Non-negative and sparse spectral clustering

Hongtao Lu a,b,*, Zhenyong Fu a,b, Xin Shu a,b

a Department of Computer Science and Engineering, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai 200240, China
b MOE-Microsoft Laboratory for Intelligent Computing and Intelligent Systems, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai 200240, China

A R T I C L E   I N F O

Article history:
Received 7 December 2012
Received in revised form 24 April 2013
Accepted 4 July 2013
Available online 16 July 2013

Keywords:
Spectral clustering
Non-negative matrix factorization
Ratio cut
Normalized cut
Sparseness

A B S T R A C T

Spectral clustering aims to partition a data set into several groups by using the Laplacian of the graph such that data points in the same group are similar while data points in different groups are dissimilar to each other. Spectral clustering is very simple to implement and has many advantages over the traditional clustering algorithms such as k-means. Non-negative matrix factorization (NMF) factorizes a non-negative data matrix into a product of two non-negative (lower rank) matrices so as to achieve dimension reduction and part-based data representation. In this work, we proved that the spectral clustering under some conditions is equivalent to NMF. Unlike the previous work, we formulate the spectral clustering as a factorization of data matrix (or scaled data matrix) rather than the symmetrical factorization of the symmetrical pairwise similarity matrix as the previous study did. Under the NMF framework, where regularization can be easily incorporated into the spectral clustering, we propose several non-negative and sparse spectral clustering algorithms. Empirical studies on real world data show much better clustering accuracy of the proposed algorithms than some state-of-the-art methods such as ratio cut and normalized cut spectral clustering and non-negative Laplacian embedding.

1. Introduction

In recent years spectral clustering has become one of the most popular modern clustering methods, e.g., [1,3–6]. Spectral clustering divides a data set into non-overlapped groups such that the data points in same group are similar as much as possible and the data points in different groups are dissimilar as much as possible. Spectral clustering is based on the spectral graph theory [2]. The main tools for spectral clustering are graph Laplacian matrices. There are different graph Laplacian matrices used for spectral clustering in the literature such as unnormalized graph Laplacian [3,4], normalized graph Laplacian [5,8]. The basic steps of the spectral clustering are the first to construct a similarity graph from data and then to compute the first k eigenvectors corresponding to the k smallest eigenvalues of the graph Laplacian matrix which is derived from the similarity graph, and finally use the k-means algorithm to cluster the k dimensional row vectors of the matrix formed by the first k eigenvectors of the Laplacian as columns [1,5,6].

Spectral clustering can be understood from the point of view of the graph cut. For the data given in the form of a similarity graph, one wants to partition the graph into subgraphs such that the edges between different subgraphs have low weights and the edges within a subgraph have high weights. Spectral clustering can be derived as an approximation of graph partition problem which can be solved by minimizing graph cuts. There are different graph cuts used for graph partition such as ratio cut (Rcut) [3], normalized cut (Ncut) [5] and min–max cut (MMcut) [7]. Spectral clustering also has a margin-based perspective [8]. Spectral clustering methods are widely applied in image segmentation [5,9], gene network analysis [10], speech separation [11] and many other fields whenever clustering methods are employed.

Non-negative matrix factorization (NMF) was first introduced into machine learning and pattern recognition communities by Lee and Seung [12,13]. NMF decomposes a non-negative data matrix into a product of two non-negative matrices. The distinguished feature of NMF is the non-negativity of all elements of the matrices involved. This non-negativity is often encountered in real world data such as intensity value of image pixels, document-term matrix, rating matrix, etc. NMF allowing only non-negative factorization makes the data representation to have part-based meaning [12]. Recent extensions of NMF include convex NMF [14], orthogonal NMF [15], convex-hull NMF [16], etc.

NMF has been proved to be closely related to some classical algorithms in machine learning. For example, probabilistic latent semantic indexing (PLSI) has been proved to be equivalent to NMF [17], kernel k-means has also been proved to be equivalent to NMF [18], and it is found that the spectral clustering with Ncut is also equivalent to NMF [19]. A non-negative Laplacian embedding
(equivalent to Rcut spectral clustering) has been proposed and its connection to NMF has been revealed in [20].

In this paper, we only consider spectral clustering methods using Rcut and Ncut as objective functions. We will prove the equivalence between spectral clustering under some conditions and non-negative matrix factorization with proper constraints. Our result is different from that in [19], where the authors proved that spectral clustering with Ncut objective function is equivalent to the symmetrical NMF which factorizes the symmetrical pairwise similarity matrix as a symmetrical product (i.e., product of a non-negative matrix with its transpose). Our result is also different from that of [20] where the authors consider the non-negative Laplacian embedding (which is equivalent to Rcut spectral clustering), which is also to factorize a symmetrical non-negative matrix into a symmetrical product. In the present work, we consider the data matrix itself. Under proper conditions, we prove that a relaxed spectral clustering algorithm with Rcut objective function is equivalent to non-negative factorization of the data matrix into a product of a non-negative matrix with orthogonal columns and another non-negative matrix. For the spectral clustering with Ncut objective function, it is proved to be equivalent to the similar factorization of the normalized data matrix. So the non-negative matrix factorization framework unifies the spectral clustering algorithms using Rcut and Ncut as objective functions.

This non-negative matrix factorization framework provides an insight into connections between spectral clustering and matrix factorization, under this framework, spectral clustering algorithms can benefit from NMF solving techniques. For example, additional constraints can conveniently be incorporated into the framework to derive new spectral clustering algorithms such as non-negative sparse spectral clustering.

2. Spectral clustering

We start with a brief introduction of spectral clustering. Suppose we are given n data points \( x_1, x_2, \ldots, x_n \), each of dimension \( p \), i.e., \( x_i \in \mathbb{R}^p \), let \( X = (x_1, x_2, \ldots, x_n) \) denotes the data matrix of \( p \times n \). From the data points, we can construct a similarity graph \( G = (V, E) \), where the node set is \( V = \{x_1, x_2, \ldots, x_n\} \), and sometimes we also say that the node set is \( V = \{1, 2, \ldots, n\} \) without inducing confusion. \( E \) is the edge set. In addition, suppose an \( n \times n \) matrix \( W \) of pairwise similarities (weights) among these \( n \) points is also available. The pairwise similarity weights can be computed from the data points. There are some typical methods to construct the similarity graph and compute the similarities between data points, for example, the \( \epsilon \)-neighborhood graphs, \( k \)-nearest neighbor graphs, and fully connected graphs with Gaussian similarity function \([1,2,5]\). The similarity matrix can also be independent of the node contents and specified in advance as side information \([22]\). In this paper, without loss of generality, we assume the pairwise similarity being nonnegative, i.e., \( w_{ij} \geq 0 \), and the similarity graph is undirected and symmetric, this means \( w_{ij} = w_{ji} \). Specifically, we consider the similarity matrix based on the inner product, i.e., \( W = X^T X \).

2.1. Graph Laplacian

For a similarity graph \( G = (V, E) \) with weight matrix \( W \), the degree \( d_i \) of node \( i \) is

\[
d_i = \sum_j w_{ij}
\]

The degree matrix \( D \) is defined as the diagonal matrix with the degrees \( d_1, d_2, \ldots, d_n \) on its diagonal, i.e., \( D = \text{diag}(d_1, d_2, \ldots, d_n) \). The unnormalized graph Laplacian matrix is defined as

\[
L = D - W.
\]

It can be proved that \( L \) is symmetric and semi-positive definite, and for any vector \( x \in \mathbb{R}^n \)

\[
x^T L x = \frac{1}{2} \sum_{i,j=1}^{n} (x_i - x_j)^2 w_{ij}.
\]

The symmetric normalized graph Laplacian matrix \( L_{\text{sym}} \) is defined as

\[
L_{\text{sym}} = D^{-\frac{1}{2}}L D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}
\]

where \( I \) is the identity matrix. Similarly

\[
x^T L_{\text{sym}} X = \frac{1}{2} \sum_{i,j=1}^{n} \left( \frac{x_i}{\sqrt{d_i}} - \frac{x_j}{\sqrt{d_j}} \right)^2 w_{ij}.
\]

2.2. Graph cut

The spectral clustering of data points can be interpreted by the partitioning of the similarity graph into several non-overlapping parts and the problem can be solved by the graph cut approaches. Graph cut algorithms divide a graph into groups such that edges between different groups have low weights and edges within each group have high weights. This can be formulated as a minimization problem of an appropriate graph cut objective function. The first attempt is to directly minimize the cut \( s(A, B) \), between two partitions \( A \) and \( B \), where the cut \( s(A, B) \) is defined as

\[
s(A, B) = \sum_{i \in A, j \in B} w_{ij},
\]

However, this mincut algorithm is likely to divide out small subgraphs, making the partition severely unbalanced. The ratio cut (Rcut) \( J_{\text{rc}} \) is one of such objective functions proposed to handle this problem \([21]\)

\[
J_{\text{rc}} = \frac{s(A, B)}{|A|} + \frac{s(A, B)}{|B|}
\]

where \( |A| \) means the cardinality of \( A \). The graph cut problem based on Rcut is solved by minimizing \( J_{\text{rc}} \) with respect to \( A \) and \( B \). Shi and Malik \([5]\) proposed another objective function: the normalized cut (Ncut) \( J_{\text{nc}} \):

\[
J_{\text{nc}} = \frac{s(A, B)}{d(A)} + \frac{s(A, B)}{d(B)}
\]

where \( d(A) = \sum_i d_i \) is the sum of node degrees of \( A \).

Chan et al. \([4]\) generalized the two-way ratio cut of \( J_{\text{rc}} \) to multi-way ratio cut where the nodes of graph \( G \) are divided into \( K \) disjoint groups \( C_k \) by minimizing the following multi-way ratio cut objective function:

\[
J_{\text{rc}} = \sum_{i=1}^{K} \left\{ \frac{s(C_i, \overline{C_i})}{|C_i|} \right\} + \sum_{i<j} \left\{ \frac{s(C_i, C_j)}{|C_i| |C_j|} \right\}
\]

where \( h_i \) be the indicator vector for cluster \( C_i \), i.e., \( h_i(i) = 1 \) if \( x_i \in C_i \), otherwise \( h_i(i) = 0 \), then \( |C_i| = h_i^T h_i \). It is easy to verify that

\[
J_{\text{rc}} = \sum_{i=1}^{K} \frac{s(C_i, \overline{C_i})}{|C_i|} - \frac{1}{2} \sum_{i<j} \frac{s(C_i, C_j)}{|C_i| |C_j|}
\]

where \( s(C_i, \overline{C_i}) \) is the cut between \( C_i \) and its complement \( \overline{C_i} = V - C_i \), \( s(C_i, C_j) \) can be expressed as

\[
s(C_i, C_j) = h_i^T (D-W) h_j
\]

If we define the \( n \times K \) cluster indicator matrix \( H \) as

\[
H = \left( \begin{array}{c}
\|h_1\| \\
\|h_2\| \\
\vdots \\
\|h_K\|
\end{array} \right)
\]
then the ratio cut can be reformulated as
\[
J_{rc} = \sum_{i=1}^{K} \frac{h_i^T(D-W)h_i}{h_i^T h_i} = \text{tr}(H^T(D-W)H)
\]
\[
= \text{tr}(H^T LH)
\]
where \(\text{tr}(\cdot)\) is the trace of a matrix. So the ratio cut spectral clustering can be solved by the following minimization problem:
\[
\min_{H,H^T} \text{tr}(H^T LH) \quad (14)
\]
The normalized cut can also be generalized to multi-way clustering as
\[
J_{nc} = \sum_{1 \leq i < j \leq K} \left\{ \frac{s(C_p,C_q)}{d(C_p)} \times \frac{s(C_p,C_q)}{d(C_q)} \right\}
\]
\[
= \sum_{i=1}^{K} \frac{s(C_i,C_i)}{d(C_i)}
\]
\[
= \sum_{i=1}^{K} \frac{s(C_i,C_i)}{h_i^T D h_i}
\]
If we define cluster indicator vector as \(z_i = D^{1/2}h_i/\|D^{1/2}h_i\|\) and the cluster indicator matrix as \(Z = (z_1, z_2, ..., z_k)\), then \(Z^T Z = I\) and
\[
J_{nc} = \sum_{i=1}^{K} \frac{h_i^T(D-W)h_i}{h_i^T D h_i}
\]
\[
= \sum_{i=1}^{K} z_i^T (D^{-1/2}WD^{-1/2}) z_i
\]
\[
= \text{tr}(Z^T L_{sym} Z)
\]
So the Ncut spectral clustering becomes the following minimization problem:
\[
\min_{Z \geq 0} \text{tr}(Z^T L_{sym} Z) \quad (17)
\]

3. Spectral clustering as NMF

In this section, we will establish a connection between spectral clustering and NMF. This connection has been revealed in the previous work to some extent. Ding et al. [19] proved that Ncut spectral clustering is equivalent to the following symmetric NMF:
\[
D^{-1/2}WD^{-1/2} \approx HH^T \quad \text{s.t.} \quad H \geq 0
\]
where \(H \geq 0\) means \(H\) is a nonnegative matrix. Luo et al. [20] have drawn similar conclusion for the nonnegative Laplacian embedding (which is equivalent to Rcut spectral clustering), which says that the nonnegative Laplacian embedding is equivalent to the following symmetric NMF:
\[
W = D + \sigma I = QQ^T \quad \text{s.t.} \quad Q^T Q = I, \quad Q \geq 0
\]
where \(\sigma\) is the largest eigenvalue of the Laplacian matrix \(L\).

Both results are in terms of factorization of symmetric similarity matrix \(W\), since \(W\) is usually computed from the data points, our question is whether spectral clustering can be casted into the factorization of data matrix itself. Under certain assumptions, we give affirmative answer. Our further assumptions include that the data matrix \(X\) is nonnegative, the similarity graph is fully connected and the similarity matrix \(W\) is given by inner product between data points, i.e., \(W = X^T X\). This similarity has been used in the experiment of [20].

We first consider the Ncut spectral clustering. Obviously, the Ncut spectral clustering (17) is equivalent to
\[
\max_{Z \geq 0} \text{tr}(Z^T D^{-1/2}WD^{-1/2}Z) \quad (20)
\]

**Theorem 1.** If the data matrix \(X \geq 0\) and the similarity between data points is defined by inner product, i.e., the similarity matrix \(W = X^T X\), then the \(K\)-way Ncut spectral clustering (20) is equivalent to the non-negative matrix factorization of the scaled data matrix \(D^{-1/2}X^T \approx ZY\) subject to \(Z^TZ = I\), where \(Z \in \mathbb{R}^{k \times K}\) and \(Y \in \mathbb{R}^{k \times p}\) are also two non-negative matrices, and the rows of \(Z\) serve as a clustering indicator vector of each data point, its columns are clustering indicator vector for each cluster.

Theorem 1 shows that Ncut spectral clustering can be casted into a NMF problem. The proof of theorem is given in the appendix.

For ratio cut spectral clustering (14), we cannot obtain exact equivalence between Rcut spectral clustering and NMF. However, we can relax the Rcut spectral clustering and get an equivalence between the relaxed Rcut spectral clustering and NMF.

**Theorem 2.** If the data matrix \(X \geq 0\) and the similarity is defined by the inner product between data points, i.e., the similarity matrix \(W = X^T X\), then we can relax the \(K\)-way Rcut spectral clustering (14) to a relaxed Rcut spectral clustering such that it is equivalent to the non-negative matrix factorization of the data matrix \(X^T \approx H Y\) such that \(H^T H = I\), where \(H \in \mathbb{R}^{k \times K}\) and \(Y \in \mathbb{R}^{k \times p}\) are also two non-negative matrices, and the rows of \(H\) serve as a clustering indicator vector of each data point, its columns are clustering indicator vector for each cluster.

The proof of Theorem 2 is also given in the appendix. Theorems 1 and 2 reveal the intrinsic connections between spectral clustering and NMF. Generally speaking, we proved that spectral clustering is equivalent to NMF of data matrix or scaled data matrix with orthonormal constraints.

In Theorem 2, the NMF \(X^T \approx HY\) can be rewritten as
\[
X \approx Y^T H \quad (21)
\]
In the interpretation of NMF, we know that (21) transforms the data matrix \(X\) into the low dimensional representation. The columns of \(Y\) (i.e., the rows of \(Y\)) consist of a basis for the low dimensional representation, and the columns of \(H^T\) (rows of \(H\)) are the coefficient vectors of original data points in the low-dimensional space. Each original \(n\)-dimensional data point (a column of \(X\)) is represented by a \(K\)-dimensional vector (a row vector of \(H\)), this is why in spectral clustering, the rows of \(H\) are used as initial cluster indicator vectors for subsequent \(K\)-means clustering [1]. Theorem 1 has similar explanation for Ncut spectral clustering, but the data matrix is replaced by the scaled one.

In traditional spectral clustering methods, the eigenvectors of the Laplacian matrices may have negative components. In our NMF based spectral clustering, the cluster indicator matrix \(H\) (or \(Z\) for Ncut) is non-negative. We call our spectral clustering algorithms as non-negative spectral clustering (NSC). We will also propose the non-negative sparse spectral clustering (NSSC) in Section 5. In the next section, we will give several update rules for solving NSC and NSSC.

4. Update algorithms for non-negative spectral clustering

By casting the spectral clustering into the NMF framework, several advantages can be achieved as follows:

1. It provides insight into connections between spectral clustering and NMF, and also between Ncut and Rcut spectral clustering. From NMF point of view, they differ only in scales.
2. One can use the update algorithms of NMF to solve spectral clustering, avoiding the computation of eigenvalues and eigenvectors in traditional spectral clustering, which is time-consuming for large scale problems.
3. NMF framework can easily incorporate additional constraints into spectral clustering, such as sparse constraints as will be illustrated later.

4. NMF framework simultaneously calculates the basis $Y$ and the cluster indicator matrix $H$ (or $Z$), so leads to a solution to out-of-sample problem. When a new sample $x$ comes, one can just project the sample $x$ onto the low dimensional space by $h = Yx$, $h$ is then the cluster indicator of $x$. Traditional spectral clustering methods cannot deal with this incremental learning situation, because they compute the eigenvectors of the graph Laplacian matrix, when new samples come, the Laplacian matrix may change. Even the symmetrical NMF methods of spectral clustering in [19,20] have no access to the basis matrix $Y$, making them inappropriate to handle the out-of-sample problem.

In this section, we employ NMF update algorithms with orthonormal constraints to solve Ncut and Rcut spectral clustering problems. The first update algorithm is the standard multiplicative update rule as in [12,13]. Taking account of the orthonormal constraints, we have the following:

- **Multiplicative update rule for Ncut NSC**:
  1. $Z_{ij}^{n+1} \propto \frac{YXD^{1/2}h_j}{(YY^TZ)^{1/2}h_j}$; \hspace{2cm} (22)
  2. $Y_{ik}^{n+1} \propto (XD^{-1/2}Z)_{hk}$.

- **Multiplicative update rule for relaxed Rcut NSC**:
  1. $H_{ij}^{n+1} \propto \frac{(YY)h_j}{(YY^HZ)^{1/2}h_j}$; \hspace{2cm} (24)
  2. $Y_{ik}^{n+1} \propto (XH)_{hk}$.

The convergence of these update rules has been proved in [13] by the auxiliary function method, which means that these update rules guarantee the decrease of error and eventually converge to a local minima.

The convergence rate of multiplicative update algorithms may be slow. Since the objective function of NMF is linear when fixing either of the matrices on the right side, an alternating least square approach can be used to optimize the solution in each iteration. In addition, to keep the non-negativity of elements of matrices, we use ideas analogous to the projected gradient methods in iterations of NMF [23]. The projected gradient methods are typical approaches to solve bound-constrained optimization problems, where variables are constrained by certain bounds. NMF is a kind of bound-constrained optimization problem. The basic idea of the projected gradient is to update variables as in normal gradient descent method, but when the variables are out of the bounds, they are pulled back into the bounds by projection. We name this algorithm the projected alternating least square, which is stated as follows:

1. **Projected alternating least square update rule for Ncut NSC**:  
   1. Orthonormalize columns of $Z$; \hspace{2cm} (26)
   2. $Y^T = XD^{-1/2}Z$.
   3. $Z^T = \max(0,(Y^T)^+XD^{-1/2})$ \hspace{2cm} (28)

1. **Projected alternating least square update rule for Rcut NSC**:
   1. Orthonormalize columns of $H$; \hspace{2cm} (29)
   2. $Y^T = XH$.
   3. $H^T = \max(0,(Y^T)^+X)$ \hspace{2cm} (31)

where $(Y^T)^+$ stands for the pseudoinverse of matrix $Y^T$.

Note that in these update rules, the update formula for $Y$ is the same, the difference only lies in update method of $H$ (or $Z$). The NSC algorithm framework is depicted in Table 1.

### 5. Non-negative sparse spectral clustering

Sparseness has recently attracted much attention in signal representation and machine learning fields. There is an evidence that human perception is closely related to sparseness. Sparse signal representation implies better robustness and lower computation complexity. The simplest way to characterize sparseness of a vector $x$ is to minimize its $l_0$ norm (in fact, pseudo-norm) $\|x\|_0$ that is the number of nonzero components of $x$. However, minimization of $\|x\|_0$ results in a NP-hard combinatorial optimization

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Non-negative spectral clustering (NSC) algorithms.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong> data matrix $X$, cluster number $K$</td>
<td></td>
</tr>
<tr>
<td>1. Randomly initialize two non-negative matrices $H$ (or $Z$) and $Y$</td>
<td></td>
</tr>
<tr>
<td>2. Update $H$ (or $Z$) and $Y$ using appropriate update rules from (22) to (31) until convergence</td>
<td></td>
</tr>
<tr>
<td>3. Output cluster indicator matrix $H$ (or $Z$), and the basis matrix $Y$</td>
<td></td>
</tr>
<tr>
<td>4. The $i$th row of $H$ serves as the cluster indicator vector of $x_i$, a $K$-means clustering procedure is applied to rows of $H$ to obtain the cluster membership of data points</td>
<td></td>
</tr>
<tr>
<td>5. If a new sample $x$ comes, its cluster indicator vector is computed by $Yx$, and its cluster membership is obtained by the $K$-means clustering</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Non-negative sparse spectral clustering (NSSC) algorithms.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong> data matrix $X$, cluster number $K$, control parameter $\lambda$</td>
<td></td>
</tr>
<tr>
<td>1. Randomly initialize two non-negative matrices $H$ (or $Z$) and $Y$</td>
<td></td>
</tr>
<tr>
<td>2. Update $Y$ using the same rules as in Table 1 and update $H$ using (38) (Rcut using (39)) until convergence</td>
<td></td>
</tr>
<tr>
<td>3. Output cluster indicator matrix $H$ (or $Z$), and the basis matrix $Y$</td>
<td></td>
</tr>
<tr>
<td>4. The $i$th row of $H$ serves as the cluster indicator vector of $x_i$, a $K$-means clustering procedure is applied to rows of $H$ to obtain the cluster membership of data points</td>
<td></td>
</tr>
<tr>
<td>5. If a new sample $x$ comes, its cluster indicator vector is computed by $Yx$, and its cluster membership is obtained by the $K$-means clustering</td>
<td></td>
</tr>
</tbody>
</table>
problem, so $l_0$ norm is usually replaced by the $l_1$ norm $\|x\|_1$. It has been proved theoretically that for an under-determined system of linear equations, if its solution is sparse enough, then the solution of minimizing $l_1$ norm is exactly the solution of minimizing $l_0$ norm, i.e., the sparsest solution [24,25]. So, most current sparse solutions are realized by adding $l_1$-norm constraints to the original problem. In the literature, different sparse variants of classical methods have been proposed, for example, sparse least square regression (Lasso) [26], sparse principal component analysis [27,28], sparse kernel principal component analysis [29], and sparse non-negative matrix factorization [30], etc.

In this section, we consider the sparse spectral clustering, to our knowledge, this problem has not been addressed thoroughly. Sparseness is more important to spectral clustering because the cluster assignment is based on the largest component of the eigenvector of the Laplacian matrix, if the eigenvector is not sparse, the method would be prone to noise. Sparseness of indicator vector will obviously increase robustness of spectral clustering. In traditional spectral clustering, sparseness constraints are not easy to incorporate into the eigenvector framework of the Laplacian. However, in our NMF framework, we can easily add sparse constraints.

We take the Ncut as example, considering the NMF equivalence of Ncut spectral clustering

$$\min_{Z,Y,Z} \frac{1}{2} \| D^{-1/2} X^T - ZY \|_F^2. \tag{32}$$

We use the sum of the $l_1$ norms of rows (or columns) of $Z$ to measure sparseness, combining these sparseness criteria into the objective function and noting the non-negativity of $Z$, we obtain

$$\min_{Z,Y,Z} \left\{ \frac{1}{2} \| D^{-1/2} X^T - ZY \|_F^2 + \lambda \sum_{i=1}^{m} \sum_{k=1}^{K} z_{ik} \right\}, \tag{33}$$

where $\lambda$ controls the trade-off between sparseness and the accuracy of reconstruction. We call this spectral clustering as non-negative sparse spectral clustering (NSSC). To solve this optimization problem, note that the update rule for $Y$ is the same as the above section without sparseness constraints. In order to derive updating rule for $Z$, we note that the objective function of (33) can be rewritten as

$$L(Z,Y) = \frac{1}{2} \| D^{-1/2} X^T - ZY \|_F^2 + \lambda \text{tr} \left( 1_n^T Z_1 \right)$$

$$+ \lambda \text{tr} \left( 1_n^T \frac{Z_1}{K} \right) \tag{34}$$

where $1_n$ is the $n$-dimensional vector of all 1s. The partial derivative is

$$\frac{\partial L}{\partial Z} = -XYD^{-1/2} + YY^T Z^T + \lambda 1_k 1_n^T \tag{35}$$

So the gradient descent based update rule is

$$Z_{ij} \leftarrow Z_{ij} + \eta_{ij} \left( XYD^{-1/2} - YY^T Z^T + \lambda 1_k 1_n^T \right) \tag{36}$$

As in [13], if we set the stepsize $\eta_{ij}$ as

$$\eta_{ij} = \frac{Z_{ij}}{(YY^T Z^T + \lambda 1_k 1_n^T)} \tag{37}$$

![Fig. 1. Thirty images of three persons used for illustrating clustering properties.](image1)

![Fig. 2. Clustering effects of (a) NLE and (b) NSC-Rcut. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)](image2)
then the update rule (36) becomes a multiplicative one as

\[ Z_{kj}^{t} \leftarrow \frac{(YX^{1/2})_{kj}}{(YY^{T})_{kj}^{1/2} + \lambda} \]  

(38)

Note that in this update rule, the positive part of the gradient serves as the denominator of the update multiplier, and the negative part of the gradient serves as the numerator of the update multiplier, so the convergence can be proved by the auxiliary function approach as in [13].

For Rcut NSSC, similar update rule can be derived as

\[ H_{kj}^{t} \leftarrow \frac{(YX)_{kj}}{(YY^{T})_{kj} + \lambda} \]  

(39)

The NSSC algorithm is described in Table 2.

6. Experimental results

In this section, we carry out extensive experiments on real world data to illustrate the effectiveness and efficacy of our algorithms, we also make comparisons between our algorithms and traditional Ncut and Rcut spectral clustering algorithms and the non-negative Laplacian embedding (NLE) proposed in [20].

Our experimental settings are similar to [20] and we use the same data sets as in [20]: AT&T face database [31] and UCI datasets [32]. Throughout the experiments, we use the inner product to compute the similarity, i.e., for any two data points \( x_i \) and \( x_j \), the similarity \( w_{ij} = x_i^T x_j \). And in all experiments, we only use multiplicative update rules, we name our algorithms as NSC-Rcut, NSC-Ncut, NSSC-Rcut, NSSC-Ncut, and we name the traditional ratio cut and normalized cut spectral clustering as Rcut and Ncut, respectively.

Table 3
Features of UCI and AT&T datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Sample</th>
<th>#Dimension</th>
<th>#Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dermatology</td>
<td>366</td>
<td>33</td>
<td>6</td>
</tr>
<tr>
<td>Glass</td>
<td>214</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Soybean</td>
<td>47</td>
<td>35</td>
<td>6</td>
</tr>
<tr>
<td>Vehicle</td>
<td>846</td>
<td>18</td>
<td>4</td>
</tr>
<tr>
<td>Zoo</td>
<td>101</td>
<td>16</td>
<td>7</td>
</tr>
<tr>
<td>AT&amp;T</td>
<td>400</td>
<td>10,304</td>
<td>40</td>
</tr>
</tbody>
</table>

Fig. 3. Comparisons of clustering accuracy between NSC-Rcut, NSC-Ncut and NLE for UCI and AT&T datasets. (a) Dermatology, (b) Glass, (c) Soybean, (d) Vehicle, (e) Zoo and (f) AT&T.
6.1. An illustrative example

In this subsection, we use images of three persons from AT&T face database to illustrate the clustering effects of our algorithms, taking NSC-Rcut as an example, we also make comparison between NSC-Rcut and NLE. Fig. 1 shows the images we used; each person has 10 images with different expressions and poses. Each image is of size 112 × 92. We cascade the rows of each image to form a 10,304-dimensional vector and the data matrix \( X \) is thus of 10,304 × 30. We randomly initialize a 30 × 3 matrix \( H \) and a 3 × 10,304 matrix \( Y \) for NSC-Rcut, and then use the update rules (24) and (25) to update \( H \) and \( Y \), respectively. Fig. 2(b) shows the clustering effect of NSC-Rcut after 300 iterations where each row of matrix \( H \) is plotted as a 3D point and different shapes and colors represent different clusters. The clustering property of same cluster and separation between different clusters are evident. For comparison, Fig. 2(a) shows the \( Q \) matrix of NLE (Eq. (9) of [20]) for same images, it is obvious that two clusters (blue and green) cannot be separated.

6.2. Comparisons between NSC-Rcut, NSC-Ncut and NLE

In this subsection, we investigate the clustering accuracy of our algorithm NSC-Rcut and NSC-Ncut on UCI and AT&T datasets, and also make comparisons with NLE. The clustering accuracy is evaluated by the percentage of data points that are correctly clustered and can be computed via the Hungarian algorithm as in [20]. The UCI and AT&T datasets used are the same as in [20] and their features such as samples numbers, dimensions of data points and class numbers are listed in Table 3. Fig. 3 shows the clustering accuracy of the three algorithms for six datasets, where the vertical axis is the clustering accuracy and the horizontal axis is the logarithm of the number of runs of the algorithms. For each algorithm, we perform independently \( N = 256 \) runs with random initializations, and then we choose the best clustering accuracy among the first \( 2^n \), \( n = 1, 2, \ldots, 8 \) runs be plotted in Fig. 3. In each run, we first randomly initialize matrices of appropriate dimensions, and then iterate the update rules for 300 times to achieve convergence and obtain the cluster indicator matrix \( H \) (for Rcut, \( Z \) for Ncut, \( Q \) for NLE), finally \( K \)-means clustering is applied to the rows of \( H \) (or \( Z, Q \)) to obtain the clustering results. We can see that for five datasets except the ‘Glass’ dataset, our NSC-Ncut and NSC-Rcut achieve much better clustering accuracy than NLE, also note that with only a few runs (even one random run), our algorithms achieve good results while NLE performs poorly. For AT&T database, there are 40 classes; our algorithms achieve clustering accuracy of about 0.4 from the very beginning, a reasonably good result for 40 classes, while the clustering accuracy of NLE is always below 0.28. Note also that in

![Fig. 4. Comparisons of clustering accuracy between NSC-Rcut, NSC-Ncut, NSSC-Rcut, NSSC-Ncut, Rcut and Ncut for UCI and AT&T datasets. (a) Dermatology, (b) Glass, (c) Soybean, (d) Vehicle, (e) Zoo and (f) AT&T.](image-url)
almost all cases, NSC-Ncut performs always better than NSC-Rcut. For the ‘Glass’ dataset, all the three algorithms perform equally not good.

6.3. Comparisons between NSC, NSSC and the standard spectral clustering

In this subsection, we compare clustering accuracy between our algorithms NSC-Ncut, NSC-Rcut, NSSC-Ncut, NSSC-Rcut and the standard ratio cut (Rcut) and normalized cut (Ncut) spectral clustering algorithms. The experimental settings are the same as in the above section. We also independently perform each algorithm of NSC-Ncut, NSC-Rcut, NSSC-Ncut, NSSC-Rcut for 256 runs with random initializations, each run is iterated for 300 iterations. But for Rcut and Ncut, the first K eigenvectors are computed once and then 256 runs of K-means are performed. For NSSC, the parameter α = 0.0005. Fig. 4 shows the clustering accuracy versus the logarithm log2 N of run number N on six datasets. We can see that for Dermatology, Soybean, Vehicle, and Zoo datasets, our four algorithms achieve better or much better clustering accuracy than Ncut perform the best in almost all cases; this illustrates that our algorithms achieve better clustering accuracy than state-of-the-art algorithms in most cases.

### Acknowledgment

This work is supported by NSFC (Nos. 61272247 and 60873133), 973 (2009CB320901), 863 (No. 2008A022310) in China and the European Union Seventh Framework Programme (Grant no. 247619).

### Appendix A

**Proof of Theorem 1.** The factorization $D^{-1/2}X^TY = ZY$ can be solved by the following optimization problem

$$\min_{Z,Y,Z} \frac{1}{2} \|D^{-1/2}X^TY - ZY\|^2_F,$$  \tag{40}

where $\| \cdot \|_F$ is the matrix Frobenius norm. The objective function of (40) can be reformulated as

$$J(Z,Y) = \frac{1}{2} \text{tr} \left( (D^{-1/2}X^TY)^T \left( D^{-1/2}X^TY \right) \right)$$

$$= \frac{1}{2} \text{tr} \left( XD^{-1/2}X^TY^T - 2Y^T Z^T D^{-1/2}X^T + Y^T Y \right)\tag{41}$$

The constraint $Z^T Z = I$ is employed in the last equality. Calculating the partial derivative of $J(Z,Y)$ with respect to $Y$ and letting it be equal to 0, we have

$$\frac{\partial J(Z,Y)}{\partial Y} = -Z^T D^{-1/2}X^T + Y = 0$$ \tag{42}

From (42), we have

$$Y = Z^T D^{-1/2}X^T$$ \tag{43}

substituting (43) back into (41), we obtain

$$J(Z) = \frac{1}{2} \text{tr} \left( XD^{-1/2}X^T - XD^{-1/2}Z I Z^T D^{-1/2}X^T \right)$$ \tag{44}

since $XD^{-1/2}X^T$ is constant, not dependent on $Z$ and $Y$, $\min_{Z,Y,Z} J(Z)$ is equivalent to

$$\max_{Z^T Z = I} \text{tr} (XD^{-1/2}Z I Z^T D^{-1/2}X^T)$$ \tag{45}

this is further identical to

$$\max_{Z^T Z = I} \text{tr} (Z^T D^{-1/2}X^T XD^{-1/2}Z)$$ \tag{46}

Since $W = X^T X$, (46) is the same as (20). The equivalence between $K$-way Ncut spectral clustering and NMF is thus proved. So the ith row of $Z$ is the clustering indicator vector of ith data point $x_i$, and the ith column of $Z$ is the clustering indicator vector for the ith cluster.

### Table 4

<table>
<thead>
<tr>
<th>Methods</th>
<th>Dermatology</th>
<th>Glass</th>
<th>Soybean</th>
<th>Zoo</th>
<th>Vehicle</th>
<th>AT&amp;T</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ncut</td>
<td>Ave 0.7547</td>
<td>0.4653</td>
<td>0.7046</td>
<td>0.6317</td>
<td>0.3776</td>
<td>0.6224</td>
</tr>
<tr>
<td></td>
<td>Best 0.888</td>
<td>0.5654</td>
<td>0.7447</td>
<td>0.8614</td>
<td>0.3806</td>
<td>0.715</td>
</tr>
<tr>
<td>Rcut</td>
<td>Ave 0.4725</td>
<td>0.4155</td>
<td>0.6397</td>
<td>0.6096</td>
<td>0.3391</td>
<td>0.3316</td>
</tr>
<tr>
<td></td>
<td>Best 0.5574</td>
<td>0.4346</td>
<td>0.7234</td>
<td>0.7228</td>
<td>0.3735</td>
<td>0.3575</td>
</tr>
<tr>
<td>NLE</td>
<td>Ave 0.3489</td>
<td>0.2505</td>
<td>0.4771</td>
<td>0.4932</td>
<td>0.2824</td>
<td>0.2072</td>
</tr>
<tr>
<td></td>
<td>Best 0.6011</td>
<td>0.3738</td>
<td>0.8085</td>
<td>0.7624</td>
<td>0.4113</td>
<td>0.2625</td>
</tr>
<tr>
<td>NSC-Ncut</td>
<td>Ave 0.7129</td>
<td>0.2522</td>
<td>0.7143</td>
<td>0.6159</td>
<td>0.3957</td>
<td>0.3518</td>
</tr>
<tr>
<td></td>
<td>Best 0.9809</td>
<td>0.3598</td>
<td>1</td>
<td>0.8911</td>
<td>0.4504</td>
<td>0.415</td>
</tr>
<tr>
<td>NSC-Rcut</td>
<td>Ave 0.6602</td>
<td>0.2516</td>
<td>0.6986</td>
<td>0.6197</td>
<td>0.3863</td>
<td>0.3533</td>
</tr>
<tr>
<td></td>
<td>Best 0.9672</td>
<td>0.3692</td>
<td>1</td>
<td>0.8812</td>
<td>0.4314</td>
<td>0.415</td>
</tr>
<tr>
<td>NSSC-Ncut</td>
<td>Ave 0.7185</td>
<td>0.3477</td>
<td>0.7167</td>
<td>0.661</td>
<td>0.4196</td>
<td>0.025</td>
</tr>
<tr>
<td></td>
<td>Best 0.9809</td>
<td>0.4486</td>
<td>1</td>
<td>0.9109</td>
<td>0.4704</td>
<td>0.2325</td>
</tr>
<tr>
<td>NSSC-Rcut</td>
<td>Ave 0.67</td>
<td>0.2616</td>
<td>0.6976</td>
<td>0.6185</td>
<td>0.3865</td>
<td>0.3513</td>
</tr>
<tr>
<td></td>
<td>Best 0.9699</td>
<td>0.3832</td>
<td>1</td>
<td>0.8812</td>
<td>0.4551</td>
<td>0.435</td>
</tr>
</tbody>
</table>

In Table 4, the average and the best clustering accuracy of all seven algorithms on all six datasets are given for a detailed comparison. Each algorithm is independently run 256 times, and the average and the best results are calculated.

7. Conclusion

In this paper, we have proposed non-negative and sparse spectral clustering algorithms. We first investigate the connections between spectral clustering and nonnegative matrix factorization, we proved that if the data points are non-negative and the similarity between the data points is defined by the inner product, then the normalized cut spectral clustering is equivalent to non-negative matrix factorization of the scaled data matrix. For the ratio cut spectral clustering, we relax it by dropping the orthonormal constraints and get a relaxed ratio cut spectral clustering, we then proved that it is equivalent to non-negative matrix factorization of the data matrix. Under the NMF framework, we derived multiplicative and projected alternating least square update rules for the normalized cut and ratio cut spectral clustering. We also consider the non-negative sparse spectral clustering by imposing a sparse constraint on the cluster indicator matrix, and derive a multiplicative update rule. Extensive experiments on real world data show that our algorithms achieve better clustering accuracy than state-of-the-art algorithms in most cases.
Proof of Theorem 2. For the Rcut spectral clustering (14), define \( Q = D^{1/2}H \), then
\[
H^T DH = Q^T Q, \quad H^T H = Q^T D^{-1}Q
\]
and the Rcut (14) can be reformulated as
\[
\min_{Q, D^{-1/2}Q} \text{tr}(Q^T (I - D^{-1/2}WH D^{-1/2})Q)
\]
This is equivalent to
\[
\max_{Q, D^{-1/2}Q} \text{tr}(Q^T D^{-1/2}WH D^{-1/2}Q) \quad \text{s.t.} \quad Q^T Q = I
\]
Notice that \( Q = D^{1/2}H \), and columns of \( H \) are cluster indicators (see Eq. (12)), the constraint \( Q^T D^{-1}Q = I \) means the columns of \( Q \) are orthogonal to each other, so here it is reasonable to drop the orthonormal constraint \( Q^T Q = I \) and get a relaxed Rcut spectral clustering:
\[
\max_{Q, D^{-1/2}Q} \text{tr}(Q^T D^{-1/2}WH D^{-1/2}Q)
\]
Substituting \( H = D^{-1/2}Q \) back into (50), we have
\[
\frac{H^T H}{2} = I
\]
In a similar way to Theorem 1, it can be proved that if \( W = X^T X \), the NMF \( X^T = HY \) subject to \( H^T H = I \) is equivalent to the relaxed Rcut spectral clustering (51).

References


Hongtao Lu got his PhD degree in Electronic Engineering from Southeast University, Nanjing, in 1997. After graduation he became a postdoctoral fellow in the Department of Computer Science, Fudan University, Shanghai, China, where he spent 2 years. In 1999, he joined the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, where he is now a professor. His research interest includes machine learning, computer vision and pattern recognition. He has published more than sixty papers in international journals such as IEEE Transactions, Neural Networks and in international conferences. His papers got more than 800 citations by other researchers.