Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence

Spectral Rotation versus K-Means in Spectral Clustering

Jin Huang and **Feiping Nie** and **Heng Huang***
Computer Science and Engineering Department
University of Texas at Arlington
Arlington,TX,76019
huangjinsuzhou@gmail.com, feipingnie@gmail.com, heng@uta.edu

Abstract

Spectral clustering has been a popular data clustering algorithm. This category of approaches often resort to other clustering methods, such as K-Means, to get the final cluster. The potential flaw of such common practice is that the obtained relaxed continuous spectral solution could severely deviate from the true discrete solution. In this paper, we propose to impose an additional orthonormal constraint to better approximate the optimal continuous solution to the graph cut objective functions. Such a method, called spectral rotation in literature, optimizes the spectral clustering objective functions better than K-Means, and improves the clustering accuracy. We would provide efficient algorithm to solve the new problem rigorously, which is not significantly more costly than K-Means. We also establish the connection between our method and K-Means to provide theoretical motivation of our method. Experimental results show that our algorithm *consistently* reaches better cut and meanwhile outperforms in clustering metrics than classic spectral clustering methods.

Introduction

Clustering is widely used for exploratory data analysis, with applications ranging from artificial intelligence, statistics to social sciences. Among various clustering methods in the literature, spectral clustering is a popular choice. It is easy to efficiently implement and often outperforms traditional clustering methods such as K-Means. There are numerous papers proposed different spectral clustering algorithms, such as Ratio Cut (Hagen and Kahng 1992), K-way Ratio Cut(Chan, Schlag, and Zien 1994), Normalized Cut (Shi and Malik 2000; Ng, Jordan, and Weiss 2002) and Min-Max Cut (Ding et al. 2001; Nie et al. 2010), also some papers establishing the connection between spectral clustering and other clustering methods (Zha et al. 2001; Dhillon, Guan, and Kulis 2004; Ding, He, and Simon 2005; Nie et al. 2011; Luo et al. 2010). There are also quite a few tutorials that give an introduction to spectral clustering, such as (Luxberg 2007).

The goal of clustering is to separate data vectors in different clusters according to their similarities. For the similarity graph constructed from data, we want to find a partition of the graph such that the edges between different clusters have low weights and the edges within the same cluster have high weights. In other words, we want to find a partition such that data vectors within same cluster are similar to each other and vectors in different clusters are dissimilar from each other. It is NP-hard to solve the multi-cluster mincut problems with various balancing condition. Spectral clustering is a way to solve relaxed versions of such problems. Classical spectral methods such as Ratio Cut (Hagen and Kahng 1992) and Normalized Cut (Shi and Malik 2000; Ng, Jordan, and Weiss 2002) then generally use K-Means to do the clustering on the relaxed continuous spectral vectors, to obtain the final clusters. The subtle disadvantage of this approach is that the obtained continuous solution from graph cut could deviate far from the discrete solution, which would affect the cluster accuracy thereafter.

In this paper, we apply the spectral rotation technique to get the continuous spectral vector which is closer to the discrete cluster indicator than existing results. As we will show, this usually leads to a better cut in terms of objective function value and improvement in clustering accuracy. We introduce the details of the optimization method and reveal its connection and difference with K-Means.

In the rest of this paper, we first introduce background material on spectral clustering that serves as our motivation. Next we provide our objective function and give an optimization algorithm for it. After that, we describe several experiments we have run, comparing our algorithm to alternatives from the literature on 12 benchmark data sets. Finally, we conclude with additional observations and future work.

Spectral Clustering Background

Given n data vectors $\{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$, we can construct a graph using the data with weight matrix $\mathbf{W} \in \mathbb{R}^{n \times n}$. The spectral clustering objective function is

$$\mathrm{J} = \sum_{1 \le \mathrm{p} < \mathrm{q} \le \mathrm{K}} \frac{\mathrm{s}\left(\mathrm{C_p}, C_q\right)}{\sigma\left(C_p\right)} + \frac{\mathrm{s}\left(\mathrm{C_p}, C_q\right)}{\sigma\left(C_q\right)} = \sum_{i=1}^{K} \frac{s\left(C_i, \bar{C}_i\right)}{\sigma\left(C_i\right)} \tag{1}$$

where

$$\sigma\left(C_i\right) = \begin{cases} |C_i| & \text{for Ratio Cut} \\ \sum_{j \in C_i} d_j & \text{for Normalized Cut} \end{cases} \tag{2}$$

*Corresponding Author

and K is the number of clusters, C_i is the i-th cluster, $\bar{C}_i$ is the complement of subset C_i in graph G, $s(A,B) = \sum_{i\in A}\sum_{j\in B} W_{ij}$ and $d_i = \sum_j W_{ij}$.

We now reformulate the objective functions for these different graph cuts. Let $\hat{\mathbf{q}}_i$ (i=1, ..., K) be the cluster indicators where the j-th element of $\hat{\mathbf{q}}_i$ is 1 if the j-th data vector $\mathbf{x}_j$ belongs to cluster i, and 0 otherwise. For instance, if we assume those 1s are *adjacent* and the size of i-th cluster is n_i, then

$$\hat{\mathbf{q}}_i = (0, \ldots, 0, \underbrace{1, \ldots, 1}_{n_i}, 0, \ldots, 0)^T \tag{3}$$

Note that

$$\hat{\mathbf{q}}_i^T \hat{\mathbf{q}}_j = \begin{cases} n_i & i = j \\ 0 & i \neq j \end{cases} \tag{4}$$

for $1 \leq i, j \leq K$. We can then observe that

$$\begin{array}{l} s(C_i, C_i) = \hat{\mathbf{q}}_i^T W \hat{\mathbf{q}}_i \\ \sum_{j\in C_i} d_j = \hat{\mathbf{q}}_i^T D \hat{\mathbf{q}}_i \\ s(C_i, \bar{C}_i) = \sum_{j\in C_i}\sum_{k\in \bar{C}_i} W_{jk} = \hat{\mathbf{q}}_i^T (D - W)\hat{\mathbf{q}}_i \end{array} \tag{5}$$

where D is a diagonal matrix with the i-th element d_i. As a result, we can write the clustering objective functions Eq. (1) in the following way:

$$\begin{array}{l} J_{rcut} = \sum_{i=1}^{K} \frac{\hat{\mathbf{q}}_i^T (D-W)\hat{\mathbf{q}}_i}{\hat{\mathbf{q}}_i^T \hat{\mathbf{q}}_i} \\ J_{ncut} = \sum_{i=1}^{K} \frac{\hat{\mathbf{q}}_i^T (D-W)\hat{\mathbf{q}}_i}{\hat{\mathbf{q}}_i^T D \hat{\mathbf{q}}_i} \end{array} \tag{6}$$

We can clearly see the connections and differences between these spectral clustering objective functions.

Now let us first look at how to minimize J_{rcut} with respect to $\hat{\mathbf{q}}_i$. Clearly if we restrict $\hat{\mathbf{q}}_i$ to be a discrete value vector whose elements can be either 0 or 1, then this problem becomes NP-hard. To make the optimization computation manageable, we relax the discrete constraint and seek a continuous solution of $\hat{\mathbf{q}}_i$. Note that

$$J_{rcut} = \sum_{i=1}^{K} \frac{\hat{\mathbf{q}}_i^T (D-W)\hat{\mathbf{q}}_i / n_i}{\hat{\mathbf{q}}_i^T \hat{\mathbf{q}}_i / n_i} = \sum_{i=1}^{K} \frac{\hat{\mathbf{q}}_i^T (D-W)\hat{\mathbf{q}}_i}{n_i} \tag{7}$$

due to Eq. (4). Let

$$Q = [\hat{\mathbf{q}}_1/\sqrt{n_1}, \ldots, \hat{\mathbf{q}}_K/\sqrt{n_K}] = [\mathbf{q}_1, \ldots, \mathbf{q}_K] \tag{8}$$

It is easy to see $Q^T Q = I$. The Eq. (7) becomes

$$J_{rcut} = \sum_{i=1}^{K} \mathbf{q}_i^T (D - W)\mathbf{q}_i \tag{9}$$

Then it is straightforward to get ratio cut optimization problem as follows:

$$\min_{Q^T Q = I} Tr(Q^T L Q) \tag{10}$$

where $L = D - W$ is the Laplacian graph (Chung 1997) and Tr denotes the trace operation of the matrix. The solution of Q is the collection of eigenvectors corresponding to the top smallest K eigenvalues of L. The derivation of relaxed Normalized Cut objective function is in a very similar manner, just let

$$Q = [\hat{\mathbf{q}}_1/\sqrt{d_1 n_1}, \ldots, \hat{\mathbf{q}}_K/\sqrt{d_K n_K}] \tag{11}$$

and we again get the same Eq. (10).

Since Q is now in relaxed continuous form, to get the final cluster solution, it is a common practice to apply K-Means to Q to get the final discrete solution.

To conclude this section, we summarize the main steps of Normalized Cut clustering in Algorithm 1 (Shi and Malik 2000).

Algorithm 1: Normalized Cut Clustering

Input: Data vector matrix $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T$, Number of clusters K
Output: Clusters C_1,...,C_K
Construct a similarity matrix S and weight matrix W.
Compute the Laplacian matrix L.
Compute the first K generalized eigenvectors $\mathbf{q}_1$,...,$\mathbf{q}_K$ of the generalized eigenproblem $L\mathbf{q} = \lambda D\mathbf{q}$.
Let $Q \in \mathbb{R}^{n\times K}$ be the matrix containing the vectors $\mathbf{q}_1$,...,$\mathbf{q}_K$ as columns.
For $i = 1, ..., n$, let $\mathbf{y}_i \in \mathbb{R}^K$ be the vector corresponding to the i-th row of Q.
Cluster the points $(\mathbf{y}_i)_{i=1,...,n}$ in $\mathbb{R}^K$ with the K-Means algorithms into clusters C_1,...,C_K.

How K-Means Works

In this section we re-derive K-Means in a way that motivates and explains our subsequent contribution.

Given the spectral vectors $\{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$, K-Means clustering aims to partition the n vectors into K sets, $C = \{C_1, ..., C_K\}$ so as to minimize the within-cluster sum of squares, mathematically, the solution of K-Means is $\arg\min_C \sum_{i=1}^{K} \sum_{\mathbf{y}_j \in C_i} \|\mathbf{y}_j - \mathbf{u}_i\|_2^2$, where $\mathbf{u}_i$ is the mean of vectors in C_i. The common way to solve it is via the EM algorithm, which repeats the cluster assignment (assign each observation to the cluster with the closest mean) and the update (calculate the new means to be the centroid of the observations in the cluster) processes (MacKay 2003).

The above K-Means objective function can be written in another way. Let $Q \in \mathbb{R}^{n\times K}$ be the eigenvector matrix, then the K-means objective function is

$$\min_{G\in Ind, H} \|Q - GH\|_F^2 , \tag{12}$$

where $G \in Ind$ denotes G is an indicator matrix. $G = [\mathbf{g}_1, \ldots, \mathbf{g}_n]^T$ and the unique 1 in each row vector $\mathbf{g}_i$ indicates its cluster membership.

In EM, we optimize this objective function by employing iterative alternative optimization steps:

When H is fixed, the solution to the indicator matrix G is

$$G_{ij} = \begin{cases} 1, & j = \arg\min_k \|\mathbf{q}_i - \mathbf{h}_k\|_F^2 \\ 0, & else \end{cases} \tag{13}$$

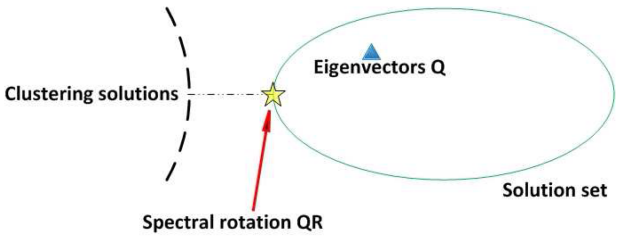

Figure 1: A demonstration of our motivation. The curve on the left represents the discrete clustering solutions. The green ellipse represents the solution set due to the orthonormal matrix R. The blue triangle represents the solution yielded via solving relaxed spectral clustering problem. The yellow star represents the solution we get via the proper rotation.

When G is fixed, it is easy to get the solution as a classic regression problem. From $G^T(GH - Q) = 0$, it is easy to get

$$H = (G^T G)^{-1} G^T Q\,. \tag{14}$$

These steps repeat until convergence and the yielded discrete $\mathbf{g}_i$s provide the cluster membership for each $\mathbf{x}_i$.

Notice that Eq. (12) only guarantees that GH combined best approximates the relaxed continuous vector matrix Q in terms of ℓ_2 distance, there is no guarantee that such yielded G best approximates Q**!** Indeed, such solution G by K-Means *generally* is **not** the one that best approximates Q. This leads to the question: how can we find a better G?

Why Spectral Rotation Works Better

We now introduce our contribution: the application of spectral rotation to spectral clustering. In the literature, research papers related to spectral rotation include multi-class clustering (Yu and Shi 2003), self-tuning spectral clustering (Zelnik-manor and Perona 2004). While these two papers also take advantage of the spectral solution invariance property, neither of them use this property to explicitly propose a new spectral clustering method.

Let us first inspect the Eq. (10) again, note that the relaxed solution to Eq. (10) is not unique. Indeed, for any solution Q, QR is another solution, where R is an arbitrary orthonormal matrix. Our goal is to find a proper orthonormal R such that the resulting QR are closer to the discrete indicator matrix solution set (in terms of ℓ_2 distance) than the Q in K-Means. Fig 1 demonstrates our idea, the rotated spectral vectors are geometrically closer to the discrete indicator matrix solution set than the traditional spectral vectors.

Similar to K-Means, we aim to find the optimal G and therefore to minimize the ℓ_2 distance between QR and G, in other words,

$$\begin{aligned} \min_{\mathrm{G,R}} \quad & \|QR - G\|_F^2 \\ s.t. \quad & G \in Ind, R^T R = I \end{aligned} \tag{15}$$

It is easy to see Eq. (15) is equivalent to the following one:

$$\begin{aligned} \min_{\mathrm{G,R}} \quad & \left\|Q - GR^T\right\|_F^2 \\ s.t. \quad & G \in Ind, R^T R = I \end{aligned} \tag{16}$$

and can be further re-written in the following form:

$$\begin{aligned} \min_{\mathrm{G,R}} \quad & \|Q - GR\|_F^2 \\ s.t. \quad & G \in Ind, R^T R = I \end{aligned} \tag{17}$$

Note that Eq. (17) and Eq. (12) are very similar, except that the orthonormal constraint imposed on the R. This is also the key difference between spectral rotation and K-Means, the additional orthonormal constraint guarantees that G best approximates QR among *all* discrete cluster membership indicator matrices, while QR is the optimal solution to Eq. (10)! It is expected and also we will demonstrate in the experiment section, such feature would lead to better optimization for both graph cut objective functions and improvement in clustering accuracy for spectral clustering.

We use the same alternative optimization method for objective function in Eq. (17).
When R is fixed,

$$G_{ij} = \begin{cases} 1, & j = \arg\min_k \|\mathbf{q}_i - \mathbf{r}_k\|_F^2 \\ 0, & else \end{cases} \tag{18}$$

When G is fixed,

$$R = UV^T \tag{19}$$

where U and V are left and right parts of the SVD decomposition of $G^T Q$. We would prove Theorem 1 to explain Eq. (19).

since Q and G are fixed at this step, Eq. (17) is equivalent to the following equation due to the trace norm property

$$\max_{R^T R = I} Tr(R^T M) \tag{20}$$

where $M = G^T Q$.

Theorem 1 *Given the objective function in Eq. (20), the optimal value is*

$$R = UV^T \tag{21}$$

where U and V are left and right singular values of SVD decomposition of M, which is defined in Eq. (20).

Proof

Suppose SVD of M is $M = U\Gamma V^T$, then we have

$$\begin{aligned} & Tr(R^T M) = Tr(R^T U \Gamma V^T) = Tr(\Gamma V^T R^T U) \\ & = Tr(\Gamma B) = \sum_i \sigma_{ii} b_{ii} \end{aligned} \tag{22}$$

where $B = V^T R^T U$,σ_{ii} and b_{ii} are the (i,i) elements of Γ and B. It is not difficult to verify B is orthonormal as

$$BB^T = V^T R^T U U^T R V = I \tag{23}$$

Therefore $-1 \le b_{ij} \le 1$. $\sigma_{ii} \ge 0$ since σ_{ii} is singular value of Γ, therefore

$$Tr(R^T M) = \sum_i \sigma_{ii} b_{ii} \le \sum_i \sigma_{ii} \tag{24}$$

The equality holds when B is the identity matrix, i.e, $V^T R^T U = I$. Therefore,

$$R = UV^T. \tag{25}$$

This completes our proof. □

The optimization of G and R repeats until convergence criteria is satisfied(the elements in indicator matrix G no longer change). The details of our spectral rotation algorithm are summarized in Algorithm 2. Note that the computation complexity of our algorithm is the same as K-Means. Given spectral vector matrix $Q \in \mathbb{R}^{n\times K}$, the time cost of our algorithm is $O(tnK^2)$, where t is the number of iterations. Therefore, our algorithm can be applied to a wide range of data sets.

Algorithm 2: Spectral Rotation Algorithm for Spectral Clustering

Input: Spectral vector matrix $Q \in \mathbb{R}^{n\times K}$, Maximum number of iteration T
Output: Cluster indicator matrix G
Construct a random initial indicator matrix G.
while *Convergence criteria not satisfied and number of iteration* $\leq T$ **do**
  For fixed G,update R according to Eq. (19).
  For fixed R,update G according to Eq. (18).
end

Experimental Results

In this section, we will evaluate the performance of the proposed method on benchmark data sets. We compare the proposed method with K-Means, NMF (Lee and Seung 2001), NMF with non-negative constraint (NMFNC) (Ding, Li, and Jordan 2008), Normalized Cut (Shi and Malik 2000) and Ratio Cut (Hagen and Kahng 1992). Note that Normalized Cut and Ratio Cut both use K-Means in their algorithms, so we denote them NC+KM and RC+KM respectively.

There are in total 12 data sets used in our experiment section, which includes 9 image ones: AR (Martinez and Kak 2001), AT&T (Samaria and Harter 1994), COIL20 (Nene, Nayar, and Murase 1996), JAFFE (Lyons et al. 1998), MNIST (LeCun et al.), PIE (Sim, Baker, and Bsat 2002), UMIST, Yale and YaleB (Georghiades and et al. 2001). The other 3 non-image ones are from UCI machine learning repository (Frank and Asuncion 2010): Abalone, Ecoli, Scale.

Table 1 summarizes the characteristics of the data sets used in the experiments.

Parameter Setting

Since this paper is mainly demonstrating the advantage of spectral rotation over K-Means in dealing with relaxed spectral vectors, we set the parameters to construct W as follows: we use the heat kernel and ℓ_2 distance, KNN neighborhood mode with K rounded to the average number of samples in each cluster for each data set. We tune the width of the neighborhood σ from list $\{1, 10, 100, 1000\}$.

For K-Means and NMF, we set the number of clusters equal to the ground truth.

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

Data set	No. of Observations	Dimensions	Classes
AR	2600	792	100
AT&T	400	168	40
COIL20	1440	1024	20
JAFFE	360	90	15
MNIST	150	196	10
PIE	680	256	68
UMIST	360	168	20
Yale	165	256	11
YaleB	1140	1024	38
Abalone	4177	8	3
Ecoli	327	7	5
Scale	625	4	3

Table 1: Description of Data Sets

Objective Function Evaluation

First, we compare our method and K-Means on the spectral clustering objective function values defined in Eq. (6), which shows spectral rotation *generally* yields a better cut than K-Means in terms of the objective functions.

Table 2 lists the normalized cut and ratio cut objective function values and the standard deviation on the 12 data sets we mentioned in previous part. We use $\simeq 0$ for the cases when standard deviation is less than 10^{-4} to save space. We do the one sided two-sample T test based on the 20 simulations on each data to test whether these cut objective function values using spectral rotation is significantly lower than the corresponding values via K-Means. Statistical significance test results at the 0.05 level are indicated in column T1.

Sometimes the T test is not able to detect the difference between the means of two groups when grouped standard deviation is relatively large. To overcome this, in this paper, we also use Mann-Whitney U test (also called Wilcoxon rank-sum test) (Mann and Whitney 1947) to assess whether objective function values with K-Means tend to have larger values than the spectral rotation group. This test is a nonparametric statistical hypothesis test and especially helpful for two reasons: first, our experiment process and results satisfy the general assumptions of the test; second, the test uses sum of rank for the test statistics, which is able to detect the cases where samples from one group are *generally* larger than the other group. We use T2 to denote the U test, where the null hypothesis that the algorithms are equal with the same type-I error rates.

From Table 2, we can see spectral rotation significantly outperforms K-Means method on 9 out of the 12 benchmark data sets for T-test. What is more important, based on U test result, for both Normalized Cut and Ratio Cut on all 12 such data sets, spectral rotation always yields a low objective function value than K-Means. This demonstrates that spectral rotation has better cut than K-Means in terms of the objective functions on these data sets.

Table 2: Objective Cut Function Values on Data Sets

(a) Normalized Cut

Data	K-means	SR	T-test	U Test
AR	53.62 ± 0.24	31.34 ± 0.53	Y	Y
AT&T	4.93 ± 0.67	3.91 ± 0.65	Y	Y
COIL20	1.28 ± 0.13	0.78 ± 0.12	Y	Y
JAFFE	1.16 ± 0.52	0.82 ± 0.55	N	Y
MNIST	5.08 ± 0.57	4.40 ± 0.56	N	Y
PIE	22.22 ± 1.77	10.38 ± 2.52	Y	Y
UMIST	2.32 ± 0.66	1.67 ± 0.59	Y	Y
Yale	6.52 ± 0.98	5.85 ± 0.72	N	Y
YaleB	21.89 ± 1.04	16.83 ± 1.21	Y	Y
Abalone	0.15 ± 0.01	0.08 ± 0.01	Y	Y
Ecoli	2.38 ± 0.69	1.48 ± 0.58	Y	Y
Scale	0.41 ± 0.01	0.38 ± 0.01	Y	Y

(b) Ratio Cut

Data	K-means	SR	T-test	U test
AR	0.35 ± 0.02	0.25 ± 0.03	Y	Y
AT&T	29.69 ± 4.47	21.41 ± 3.85	Y	Y
COIL20	3.54 ± 0.88	2.65 ± 0.23	Y	Y
JAFFE	10.87 ± 3.04	7.84 ± 2.13	Y	Y
MNIST	50.42 ± 9.69	35.99 ± 7.56	Y	Y
PIE	121.24 ± 15.54	44.45 ± 8.63	Y	Y
UMIST	11.62 ± 5.22	6.20 ± 1.88	Y	Y
Yale	56.60 ± 9.75	51.95 ± 7.43	N	Y
YaleB	82.71 ± 12.71	60.50 ± 11.68	Y	Y
Abalone	0.11 ± 0.01	0.08 ± 0.01	Y	Y
Ecoli	8.76 ± 2.62	7.43 ± 1.24	N	Y
Scale	2.61 ± 0.05	2.56 ± 0.12	N	Y

Evaluation Metrics

To evaluate the clustering results, we adopt the three widely used clustering performance measures.

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:

$$Acc = \frac{\sum_{i=1}^{n} \delta(map(r_i), l_i)}{n}, \tag{26}$$

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 samples, $\delta(x, y)$ 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 (NMI) is used for determining the quality of clusters. Given a clustering result, the NMI is estimated by

$$NMI = \frac{\sum_{i=1}^{c}\sum_{j=1}^{c} n_{i,j} \log \frac{n_{i,j}}{n_i \hat{n}_j}}{\sqrt{(\sum_{i=1}^{c} n_i \log \frac{n_i}{n})(\sum_{j=1}^{c} \hat{n}_j \log \frac{\hat{n}_j}{n})}} \tag{27}$$

where n_i denotes the number of data contained in the cluster $C_i (1 \leq i \leq c)$, $\hat{n}_j$ is the number of data belonging to the $L_j (1 \leq j \leq c)$, and $n_{i,j}$ denotes the number of data that are in the intersection between cluster C_i and the class L_j.

Purity measures the extent to which each cluster contained data points from primarily one class. The purity of a clustering is observed by the weighted sum of individual cluster purity values, given as follows:

$$Purity = \sum_{i=1}^{K} \frac{n_i}{n} P(S_i), \quad P(S_i) = \frac{1}{n_i} \max_j (n_i^j) \tag{28}$$

where S_i is a particular cluster size of n_i, n_i^j is the number of the i-th input class that was assigned to the j-th cluster. K is the number of the clusters and n is the total number of the data points.

Clustering Results

Table 3 displays the average clustering results on different data sets for all the methods we mentioned. We can see that spectral clustering with different cuts using our method has superior performance on most benchmark data sets. We do not list many competitive methods here for 2 primary reasons: (1) The purpose of this paper is to show our proposed approach can bring significant improvement over conventional spectral clustering approaches, not to claim this is the best method among all clustering methods. In fact, if conventional spectral clustering methods outperforms other methods, then our approach has a high probability to do even better. (ii) This gives our figure better visual effects.

Conclusion

In this paper, we employ spectral rotation (SR) in spectral clustering to convert relaxed continuous spectral vectors into a cluster membership indicator matrix in a more effective way than traditional K-Means. Our method takes advantage of the rotation invariant property of the spectral solution vectors, but still uses an iterative optimization method to get the final solution. Such a solution better approximates graph cut objective functions and generally has better performance than the corresponding K-Means method. We also establish a theoretical connection between our method and simple K-Means, which explains the performance improvement. We are then able to demonstrate this improvement experimentally, both in terms of solutions associated with smaller objective function values, as well as improved clustering metrics. Our proposed method *consistently* yields better graph cut objective function values and clustering performance, shows its significance in both theory and application.

Acknowledgements

This work was partially funded by NSF CCF-0830780, CCF-0917274, DMS-0915228, and IIS-1117965.

References

Chan, P.; Schlag, D.; and Zien, J. 1994. Spectral k-way ratio-cut partitioning and clustering. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 13(9):1088–1096.

(c) Accuracy

Data	NC+KM	NC+SR	RC+KM	RC+SR	K-Means	NMF	NMFNC
AR	0.1062 ± 0.01	**0.1423** ± 0.02	0.1055 ± 0.02	0.1383 ± 0.02	0.1315 ± 0.02	0.1347 ± 0.02	0.1389 ± 0.02
AT&T	0.6600 ± 0.03	**0.7110** ± 0.01	0.6598 ± 0.03	0.7080 ± 0.02	0.6790 ± 0.03	0.6720 ± 0.02	0.6685 ± 0.02
Coil20	0.6121 ± 0.02	**0.6753** ± 0.02	0.6228 ± 0.02	0.6694 ± 0.03	0.6092 ± 0.05	0.6034 ± 0.03	0.6143 ± 0.04
Jaffe	0.7317 ± 0.02	0.7746 ± 0.02	0.7647 ± 0.02	**0.7981** ± 0.02	0.7543 ± 0.02	0.7598 ± 0.02	0.7654 ± 0.02
MNIST	0.7134 ± 0.04	0.7316 ± 0.04	0.7145 ± 0.05	**0.7441** ± 0.05	0.6913 ± 0.05	0.7014 ± 0.05	0.5267 ± 0.03
PIE	0.1956 ± 0.03	0.2276 ± 0.01	0.1818 ± 0.01	**0.2344** ± 0.01	0.2250 ± 0.01	0.2245 ± 0.02	0.2126 ± 0.01
Umist	0.5212 ± 0.03	0.5612 ± 0.03	0.5421 ± 0.03	**0.5717** ± 0.03	0.4953 ± 0.03	0.5024 ± 0.03	0.5133 ± 0.04
Yale	0.3319 ± 0.03	0.3502 ± 0.02	0.3592 ± 0.03	**0.3724** ± 0.02	0.3418 ± 0.02	0.3524 ± 0.02	0.3542 ± 0.02
YaleB	0.1466 ± 0.01	**0.1527** ± 0.01	0.1235 ± 0.01	0.1442 ± 0.01	0.1250 ± 0.01	0.1320 ± 0.01	0.1474 ± 0.01
Abalone	0.4652 ± 0.01	0.5237 ± 0.02	0.4873 ± 0.02	**0.5437** ± 0.02	0.5083 ± 0.02	0.5194 ± 0.02	0.5243 ± 0.03
Ecoli	0.5394 ± 0.03	0.5785 ± 0.03	0.5506 ± 0.02	**0.6130** ± 0.03	0.4967 ± 0.03	0.5132 ± 0.02	0.5383 ± 0.04
Scale	0.5364 ± 0.03	**0.5824** ± 0.03	0.5125 ± 0.02	0.5372 ± 0.02	0.5174 ± 0.02	0.5347 ± 0.02	0.5572 ± 0.03

(d) NMI

Data	NC+KM	NC+SR	RC+KM	RC+SR	K-Means	NMF	NMFNC
AR	0.3063 ± 0.02	0.3943 ± 0.02	0.3392 ± 0.03	0.3843 ± 0.03	**0.4273** ± 0.03	0.4163 ± 0.02	0.4028 ± 0.02
AT&T	0.8272 ± 0.01	**0.8710** ± 0.01	0.8391 ± 0.01	0.8640 ± 0.01	0.8531 ± 0.02	0.8490 ± 0.02	0.8376 ± 0.01
Coil20	0.7482 ± 0.03	**0.8141** ± 0.04	0.7468 ± 0.03	0.7748 ± 0.02	0.7432 ± 0.03	0.7384 ± 0.03	0.7532 ± 0.02
Jaffe	0.7822 ± 0.03	0.8005 ± 0.03	0.7932 ± 0.03	**0.8174** ± 0.03	0.7865 ± 0.03	0.7885 ± 0.03	0.7943 ± 0.03
MNIST	0.7062 ± 0.03	0.7253 ± 0.03	0.7147 ± 0.02	**0.7381** ± 0.02	0.7005 ± 0.03	0.7025 ± 0.04	0.5217 ± 0.04
PIE	0.3883 ± 0.09	0.4970 ± 0.01	0.3658 ± 0.01	**0.506** ± 0.01	0.4950 ± 0.01	0.4980 ± 0.01	0.4600 ± 0.03
Umist	0.5212 ± 0.03	0.5612 ± 0.03	0.5421 ± 0.03	**0.5717** ± 0.03	0.4953 ± 0.03	0.5024 ± 0.03	0.5133 ± 0.04
Yale	0.3319 ± 0.03	0.3502 ± 0.02	0.3592 ± 0.03	**0.3724** ± 0.02	0.3418 ± 0.02	0.3524 ± 0.02	0.3542 ± 0.02
YaleB	0.7010 ± 0.02	**0.7409** ± 0.04	0.7128 ± 0.02	0.7563 ± 0.02	0.6751 ± 0.02	0.6772 ± 0.02	0.7017 ± 0.02
Abalone	0.0860 ± 0.01	0.1324 ± 0.01	0.0970 ± 0.01	0.1472 ± 0.01	0.1153 ± 0.01	0.1543 ± 0.01	**0.1572** ± 0.01
Ecoli	0.7162 ± 0.02	0.7452 ± 0.02	0.7136 ± 0.01	**0.7730** ± 0.02	0.6784 ± 0.02	0.6935 ± 0.02	0.7142 ± 0.03
Scale	0.1074 ± 0.01	**0.1435** ± 0.01	0.097 ± 0.01	0.1325 ± 0.01	0.1293 ± 0.01	0.089 ± 0.01	0.1274 ± 0.01

(e) Purity

Data	NC+KM	NC+SR	RC+KM	RC+SR	K-Means	NMF	NMFNC
AR	0.1108 ± 0.01	**0.1473** ± 0.02	0.1034 ± 0.01	0.1282 ± 0.02	0.1323 ± 0.02	0.1413 ± 0.02	0.1358 ± 0.02
AT&T	0.6820 ± 0.03	**0.7340** ± 0.02	0.6930 ± 0.02	0.7290 ± 0.02	0.7282 ± 0.03	0.7214 ± 0.02	0.7052 ± 0.02
Coil20	0.6115 ± 0.02	**0.6836** ± 0.03	0.6267 ± 0.02	0.6534 ± 0.02	0.6084 ± 0.04	0.5973 ± 0.03	0.6124 ± 0.03
Jaffe	0.7596 ± 0.02	0.7908 ± 0.02	0.8032 ± 0.03	**0.8152** ± 0.03	0.7829 ± 0.03	0.7884 ± 0.03	0.8065 ± 0.03
MNIST	0.7124 ± 0.03	0.7253 ± 0.03	0.7185 ± 0.03	**0.7326** ± 0.02	0.7113 ± 0.04	0.7037 ± 0.03	0.5333 ± 0.03
PIE	0.2265 ± 0.04	0.2559 ± 0.01	0.2041 ± 0.01	**0.2600** ± 0.01	0.2550 ± 0.01	0.2560 ± 0.01	0.2530 ± 0.01
Umist	0.5643 ± 0.03	0.6135 ± 0.05	0.5799 ± 0.03	**0.6262** ± 0.01	0.5513 ± 0.02	0.5524 ± 0.02	0.5674 ± 0.03
Yale	0.3398 ± 0.03	0.3578 ± 0.04	0.3599 ± 0.03	**0.3837** ± 0.04	0.3516 ± 0.02	0.3742 ± 0.02	0.3647 ± 0.02
YaleB	0.1564 ± 0.01	**0.1618** ± 0.01	0.1370 ± 0.01	0.1552 ± 0.01	0.1324 ± 0.01	0.1435 ± 0.01	0.1572 ± 0.01
Abalone	0.4360 ± 0.02	0.4740 ± 0.02	0.4680 ± 0.01	0.5042 ± 0.02	0.4804 ± 0.02	0.5142 ± 0.02	**0.5245** ± 0.02
Ecoli	0.5819 ± 0.03	0.6123 ± 0.02	0.5822 ± 0.02	**0.6434** ± 0.03	0.5458 ± 0.02	0.5643 ± 0.02	0.5836 ± 0.03
Scale	0.6554 ± 0.03	**0.6943** ± 0.03	0.6341 ± 0.03	0.6724 ± 0.03	0.6674 ± 0.03	0.6583 ± 0.03	0.6721 ± 0.03

Table 3: Clustering Performance on Benchmark Data Sets

Chung, F. 1997. *Spectral Graph Theory*. American Mathematical Society.

Dhillon, I.; Guan, Y.; and Kulis, B. 2004. Kernel kmeans, spectral clustering and normalized cuts. In *KDD*, 551–556.

Ding, C.; He, X.; Zha, H.; Gu, M.; and Simon, H. 2001. A min-max cut algorithm for graph partitioning and data clustering. In *ICDM*, 107–114.

Ding, C.; He, X.; and Simon, H. 2005. On the equivalence of nonnegative matrix factorization and spectral clustering. In *SDM*, 606–610.

Ding, C.; Li, T.; and Jordan, M. 2008. Nonnegative matrix factorization for combinatorial optimization: Spectral clustering, graph matching, and clique finding. In *ICDM*, 183–192.

Frank, A., and Asuncion, A. 2010. Uci machine learning repository.

Georghiades, A., and et al. 2001. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 23(6):643–660.

Hagen, L., and Kahng, A. 1992. New spectral methods for ratio cut partioning and clustering. *IEEE Transactions on Computer-Aided Design* 11(6):1074–1085.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11):2278–2324.

Lee, D., and Seung, H. 2001. Algorithms for non-negative matrix factorization. In *NIPS*, 556–562.

Luo, D.; Huang, H.; Ding, C.; and Nie, F. 2010. On the eigenvectors of p-laplacian. *Machine Learning* 81(1):37–51.

Luxberg, U. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17(4):395–416.

Lyons, M.; Akamatsu, S.; Kamachi, M.; and Gyoba, J. 1998. Coding facial expressions with gabor wavelets. In *3rd IEEE International Conference on Automatic Face and Gesture Recognition*.

MacKay, D. 2003. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

Mann, H., and Whitney, D. 1947. On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics* 18(1):50–60.

Martinez, A., and Kak, A. 2001. Pca versus lda. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2(23):228–233.

Nene, S.; Nayar, S.; and Murase, H. 1996. Technical report, Columbia University.

Ng, A.; Jordan, M.; and Weiss, Y. 2002. On spectral clustering: Analysis and an algorithm. In *NIPS*, 849–856.

Nie, F.; Ding, C.; Luo, D.; and Huang, H. 2010. Improved minmax cut graph clustering with nonnegative relaxation. In *ECMLPKDD*, 451–466.

Nie, F.; Wang, H.; Huang, H.; and Ding, C. 2011. Unsupervised and semi-supervised learning via l_1-norm graph. In *ICCV*, 2268–2273.

Samaria, F., and Harter, A. 1994. Parameterisation of a stochastic model for human face identification. In *2nd IEEE Workshop on Applications of Computer Vision*.

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(8):888–905.

Sim, T.; Baker, S.; and Bsat, M. 2002. The cmu pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25(12):1615–1618.

Yu, X., and Shi, J. 2003. Multiclass spectral clustering. In *ICCV*, 313–319.

Zelnik-manor, L., and Perona, P. 2004. Self-tuning spectral clustering. In *NIPS*, 1601–1608.

Zha, H.; He, X.; Ding, C.; and Simon, H. 2001. Spectral relaxation for k-means clustering. In *NIPS*, 1057–1064.