraging results, which further strengthens the advantages of co-clustering features and data points simultaneously, and considering the geometric structure in the data. Note that DRCC owns all these advantages.

4.5 Study on the Neighborhood Size

In this subsection, we will investigate the sensitivity with respect to the neighborhood size k. When we vary the value of k, we keep the other parameters fixed at the optimal value. We plot the clustering accuracy with respect to k in Figure 1.

As we can see, DRCC is a little sensitive to the neighborhood size of the graph. Fortunately, it usually achieves good result when the neighborhood size is large enough, e.g. $k = 10$ in our experiments.

4.6 Study on the Regularization Parameter

Next, we will investigate the sensitivity with respect to the regularization parameter λ $(= \mu)$. When we vary the value of λ, we keep the other parameters fixed at the optimal value. We plot the clustering accuracy with respect to λ in Figure 2.

We can see that DRCC is very stable with respect to the regularization parameter. It achieves consistent good result with the regularization parameter varying from 100 to 1000.

In summary, we may set $k = 10$ and $\lambda = \mu = 500$ in application for simplicity.

5. CONCLUSIONS AND FUTURE WORKS

In this paper, we propose a Dual Regularized Co-Clustering (DRCC) method based on semi-nonnegative matrix tri-factorization

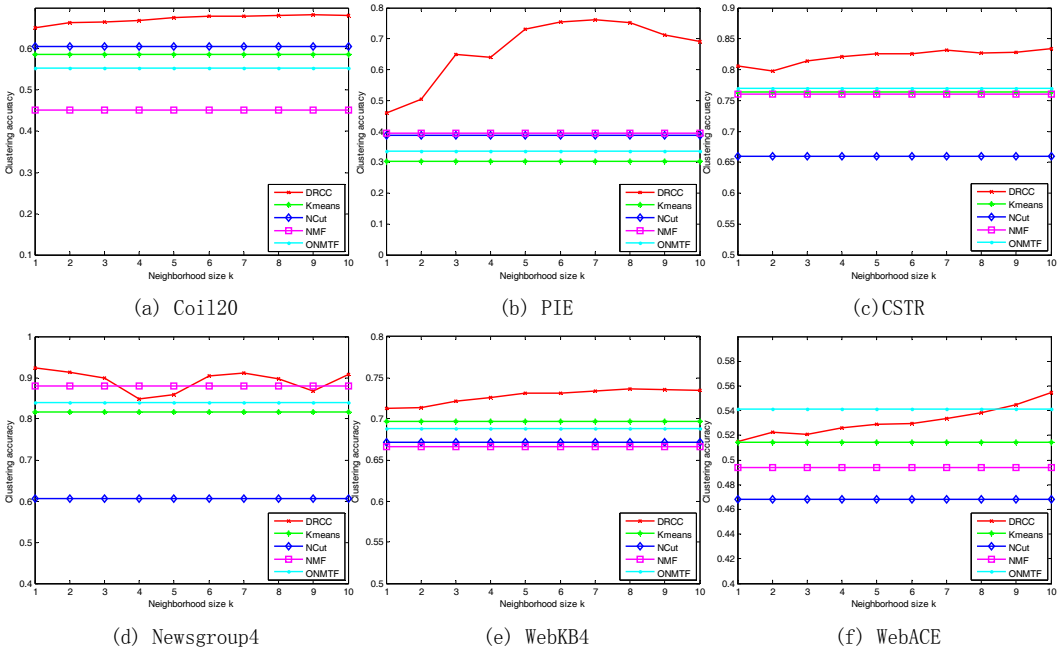
[2]http://people.csail.mit.edu/jrennie/20Newsgroups/

Table 3: Clustering Accuracy on the 6 data sets.

Data Sets	Kmeans	NCut	NMF	SNMF	ONMTF	GNMF	DRCC	RCC
Coil20	0.5864	0.6056	0.4517	0.3678	0.5527	0.6665	**0.6938**	0.6215
PIE	0.3018	0.3880	0.3952	0.2975	0.3351	0.7583	**0.7624**	0.7069
CSTR	0.7634	0.6597	0.7597	0.6976	0.7700	0.7437	0.8341	**0.8640**
Newsgroup4	0.8158	0.6056	0.8805	0.8214	0.8399	0.8877	**0.9240**	0.8817
WebKB4	0.6973	0.6716	0.6659	0.6214	0.6885	0.7264	**0.7361**	0.7130
WebACE	0.5142	0.4679	0.4936	0.4007	0.5415	0.5047	**0.5549**	0.5536

Table 4: Normalized Mutual Information on the 6 data sets.

Data Sets	Kmeans	NCut	NMF	SNMF	ONMTF	GNMF	DRCC	RCC
Coil20	0.7588	0.7407	0.5954	0.4585	0.7110	0.8136	**0.8822**	0.7907
PIE	0.6276	0.6843	0.6743	0.5430	0.6787	0.9368	**0.9377**	0.8078
CSTR	0.6531	0.5761	0.6645	0.5941	0.6716	0.6302	0.6923	**0.7167**
Newsgroup4	0.7129	0.7212	0.7294	0.6432	0.7053	0.7106	**0.7725**	0.7488
WebKB4	0.4665	0.4437	0.4255	0.3643	0.4552	0.4571	**0.4855**	0.4798
WebACE	0.6157	0.5959	0.5850	0.4649	0.6012	0.6007	**0.6244**	0.5849



(a) Coil20 (b) PIE (c) CSTR

(d) Newsgroup4 (e) WebKB4 (f) WebACE

Figure 1: Clustering accuracy with respect to the neighborhood size k.

with two graph regularizers, requiring that the cluster labels of data points are smooth with respect to the intrinsic data manifold, while the cluster labels of features are smooth with respect to the intrinsic feature manifold. DRCC is solved via alternating minimization, and its convergence is theoretically guaranteed. Experiments of clustering on many benchmark data sets demonstrate that the proposed method outperforms many state of the art clustering methods.

In our future work, we will investigate other kind of affinity in the graph regularization, e.g. *Local Learning Assumption* [22], which says the cluster label of each sample can be predicted by the samples in its neighborhood.

6. ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No.60721003, No.60673106 and No.60573062) and the Specialized Research Fund for the Doctoral Program of Higher Education. We thank the anonymous reviewers for their helpful comments.

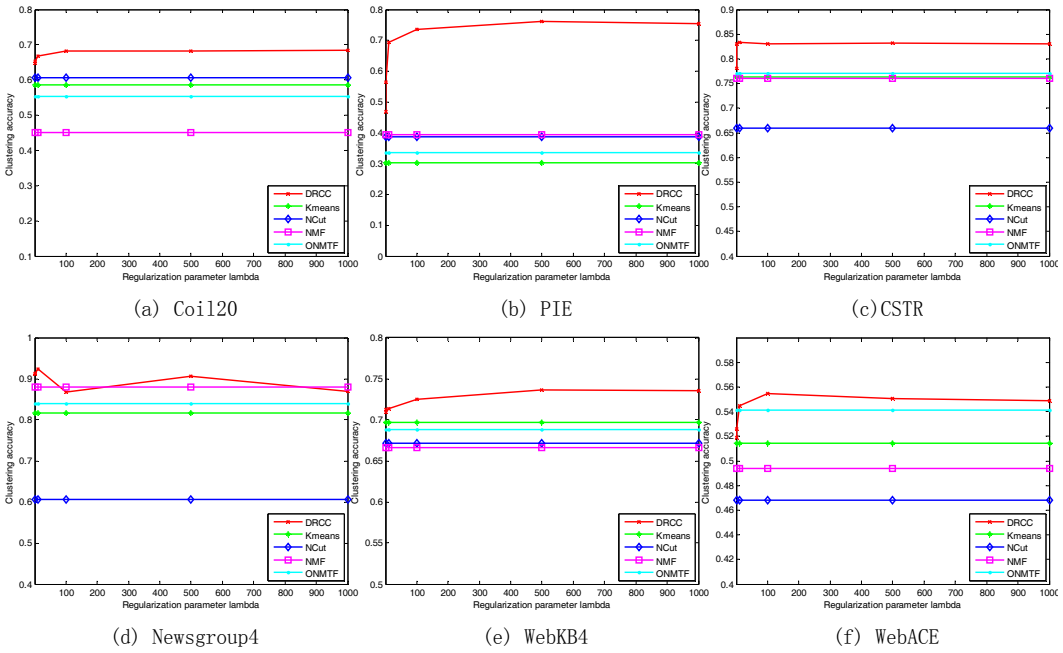
7. REFERENCES

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, Cambridge, 2004.

[3] D. Cai, X. He, X. Wu, and J. Han. Non-negative matrix factorization on manifold. In *ICDM*, 2008.

[4] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.



(a) Coil20 (b) PIE (c)CSTR

(d) Newsgroup4 (e) WebKB4 (f) WebACE

Figure 2: Clustering accuracy with respect to the regularization parameter λ.

[5] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, February 1997.

[6] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *KDD*, pages 269–274, 2001.

[7] I. S. Dhillon, Y. Guan, and B. Kulis. Kernel k-means: spectral clustering and normalized cuts. In *KDD*, pages 551–556, 2004.

[8] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretic co-clustering. In *KDD*, pages 89–98, 2003.

[9] C. H. Ding, T. Li, and M. I. Jordan. Convex and semi-nonnegative matrix factorizations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(1), 2008.

[10] C. H. Q. Ding and X. He. On the equivalence of nonnegative matrix factorization and spectral clustering. In *SDM*, 2005.

[11] C. H. Q. Ding, T. Li, W. Peng, and H. Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *KDD*, pages 126–135, 2006.

[12] X. He and P. Niyogi. Locality preserving projections. In *NIPS*, 2003.

[13] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[14] T. Li and C. H. Q. Ding. The relationships among various nonnegative matrix factorization methods for clustering. In *ICDM*, pages 362–371, 2006.

[15] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, pages 849–856, 2001.

[16] P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373–1396, 2003.

[17] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 2000.

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

[19] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination, and expression database. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(12):1615–1618, 2003.

[20] N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. *CoRR*, physics/0004057, 2000.

[21] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[22] M. Wu and B. Schölkopf. A local learning approach for clustering. In *NIPS*, pages 1529–1536, 2006.

[23] Xu, Wei, Liu, Xin, and Gong, Yihong. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 267–273, 2003.

[24] H. Zha, X. He, C. H. Q. Ding, M. Gu, and H. D. Simon. Spectral relaxation for k-means clustering. In *NIPS*, pages 1057–1064, 2001.

[25] X. Zhu. Semi-supervised learning literature survey. Technical report, Computer Sciences, University of Wisconsin-Madison, 2008.

APPENDIX

A. PROOF OF THEOREM 2.4

PROOF. We rewrite Eq.(23) as

$$\begin{aligned} L(\mathbf{F}) &= \mathrm{tr}(\lambda\mathbf{F}^T\mathbf{L}_F^+\mathbf{F} - \lambda\mathbf{F}^T\mathbf{L}_F^-\mathbf{F} \\ &- 2\mathbf{F}^T\mathbf{A}^+ + 2\mathbf{F}^T\mathbf{A}^- + \mathbf{F}\mathbf{B}^+\mathbf{F}^T - \mathbf{F}\mathbf{B}^-\mathbf{F}^T) \quad (33) \end{aligned}$$

By applying Lemma 2.3, we have

$$\mathbf{tr}(\mathbf{F}^T\mathbf{L}_F^+\mathbf{F}) \leq \sum_{ij}\frac{(\mathbf{L}_F^+\mathbf{F}')_{ij}\mathbf{F}_{ij}^2}{\mathbf{F}'_{ij}}$$

$$\mathbf{tr}(\mathbf{F}\mathbf{B}^+\mathbf{F}^T) \leq \sum_{ij}\frac{(\mathbf{F}'\mathbf{B}^+)_{ij}\mathbf{F}_{ij}^2}{\mathbf{F}'_{ij}}$$

Moreover, by the inequality $a \leq \frac{(a^2+b^2)}{2b}, \forall a, b > 0$, we have

$$\mathrm{tr}(\mathbf{F}^T\mathbf{A}^-) = \sum_{ij}\mathbf{A}_{ij}^-\mathbf{F}_{ij} \leq \sum_{ij}\mathbf{A}_{ij}^-\frac{\mathbf{F}_{ij}^2+\mathbf{F}_{ij}'^2}{2\mathbf{F}'_{ij}}$$

To obtain the lower bound for the remaining terms, we use the inequality that $z \geq 1+\log z, \forall z > 0$, then

$$\begin{aligned}
\mathrm{tr}(\mathbf{F}^T\mathbf{A}^+) &\geq \sum_{ij}\mathbf{A}_{ij}^+\mathbf{F}'_{ij}(1+\log\frac{\mathbf{F}_{ij}}{\mathbf{F}'_{ij}}) \\
\mathrm{tr}(\mathbf{F}^T\mathbf{L}_F^-\mathbf{F}) &\geq \sum_{ijk}(\mathbf{L}_F^-)_{jk}\mathbf{F}'_{ji}\mathbf{F}'_{ki}(1+\log\frac{\mathbf{F}_{ji}\mathbf{F}_{ki}}{\mathbf{F}'_{ji}\mathbf{F}'_{ki}}) \\
\mathrm{tr}(\mathbf{F}\mathbf{B}^-\mathbf{F}^T) &\geq \sum_{ijk}\mathbf{B}_{jk}^-\mathbf{F}'_{ij}\mathbf{F}'_{ik}(1+\log\frac{\mathbf{F}_{ij}\mathbf{F}_{ik}}{\mathbf{F}'_{ij}\mathbf{F}'_{ik}})
\end{aligned}$$

By summing over all the bounds, we can get $\mathbf{Z}(\mathbf{F},\mathbf{F}')$, which obviously satisfies (1) $\mathbf{Z}(\mathbf{F},\mathbf{F}') \geq J_{DRCC}(\mathbf{F})$; (2)$\mathbf{Z}(\mathbf{F},\mathbf{F}) = J_{DRCC}(\mathbf{F})$

To find the minimum of $\mathbf{Z}(\mathbf{F},\mathbf{F}')$, we take

$$\begin{aligned}
\frac{\partial Z(\mathbf{F},\mathbf{F}')}{\partial \mathbf{F}_{ij}} &= 2\lambda\frac{(\mathbf{L}_F^+\mathbf{F}')_{ij}\mathbf{F}_{ij}}{\mathbf{F}'_{ij}} - 2\lambda(\mathbf{L}_F^-\mathbf{F}')_{ij}\frac{\mathbf{F}'_{ij}}{\mathbf{F}_{ij}} \\
&- 2\mathbf{A}_{ij}^+\frac{\mathbf{F}'_{ij}}{\mathbf{F}_{ij}} + 2\mathbf{A}_{ij}^-\frac{\mathbf{F}_{ij}}{\mathbf{F}'_{ij}} \\
&+ 2\frac{(\mathbf{F}'\mathbf{B}^+)_{ij}\mathbf{F}_{ij}}{\mathbf{F}'_{ij}} - 2(\mathbf{F}'\mathbf{B}^-)_{ij}\frac{\mathbf{F}'_{ij}}{\mathbf{F}_{ij}}
\end{aligned}$$

and the Hessian matrix of $Z(\mathbf{F},\mathbf{F}')$

$$\begin{aligned}
\frac{\partial^2 Z(\mathbf{F},\mathbf{F}')}{\partial \mathbf{F}_{ij}\partial \mathbf{F}_{kl}} &= \delta_{ik}\delta_{jl}(2\lambda\frac{(\mathbf{L}_F^+\mathbf{F}')_{ij}}{\mathbf{F}'_{ij}} + 2\lambda(\mathbf{L}_F^-\mathbf{F}')_{ij}\frac{\mathbf{F}'_{ij}}{\mathbf{F}_{ij}^2} \\
&+ 2\mathbf{A}_{ij}^+\frac{\mathbf{F}'_{ij}}{\mathbf{F}_{ij}^2} + 2\frac{\mathbf{A}_{ij}^-}{\mathbf{F}'_{ij}} \\
&+ 2\frac{(\mathbf{F}'\mathbf{B}^+)_{ij}}{\mathbf{F}'_{ij}} + 2(\mathbf{F}'\mathbf{B}^-)_{ij}\frac{\mathbf{F}'_{ij}}{\mathbf{F}_{ij}^2})
\end{aligned}$$

is a diagonal matrix with positive diagonal elements.

Thus $Z(\mathbf{F},\mathbf{F}')$ is a convex function of $\mathbf{F}$. Therefore, we can obtain the global minimum of $Z(\mathbf{F},\mathbf{F}')$ by setting $\frac{\partial Z(\mathbf{F},\mathbf{F}')}{\partial \mathbf{F}_{ij}} = 0$ and solving for $\mathbf{F}$, from which we can get Eq.(24). □

B. PROOF OF THEOREM 2.6

Proof. We rewrite Eq.(25) as

$$\begin{aligned}
L(\mathbf{G}) &= \mathrm{tr}(\mu\mathbf{G}^T\mathbf{L}_G^+\mathbf{G} - \mu\mathbf{G}^T\mathbf{L}_G^-\mathbf{G} - 2\mathbf{G}^T\mathbf{P}^+ \\
&+ 2\mathbf{G}^T\mathbf{P}^- + \mathbf{G}\mathbf{Q}^+\mathbf{G}^T - \mathbf{G}\mathbf{Q}^-\mathbf{G}^T) \qquad (34)
\end{aligned}$$

By applying Lemma 2.3, we have

$$\mathbf{tr}(\mathbf{G}^T\mathbf{L}_G^+\mathbf{G}) \leq \sum_{ij}\frac{(\mathbf{L}_G^+\mathbf{G}')_{ij}\mathbf{G}_{ij}^2}{\mathbf{G}'_{ij}}$$

$$\mathbf{tr}(\mathbf{G}\mathbf{Q}^+\mathbf{G}^T) \leq \sum_{ij}\frac{(\mathbf{G}'\mathbf{Q}^+)_{ij}\mathbf{G}_{ij}^2}{\mathbf{G}'_{ij}}$$

Moreover, by the inequality $a \leq \frac{(a^2+b^2)}{2b}, \forall a, b > 0$, we have

$$\mathrm{tr}(\mathbf{G}^T\mathbf{P}^-) = \sum_{ij}\mathbf{P}_{ij}^-\mathbf{G}_{ij} \leq \sum_{ij}\mathbf{P}_{ij}^-\frac{\mathbf{G}_{ij}^2+\mathbf{G}_{ij}'^2}{2\mathbf{G}'_{ij}}$$

To obtain the lower bound for the remaining terms, we use the inequality that $z \geq 1+\log z, \forall z > 0$, then

$$\begin{aligned}
\mathrm{tr}(\mathbf{G}^T\mathbf{P}^+) &\geq \sum_{ij}\mathbf{P}_{ij}^+\mathbf{G}'_{ij}(1+\log\frac{\mathbf{G}_{ij}}{\mathbf{G}'_{ij}}) \\
\mathrm{tr}(\mathbf{G}^T\mathbf{L}_G^-\mathbf{G}) &\geq \sum_{ijk}(\mathbf{L}_G^-)_{jk}\mathbf{G}'_{ji}\mathbf{G}'_{ki}(1+\log\frac{\mathbf{G}_{ji}\mathbf{G}_{ki}}{\mathbf{G}'_{ji}\mathbf{G}'_{ki}}) \\
\mathrm{tr}(\mathbf{G}\mathbf{Q}^-\mathbf{G}^T) &\geq \sum_{ijk}\mathbf{Q}_{jk}^-\mathbf{G}'_{ij}\mathbf{G}'_{ik}(1+\log\frac{\mathbf{G}_{ij}\mathbf{G}_{ik}}{\mathbf{G}'_{ij}\mathbf{G}'_{ik}})
\end{aligned}$$

By summing over all the bounds, we can get $\mathbf{Z}(\mathbf{G},\mathbf{G}')$, which obviously satisfies (1) $\mathbf{Z}(\mathbf{G},\mathbf{G}') \geq J_{DRCC}(\mathbf{G})$; (2)$\mathbf{Z}(\mathbf{G},\mathbf{G}) = J_{DRCC}(\mathbf{G})$

To find the minimum of $\mathbf{Z}(\mathbf{G},\mathbf{G}')$, we take

$$\begin{aligned}
\frac{\partial Z(\mathbf{G},\mathbf{G}')}{\partial \mathbf{G}_{ij}} &= 2\mu\frac{(\mathbf{L}_G^+\mathbf{G}')_{ij}\mathbf{G}_{ij}}{\mathbf{G}'_{ij}} - 2\mu(\mathbf{L}_G^-\mathbf{G}')_{ij}\frac{\mathbf{G}'_{ij}}{\mathbf{G}_{ij}} \\
&- 2\mathbf{P}_{ij}^+\frac{\mathbf{G}'_{ij}}{\mathbf{G}_{ij}} + 2\mathbf{P}_{ij}^-\frac{\mathbf{G}_{ij}}{\mathbf{G}'_{ij}} \\
&+ 2\frac{(\mathbf{G}'\mathbf{Q}^+)_{ij}\mathbf{G}_{ij}}{\mathbf{G}'_{ij}} - 2(\mathbf{G}'\mathbf{Q}^-)_{ij}\frac{\mathbf{G}'_{ij}}{\mathbf{G}_{ij}}
\end{aligned}$$

and the Hessian matrix of $Z(\mathbf{G},\mathbf{G}')$

$$\begin{aligned}
\frac{\partial^2 Z(\mathbf{G},\mathbf{G}')}{\partial \mathbf{G}_{ij}\partial \mathbf{G}_{kl}} &= \delta_{ik}\delta_{jl}(2\mu\frac{(\mathbf{L}_G^+\mathbf{G}')_{ij}}{\mathbf{G}'_{ij}} + 2\mu(\mathbf{L}_G^-\mathbf{G}')_{ij}\frac{\mathbf{G}'_{ij}}{\mathbf{G}_{ij}^2} \\
&+ 2\mathbf{P}_{ij}^+\frac{\mathbf{G}'_{ij}}{\mathbf{G}_{ij}^2} + 2\frac{\mathbf{P}_{ij}^-}{\mathbf{G}'_{ij}} \\
&+ 2\frac{(\mathbf{G}'\mathbf{Q}^+)_{ij}}{\mathbf{G}'_{ij}} + 2(\mathbf{G}'\mathbf{Q}^-)_{ij}\frac{\mathbf{G}'_{ij}}{\mathbf{G}_{ij}^2})
\end{aligned}$$

is a diagonal matrix with positive diagonal elements.

Thus $Z(\mathbf{G},\mathbf{G}')$ is a convex function of $\mathbf{G}$. Therefore, we can obtain the global minimum of $Z(\mathbf{G},\mathbf{G}')$ by setting $\frac{\partial Z(\mathbf{G},\mathbf{G}')}{\partial \mathbf{G}_{ij}} = 0$ and solving for $\mathbf{G}$, from which we can get Eq.(26). □