The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Unified Spectral Clustering with Optimal Graph

Zhao Kang,[1] Chong Peng,[2] Qiang Cheng,[3] Zenglin Xu[1]*
[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China, China
[2]Department of Computer Science, Southern Illinois University, Carbondale, USA
[3]Institute of Biomedical Informatics and Department of Computer Science, University of Kentucky, Lexington, USA
zkang@uestc.edu.cn, pchong@siu.edu, qiang.cheng@uky.edu, zlxu@uestc.edu.cn

Abstract

Spectral clustering has found extensive use in many areas. Most traditional spectral clustering algorithms work in three separate steps: similarity graph construction; continuous labels learning; discretizing the learned labels by k-means clustering. Such common practice has two potential flaws, which may lead to severe information loss and performance degradation. First, predefined similarity graph might not be optimal for subsequent clustering. It is well-accepted that similarity graph highly affects the clustering results. To this end, we propose to automatically learn similarity information from data and simultaneously consider the constraint that the similarity matrix has exact c connected components if there are c clusters. Second, the discrete solution may deviate from the spectral solution since k-means method is well-known as sensitive to the initialization of cluster centers. In this work, we transform the candidate solution into a new one that better approximates the discrete one. Finally, those three subtasks are integrated into a unified framework, with each subtask iteratively boosted by using the results of the others towards an overall optimal solution. It is known that the performance of a kernel method is largely determined by the choice of kernels. To tackle this practical problem of how to select the most suitable kernel for a particular data set, we further extend our model to incorporate multiple kernel learning ability. Extensive experiments demonstrate the superiority of our proposed method as compared to existing clustering approaches.

Introduction

Clustering is a fundamental technique in machine learning, pattern recognition, and data mining (Huang et al. 2017). In past decades, a variety of clustering algorithms have been developed, such as k-means clustering and spectral clustering.

With the benefits of simplicity and effectiveness, k-means clustering algorithm is often adopted in various real-world problems. To deal with the nonlinear structure of many practical data sets, kernel k-means (KKM) algorithm has been developed (Schölkopf, Smola, and Müller 1998), where data points are mapped through a nonlinear transformation into a higher dimensional feature space in which the data points are linearly separable. KKM usually achieves better performance than the standard k-means. To cope with noise and outliers, robust kernel k-means (RKKM) (Du et al. 2015) algorithm has been proposed. In this approach, the squared ℓ_2 norm of error construction term is replaced by $\ell_{2,1}$ norm. RKKM demonstrates superior performance on a number of benchmark data sets. The performance of such model-based methods heavily depends on whether the data fit the model. Unfortunately, in most cases, we do not know the distribution of data in advance. To some extent, this problem is alleviated by multiple kernel learning.

Spectral clustering is another widely used clustering method (Kumar, Rai, and Daume 2011). It enjoys the advantage of exploring the intrinsic data structures by exploiting the different similarity graphs of data points (Yang et al. 2015). There are three kinds of similarity graph constructing strategies: k-nearest-neighborhood (knn); ϵ-nearest-neighborhood; The fully connected graph. Here, some open issues arise (Huang, Nie, and Huang 2015): 1) how to choose a proper neighbor number k or radius ϵ; 2) how to select an appropriate similarity metric to measure the similarity among data points; 3) how to counteract the adverse effect of noise and outliers; 4) how to tackle data with structures at different scales of size and density. Unfortunately, all of these issues heavily influence the clustering results (Zelnik-Manor and Perona 2004). Nowadays, many data are often high dimensional, heterogeneous, and without prior knowledge, and it is therefore a fundamental challenge to define a pairwise similarity graph for effective spectral clustering.

Recently, (Zhu, Change Loy, and Gong 2014) construct robust affinity graphs for spectral clustering by identifying discriminative features. It adopts a random forest approach based on the motivation that tree leaf nodes contain discriminative data partitions, which can be exploited to capture subtle and weak data affinity. This approach shows better performance than other state-of-the-art methods including the Euclidean-distance-based knn (Wang et al. 2008), dominant neighbourhoods (Pavan and Pelillo 2007), consensus of knn (Premachandran and Kakarala 2013), and non-metric based unsupervised manifold forests (Pei, Kim, and Zha 2013).

The second step of spectral clustering is to use the spectrum of the similarity graph to reveal the cluster structure of the data. Due to the discrete constraint on the cluster labels,

*Corresponding author.

this problem is NP-hard. To obtain a feasible approximation solution, spectral clustering solves a relaxed version of this problem, i.e., the discrete constraint is relaxed to allow continuous values. It first performs eigenvalue decomposition on the Laplacian matrix to generate an approximate indicator matrix with continuous values. Then, k-means is often implemented to produce final clustering labels (Huang, Nie, and Huang 2013). Although this approach has been widely used in practice, it may exhibit poor performance since the k-means method is well-known as sensitive to the initialization of cluster centers (Ng et al. 2002).

To address the aforementioned problems, in this paper, we propose a unified spectral clustering framework. It jointly learns the similarity graph from the data and the discrete clustering labels by solving an optimization problem, in which the continuous clustering labels just serve as intermediate products. To the best of our knowledge, this is the first work that combine the three steps into a single optimization problem. As we show later, it is not trivial to unify them. The contributions of our work are as follows:

1. Rather than using predefined similarity metrics, the similarity graph is adaptively learned from the data in a kernel space. By combining similarity learning with subsequentl clustering into a unified framework, we can ensure the optimality of the learned similarity graph.
2. Unlike existing spectral clustering methods that work in three separate steps, we simultaneously learn similarity graph, continuous labels, and discrete cluster labels. By leveraging the inherent interactions between these three subtasks, they can be boosted by each other.
3. Based on our single kernel model, we further extend it to have the ability to learn the optimal combination of multiple kernels.

Notations. Given a data set $[x_1, x_2, \cdots, x_n]$, we denote $X \in \mathcal{R}^{m\times n}$ with m features and n samples. Then the i-th sample and (i,j)-th element of matrix X are denoted by $x_i \in \mathcal{R}^{m\times 1}$ and x_{ij}, respectively. The ℓ_2-norm of a vector x is defined as $\|x\|^2 = x^\top \cdot x$, where $\top$ means transpose. The squared Frobenius norm is denoted by $\|X\|_F^2 = \sum_{ij} x_{ij}^2$. The ℓ_1-norm of matrix X is defined as the absolute summation of its entries, i.e., $\|X\|_1 = \sum_i \sum_j |x_{ij}|$. I denotes the identity matrix. Tr$(\cdot)$ is the trace operator. $Z \geq 0$ means all the elements of Z are nonnegative.

Preliminary Knowledge

Sparse Representation

Recently, sparse representation, which assumes that each data point can be reconstructed as a linear combination of the other data points, has shown its power in many tasks (Cheng et al. 2010; Peng et al. 2016). It often solves the following problem:

$$\min_Z \|X - XZ\|_F^2 + \alpha\|Z\|_1, \ s.t.\ Z \geq 0, \ diag(Z) = 0, \ (1)$$

where $\alpha > 0$ is a balancing parameter. Eq. (1) simultaneously determines both the neighboring samples of a data point and the corresponding weights by the sparse reconstruction from the remaining samples. In principle, more similar points should receive bigger weights and the weights should be smaller for less similar points. Thus Z is also called similarity graph matrix (Kang, Peng, and Cheng 2015). In addition, sparse representation enjoys some nice properties, e.g., the robustness to noise and datum-adaptive ability (Huang, Nie, and Huang 2015). On the other hand, model (1) has a drawback, i.e., it does not consider nonlinear data sets where data points reside in a union of manifolds (Kang, Peng, and Cheng 2017a).

Spectral Clustering

Spectral clustering requires Laplacian matrix $L \in \mathcal{R}^{n\times n}$ as an input, which is computed as $L = D - \frac{Z^\top + Z}{2}$, where $D \in \mathcal{R}^{n\times n}$ is a diagonal matrix with the i-th diagonal element $\sum_j \frac{z_{ij}+z_{ij}}{2}$. In traditional spectral clustering methods, similarity graph $Z \in \mathcal{R}^{n\times n}$ is often constructed in one of the three ways aforementioned. Supposing there are c clusters in the data X, spectral clustering solves the following problem:

$$\min_F Tr(F^\top LF), \quad s.t. \quad F \in Idx, \qquad (2)$$

where $F = [f_1, f_2, \cdots, f_n]^\top \in \mathcal{R}^{n\times c}$ is the cluster indicator matrix and $F \in Idx$ represents the clustering label vector of each point $f_i \in \{0,1\}^{c\times 1}$ contains one and only one element "1" to indicate the group membership of x_i. Due to the discrete constraint on F, problem (2) is NP-hard. In practice, F is relaxed to allow continuous values and solve

$$\min_P Tr(P^\top LP), \quad s.t. \quad P^\top P = I, \qquad (3)$$

where $P \in \mathcal{R}^{n\times c}$ is the relaxed continuous clustering label matrix, and the orthogonal constraint is adopted to avoid trivial solutions. The optimal solution is obtained from the c eigenvectors of L corresponding to the c smallest eigenvalues. After obtaining F, traditional clustering method, e.g., k-means, is implemented to obtain discrete cluster labels (Huang, Nie, and Huang 2013).

Although this three-steps approach provides a feasible solution, it comes with two potential risks. First, since the similarity graph computation is independent of the subsequent steps, it may be far from optimal. As we discussed before, the clustering performance is largely determined by the similarity graph. Thus, final results may be degraded. Second, the final solution may unpredictably deviate from the ground-truth discrete labels (Yang et al. 2016). To address these problems, we propose a unified spectral clustering model.

Spectral Clustering with Single Kernel

Model

One drawback of Eq. (1) is that it assumes that all the points lie in a union of independent or disjoint subspaces and are noiseless. In the presence of dependent subspaces, nonlinear manifolds and/or data errors, it may select points from different structures to represent a data point and makes the representation less informative (Elhamifar and Vidal 2009). It is

recognized that nonlinear data may represent linearity when mapped to an implicit, higher-dimensional space via a kernel function. To fully exploit data information, we formulate Eq. (1) in a general manner with a kernelization framework.

Let $\phi : \mathcal{R}^D \to \mathcal{H}$ be a kernel mapping the data samples from the input space to a reproducing kernel Hilbert space $\mathcal{R}$. Then X is transformed to $\phi(X) = [\phi(x_1), \cdots, \phi(x_n)]$. The kernel similarity between data samples x_i and x_j is defined through a predefined kernel as $K_{x_i,x_j} =< \phi(x_i), \phi(x_j) >$. By applying this kernel trick, we do not need to know the transformation ϕ. In the new space, Eq. (1) becomes (Zhang, Nie, and Xiang 2010)

$$\begin{aligned}
&\min_Z \|\phi(X) - \phi(X)Z\|_F^2 + \alpha\|Z\|_1, \\
\Longleftrightarrow &\min_Z Tr(\phi(X)^T\phi(X) - \phi(X)^T\phi(X)Z \\
&- Z^T\phi(X)^T\phi(X) + Z^T\phi(X)^T\phi(X)Z) + \alpha\|Z\|_1, \\
\Longleftrightarrow &\min_Z Tr(K - 2KZ + Z^\top KZ) + \alpha\|Z\|_1, \\
&s.t. \quad Z \geq 0, \quad diag(Z) = 0,
\end{aligned} \tag{4}$$

This model recovers the linear relations among the data in the new space, and thus the nonlinear relations in the original representation. Eq. (4) is more general than Eq. (1) and is supposed to learn arbitrarily shaped data structure. Moreover, Eq. (4) goes back to Eq. (1) when a linear kernel is applied.

To fulfill the clustering task, we propose our spectral clustering with single kernel (SCSK) model as following:

$$\begin{aligned}
&\min_{Z,F,P,Q} \underbrace{Tr(K - 2KZ + Z^\top KZ) + \alpha\|Z\|_1}_{\text{similarity learning}} \\
&+ \underbrace{\beta Tr(P^\top LP)}_{\text{continuous label learning}} + \underbrace{\gamma\|F - PQ\|_F^2}_{\text{discrete label learning}}, \\
&s.t. \quad Z \geq 0, \quad diag(Z) = 0, \\
&\quad P^\top P = I, \quad Q^\top Q = I, \quad F \in Idx,
\end{aligned} \tag{5}$$

where α, β, and γ are penalty parameters, and Q is a rotation matrix. Due to the spectral solution invariance property (Yu and Shi 2003), for any solution P, PQ is another solution. The purpose of the last term is to find a proper orthonormal Q such that the resulting PQ is close to the real discrete clustering labels. In Eq. (5), the similarity graph and the final discrete clustering labels are automatically learned from the data. Ideally, whenever data points i and j belong to different clusters, we must have $z_{ij} = 0$ and it is also true vice versa. That is to say, we have $z_{ij} \neq 0$ if and only if data points i and j are in the same cluster, or, equivalently $f_i = f_j$. Therefore, our unified framework Eq. (5) can exploit the correlation between the similarity matrix and the labels. Because of the feedback of inferred labels to induce the similarity matrix and vice versa, we say that our clustering framework has a self-taught property.

In fact, Eq. (5) is not a simple unification of the pipeline of steps. It learns a similarity graph with optimal structure for clustering. Ideally, Z should have exactly c connected components if there are c clusters in the data set (Kang, Peng, and Cheng 2017b). This is to say that the Laplacian matrix L has c zero eigenvalues (Mohar et al. 1991), i.e., the summation of the smallest c eigenvalues is zero. To ensure the optimality of the similarity graph, we can minimize $\sum_{i=1}^{c} \sigma_i(L)$. According to Ky Fan's theorem (Fan 1949), $\sum_{i=1}^{c} \sigma_i(L) = \min_{P^\top P = I} Tr(P^\top LP)$. Therefore, the spectral clustering term, i.e., the second term in Eq. (5), will ensure learned Z is optimal for clustering.

Optimization

To efficiently and effectively solve Eq. (5), we design an alternated iterative method.

Computation of Z: With F, P, Q fixed, the problem is reduced to

$$\begin{aligned}
&\min_Z Tr(K - 2KZ + Z^\top KZ) + \alpha\|Z\|_1 + \beta Tr(P^\top LP), \\
&s.t. \quad Z \geq 0, \quad diag(Z) = 0.
\end{aligned} \tag{6}$$

We introduce an auxiliary variable S to make above objective function separable and solve the following equivalent problem:

$$\begin{aligned}
&\min_Z Tr(K - 2KZ + Z^\top KZ) + \alpha\|S\|_1 + \beta Tr(P^\top LP), \\
&s.t. \quad Z \geq 0, \quad diag(Z) = 0, \quad S = Z.
\end{aligned} \tag{7}$$

This can be solved by using the augmented Lagrange multiplier (ALM) type of method. We turn to minimizing the following augmented Lagrangian function:

$$\begin{aligned}
\mathcal{L}(S, Z, Y) =& Tr(K - 2KZ + Z^\top KZ) + \alpha\|S\|_1 \\
&+ \beta Tr(P^\top LP) + \frac{\mu}{2}\|S - Z + \frac{Y}{\mu}\|_F^2,
\end{aligned} \tag{8}$$

where $\mu > 0$ is the penalty parameter and Y is the Lagrange multiplier. This problem can be minimized with respect to S, Z, and Y alternatively, by fixing the other variables.

For S, by letting $H = Z - \frac{Y}{\mu}$, it can be updated elementwisely as below

$$S_{ij} = max(|H_{ij}| - \alpha/\mu, 0) \cdot sign(H_{ij}). \tag{9}$$

For Z, by letting $E = S + \frac{Y}{\mu}$, it can be updated columnwisely as:

$$\min_{Z_i} Z_i^T(\frac{\mu}{2}I + K)Z_i + (\frac{\beta}{2}d_i^T - \mu E_i^T - 2K_{i,:})Z_i, \tag{10}$$

where $d_i \in \mathcal{R}^{n\times 1}$ is a vector with the j-th element d_{ij} being $d_{ij} = \|P_{i,:} - P_{j,:}\|^2$. It is easy to obtain Z_i by setting the derivative of Eq. (10) w.r.t. Z_i to be zero.

Computation of P: With F, Z, and Q fixed, it is equivalent to solving

$$\min_P \beta Tr(P^\top LP) + \gamma\|F - PQ\|_F^2 \; s.t. \; P^\top P = I. \tag{11}$$

The above problem with orthogonal constraint can be efficiently solved by the algorithm proposed by Wen and Yin

Algorithm 1 The algorithm of SCSK

Input: Kernel matrix K, parameters $\alpha > 0, \beta > 0, \gamma > 0, \mu > 0$.
Initialize: Random matrices Z, P, and Q. $Y = 0$ and $F = 0$.
REPEAT

1: Update S according to Eq. (9).
2: $S = S - diag(diag(S))$ and $S = max(S, 0)$.
3: Update Z according to Eq. (10).
4: $Y = Y + \mu(S - Z)$.
5: Update P by solving the problem of Eq. (11).
6: Update Q according to Eq. (13).
7: Update F according to Eq. (16).

UNTIL stopping criterion is met.

(Wen and Yin 2013).

Computation of Q: With F, Z, and P fixed, we have

$$\min_{Q} \|F - PQ\|_F^2 \quad s.t. \quad Q^\top Q = I. \tag{12}$$

It is the orthogonal Procrustes problem (Schönemann 1966), which admits a closed-form solution. The solution is

$$Q = UV^\top, \tag{13}$$

where U and V are left and right parts of the SVD decomposition of $F^\top P$.

Computation of F: With Z, P and Q fixed, the problem becomes

$$\min_{F} \|F - PQ\|_F^2, \quad s.t. \quad F \in Idx. \tag{14}$$

Note that $Tr(F^\top F) = n$, the above subproblem can be rewritten as below:

$$\max_{F} Tr(F^\top PQ) \quad s.t. \quad F \in Idx. \tag{15}$$

The optimal solution can be easily obtained as follows:

$$F_{ij} = \begin{cases} 1, & j = \underset{k}{\operatorname{argmax}} \ (PQ)_{ik} \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

The updates of Z, P, F, and Q are coupled with each other, so we could reach an overall optimal solution. The details of our SCSK optimization are summarized in Algorithm 1.

Complexity Analysis

With our optimization strategy, the updating of S requires $\mathcal{O}(n^2)$ complexity. The solution of Q involves SVD and its complexity is $\mathcal{O}(nc^2 + c^3)$. To update P, we need $\mathcal{O}(nc^2 + c^3)$. The complexity for F is $\mathcal{O}(nc^2)$. Note that the number of clusters c is often a small number. Therefore, the main computation load is from solving Z, which involves matrix inversion. Fortunately, Z is solved in parallel.

Spectral Clustering with Multiple Kernels

Model

Although the model in Eq. (5) can automatically learn the similarity graph matrix and discrete cluster labels, its performance will strongly depend on the choice of kernels. It is often impractical to exhaustively search for the most suitable kernel. Moreover, real world data sets are often generated from different sources along with heterogeneous features. Single kernel method may not be able to fully utilize such information. Multiple kernel learning has the ability to integrate complementary information and identify a suitable kernel for a given task. Here we present a way to learn an appropriate consensus kernel from a convex combination of a number of predefined kernel functions.

Suppose there are a total number of r different kernel functions $\{K^i\}_{i=1}^r$. An augmented Hilbert space can be constructed by using the mapping of $\tilde{\phi}(x) = [\sqrt{w_1}\phi_1(x), \sqrt{w_2}\phi_2(x), ..., \sqrt{w_r}\phi_r(x)]^\top$ with different weights $\sqrt{w_i}(w_i \geq 0)$. Then the combined kernel K_w can be represented as (Zeng and Cheung 2011)

$$K_w(x, y) =< \phi_w(x), \phi_w(y) >= \sum_{i=1}^{r} w_i K^i(x, y). \tag{17}$$

Note that the convex combination of the positive semidefinite kernel matrices $\{K^i\}_{i=1}^r$ is still a positive semidefinite kernel matrix. Thus the combined kernel still satisfies Mercer's condition. Then our proposed method of spectral clustering with multiple kernels (SCMK) can be formulated as

$$\begin{aligned} \min_{Z,F,P,Q,w} \ & Tr(K_w - 2K_w Z + Z^\top K_w Z) + \alpha\|Z\|_1 + \\ & \beta Tr(P^\top LP) + \gamma\|F - PQ\|_F^2, \\ & s.t. \quad Z \geq 0, \quad diag(Z) = 0, \\ & P^\top P = I, \quad Q^\top Q = I, \quad F \in Idx, \\ & K_w = \sum_{i=1}^{r} w_i K^i, \quad \sum_{i=1}^{r} \sqrt{w_i} = 1, \quad w_i \geq 0. \end{aligned} \tag{18}$$

Now above model will learn the similarity graph, discrete clustering labels, and kernel weights by itself. By iteratively updating Z, F, and w, each of them will be iteratively refined according to the results of the others.

Optimization

In this part, we show an efficient and effective algorithm to iteratively and alternatively solve Eq. (18).

w **is fixed:** Update other variables when w is fixed: We can directly calculate K_w, and the optimization problem is exactly Eq. (5). Thus we just need to use Algorithm 1 with K_w as the input kernel matrix.

Update w**:** Optimize with respect to w when other variables are fixed: Solving Eq. (18) with respect to w can be rewritten as (Cai et al. 2013)

$$\begin{aligned} \min_{w} \ & \sum_{i=1}^{r} w_i h_i \\ s.t. \quad & \sum_{i=1}^{r} \sqrt{w_i} = 1, \quad w_i \geq 0, \end{aligned} \tag{19}$$

where

$$h_i = Tr(K^i - 2K^i Z + Z^\top K^i Z). \tag{20}$$

Algorithm 2 The algorithm of SCMK

Input: A set of kernel matrix $\{K^i\}_{i=1}^r$, parameters $\alpha > 0$, $\beta > 0$, $\gamma > 0$, $\mu > 0$.
Initialize: Random matrices Z, P, and Q. $Y = 0$ and $F = 0$. $w_i = 1/r$.
REPEAT

1: Calculate K_w by Eq. (17).
2: Do steps 1-7 in Algorithm 1.
3: Calculate h by Eq. (20).
4: Calculate w by Eq. (22).

UNTIL stopping criterion is met.

The Lagrange function of Eq. (19) is

$$\mathcal{J}(w) = w^\top h + \gamma(1 - \sum_{i=1}^{r} \sqrt{w_i}). \tag{21}$$

By utilizing the Karush-Kuhn-Tucker (KKT) condition with $\frac{\partial \mathcal{J}(w)}{\partial w_i} = 0$ and the constraint $\sum_{i=1}^{r} \sqrt{w_i} = 1$, we obtain the solution of w as follows:

$$w_i = \left(h_i \sum_{j=1}^{r} \frac{1}{h_j} \right)^{-2}. \tag{22}$$

We can see that w is closely related to Z. Therefore, we could obtain both optimal similarity matrix Z and kernel weight w. We summarize the optimization process of Eq. (18) in Algorithm 2.

Experiments

Table 1: Description of the data sets

	# instances	# features	# classes
YALE	165	1024	15
JAFFE	213	676	10
ORL	400	1024	40
AR	840	768	120
COIL20	1440	1024	20
BA	1404	320	36
TR11	414	6429	9
TR41	878	7454	10
TR45	690	8261	10
TDT2	9394	36771	30

Data Sets

There are altogether ten real benchmark data sets used in our experiments. Table 1 summarizes the statistics of these data sets. Among them, the first six are image data, and the other four are text corpora[1,2].

[1]http://www-users.cs.umn.edu/ han/data/tmdata.tar.gz
[2]http://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html

The six image data sets consist of four famous face databases (ORL, YALE[3], AR[4] and JAFFE[5]), a toy image database COIL20[6], and a binary alpha digits data set BA[7]. Specifically, COIL20 contains images of 20 objects. For each object, the images were taken five degrees apart as the object is rotating on a turntable. There are 72 images for each object. Each image is represented by a 1,024-dimensional vector. BA consists of digits of "0" through "9" and letters of capital "A" through "Z". There are 39 examples for each class. YALE, ORL, AR, and JAFEE contain images of individuals. Each image has different facial expressions or configurations due to times, illumination conditions, and glasses/no glasses.

Kernel Design

To assess the effectiveness of multiple kernel learning, we adopted 12 kernels. They include: seven Gaussian kernels of the form $K(x, y) = exp(-\|x - y\|_2^2/(td_{max}^2))$, where d_{max} is the maximal distance between samples and t varies over the set $\{0.01, 0.0, 0.1, 1, 10, 50, 100\}$; a linear kernel $K(x, y) = x^\top y$; four polynomial kernels $K(x, y) = (a + x^\top y)^b$ with $a \in \{0, 1\}$ and $b \in \{2, 4\}$. Furthermore, all kernels are rescaled to $[0, 1]$ by dividing each element by the largest pair-wise squared distance.

Comparison Algorithms

For single kernel methods, we run downloaded kernel k-means (KKM) (Schölkopf, Smola, and Müller 1998), spectral clustering (SC) (Ng et al. 2002), robust kernel k-means (RKKM) (Du et al. 2015), and SCSK on each kernel separately. To demonstrate the advantage of our unified framework, we also implement three separate steps method (TSEP), i.e., learn the similarity matrix by (4), spectral clustering, k-means (repeat 20 times). And we report both the best and the average results over all these kernels.

In addition, we also implement the recent simplex sparse representation (SSR) (Huang, Nie, and Huang 2015) method and robust affinity graph construction methods by using random forest approach: ClustRF-u and ClustRF-a (Zhu, Change Loy, and Gong 2014). ClustRF-u assumes all tree nodes are uniformly important, while ClustRF-a assigns an adaptive weight to each node. Note that these three methods can only process data in the original feature space. Moreover, ClusteRF has a high demand for memory and cannot process high dimensional data directly. Thus we follow the authors' strategy and perform PCA on TR11, TR41, and TR45 to reduce the dimension. We use different numbers of dominant components and report the best clustering results. Nevertheless, we still cannot handle TDT2 data set with them.

For multiple kernel methods, we implement our proposed method and directly use the downloaded programs for the

[3]http://vision.ucsd.edu/content/yale-face-database
[4]http://www2.ece.ohio-state.edu/ aleix/ARdatabase.html
[5]http://www.kasrl.org/jaffe.html
[6]http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php
[7]http://www.cs.nyu.edu/ roweis/data.html

(a) Accuracy(%)

Data	KKM	KKM-m	SC	SC-m	RKKM	RKKM-m	ClustRF-u	ClustRF-a	SSR	TSEP	TSEP-m	SCSK	SCSK-m	MKKM	AASC	RMKKM	SCMK
YALE	47.12	38.97	49.42	40.52	48.09	39.71	57.58	57.58	54.55	62.58	44.60	**63.05**	52.88	45.70	40.64	52.18	**63.25**
JAFFE	74.39	67.09	74.88	54.03	75.61	67.98	97.65	98.59	87.32	98.30	73.88	**99.53**	90.06	74.55	30.35	87.07	**99.69**
ORL	53.53	45.93	57.96	46.65	54.96	46.88	60.75	62.75	69.00	70.15	41.45	**74.05**	53.56	47.51	27.20	55.60	**74.52**
AR	33.02	30.89	28.83	22.22	33.43	31.20	24.17	35.59	65.00	65.03	46.41	**78.90**	68.21	28.61	33.23	34.37	**79.29**
COIL20	59.49	50.74	67.70	43.65	61.64	51.89	74.44	72.99	76.32	77.68	61.03	**81.48**	62.59	54.82	34.87	66.65	**82.21**
BA	41.20	33.66	31.07	26.25	42.17	34.35	39.89	44.01	23.97	45.92	30.75	**46.02**	31.50	40.52	27.07	43.42	**45.57**
TR11	51.91	44.65	50.98	43.32	53.03	45.04	29.24	34.54	41.06	71.05	42.08	**74.22**	55.09	50.13	47.15	57.71	**74.26**
TR41	55.64	46.34	63.52	44.80	56.76	46.80	53.19	60.93	63.78	69.45	50.17	**70.17**	53.05	56.10	45.90	62.65	**70.25**
TR45	58.79	45.58	57.39	45.96	58.13	45.69	42.17	48.41	71.45	76.54	51.07	**77.74**	59.53	58.46	52.64	64.00	**77.47**
TDT2	47.05	35.58	52.63	45.26	48.35	36.67	-	-	20.86	54.78	46.35	**56.04**	45.02	34.36	19.82	37.57	**56.29**

(b) NMI(%)

Data	KKM	KKM-m	SC	SC-m	RKKM	RKKM-m	ClustRF-u	ClustRF-a	SSR	TSEP	TSEP-m	SCSK	SCSK-m	MKKM	AASC	RMKKM	SCMK
YALE	51.34	42.07	52.92	44.79	52.29	42.87	58.76	60.25	57.26	60.13	46.10	**60.58**	52.72	50.06	46.83	55.58	**61.04**
JAFFE	80.13	71.48	82.08	59.35	83.47	74.01	97.00	98.16	92.93	98.61	71.95	**99.18**	88.86	79.79	27.22	89.37	**99.20**
ORL	73.43	63.36	75.16	66.74	74.23	63.91	78.69	79.87	84.23	83.28	50.76	**84.78**	70.93	68.86	43.77	74.83	**85.21**
AR	65.21	60.64	58.37	56.05	65.44	60.81	57.09	66.64	84.16	84.69	64.63	**89.61**	80.34	59.17	65.06	65.49	**89.93**
COIL20	74.05	63.57	80.98	54.34	74.63	63.70	83.91	82.26	86.89	84.16	71.36	**87.03**	72.41	70.64	41.87	77.34	**86.72**
BA	57.25	46.49	50.76	40.09	57.82	46.91	54.66	58.17	30.29	59.47	32.45	**60.34**	42.91	56.88	42.34	58.47	**60.55**
TR11	48.88	33.22	43.11	31.39	49.69	33.48	18.97	24.77	27.60	62.71	29.88	**64.60**	44.48	44.56	39.39	56.08	**64.89**
TR41	59.88	40.37	61.33	36.60	60.77	40.86	52.63	56.78	59.56	64.07	39.58	**64.92**	47.97	57.75	43.05	63.47	**64.89**
TR45	57.87	38.69	48.03	33.22	57.86	38.96	38.12	43.70	67.82	70.03	40.17	**70.75**	50.47	56.17	41.94	62.73	**70.79**
TDT2	55.28	38.47	52.23	27.16	54.46	42.19	-	-	02.44	57.74	45.38	**59.25**	48.73	41.36	02.14	47.13	**58.66**

(c) Purity(%)

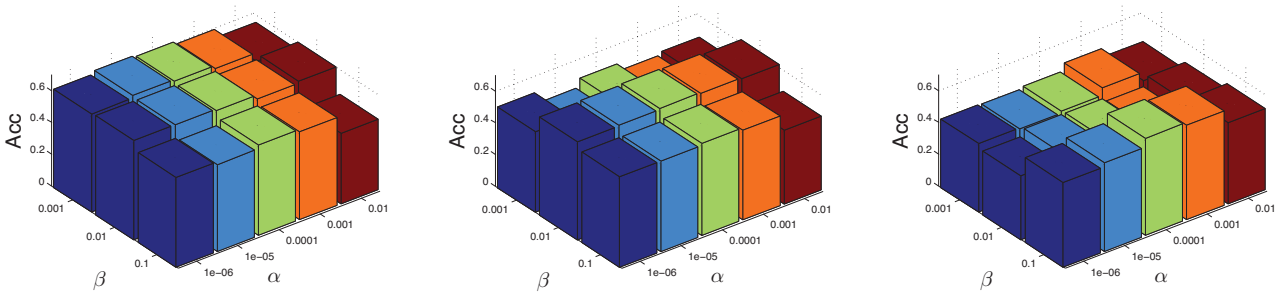
Data	KKM	KKM-m	SC	SC-m	RKKM	RKKM-m	ClustRF-u	ClustRF-a	SSR	TSEP	TSEP-m	SCSK	SCSK-m	MKKM	AASC	RMKKM	SCMK
YALE	49.15	41.12	51.61	43.06	49.79	41.74	63.64	63.03	58.18	64.77	55.38	**65.87**	56.19	47.52	42.33	53.64	**67.39**
JAFFE	77.32	70.13	76.83	56.56	79.58	71.82	97.65	98.59	96.24	99.06	77.08	**99.23**	91.24	76.83	33.08	88.90	**99.51**
ORL	58.03	50.42	61.45	51.20	59.60	51.46	67.25	66.00	76.50	76.00	52.39	**77.02**	57.96	52.85	31.56	60.23	**78.31**
AR	35.52	33.64	33.24	25.99	35.87	33.88	40.71	46.79	69.52	72.44	57.25	**83.08**	70.69	30.46	34.98	36.78	**83.20**
COIL20	64.61	55.30	69.92	46.83	66.35	56.34	80.83	77.71	**89.03**	84.03	74.89	84.24	75.58	58.95	39.14	69.95	**83.78**
BA	44.20	36.06	34.50	29.07	45.28	36.86	41.95	49.14	40.85	55.03	43.07	**55.49**	40.45	43.47	30.29	46.27	**55.72**
TR11	67.57	56.32	58.79	50.23	67.93	56.40	35.75	49.76	85.02	85.95	63.15	**86.25**	63.36	65.48	54.67	72.93	**85.84**
TR41	74.46	60.00	73.68	56.45	74.99	60.21	55.58	65.60	75.40	77.02	56.33	**78.53**	57.19	72.83	62.05	77.57	**78.49**
TR45	68.49	53.64	61.25	50.02	68.18	53.75	45.51	57.83	**83.62**	77.28	60.52	79.70	61.06	69.14	57.49	75.20	**79.78**
TDT2	52.79	49.26	50.39	42.81	62.13	52.60	-	-	46.79	67.75	56.07	**70.69**	64.53	54.89	21.73	60.02	**72.84**

Table 2: Clustering results obtained on benchmark data sets. '-m' denotes the average performance on the 12 kernels. Both the best results for single kernel and multiple kernel methods are highlighted in boldface.

methods in comparison on a combination of these 12 kernels:

MKKM[8]. The MKKM (Huang, Chuang, and Chen 2012b) extends k-means in a multiple kernel setting. However, it imposes a different constraint on the kernel weight distribution.

[8]http://imp.iis.sinica.edu.tw/IVCLab/research/Sean/mkfc/code



(a) $\gamma = 10^{-5}$ (b) $\gamma = 10^{-4}$ (c) $\gamma = 10^{-3}$

Figure 1: Parameter influence on accuracy for YALE data set.

AASC[9]. The AASC (Huang, Chuang, and Chen 2012a) is an extension of spectral clustering to the situation when multiple affinities exist. It is different from our approach since our method tries to learn an optimal similarity graph.

RMKKM[10]. The RMKKM (Du et al. 2015) extends k-means to deal with noise and outliers in a multiple kernel setting.

SCMK. Our proposed method of spectral clustering with multiple kernels. For the purpose of reproducibility, the code is publicly available[11].

For our method, we only need to run once. For those methods that involve K-means, we follow the strategy suggested in (Yang et al. 2010); i.e., we repeat clustering 20 times and present the results with the best objective values. We set the number of clusters to the true number of classes for all clustering algorithms.

Results

We present the clustering results of different methods on those benchmark data sets in Table 2. In terms of accuracy, NMI and Purity, our proposed methods obtain superior results. The big difference between the best and average results confirms that the choice of kernels has a huge influence on the performance of single kernel methods. This motivates our extended model for multiple kernel learning. Besides, our extended model for multiple kernel clustering usually improves the results over our model for single kernel clustering.

Although the best results of the three separate steps approach are sometimes close to our proposed unified method, their average values are often lower than our method. We notice that random forest based affinity graph method achieves good performance on image data sets. This observation can be explained by the fact that ClustRF is suitable to handle ambiguous and unreliable features caused by variation in illumination, face expression or pose on those data sets. On the other hand, it is not effective for text data sets. In most cases, ClustRF-a behaves better than ClustRF-u. This justifies the importance of considering neighbourhood-scale-adaptive weighting on the nodes.

[9]http://imp.iis.sinica.edu.tw/IVCLab/research/Sean/aasc/code

[10]https://github.com/csliangdu/RMKKM

[11]https://github.com/sckangz/AAAI18

Parameter Sensitivity

There are three parameters in our model: α, β, and γ. We use YALE data set as an example to demonstrate the sensitivity of our model SCMK to parameters. As shown in Figure 1, our model is quite insensitive to α and β, and γ over wide ranges of values. In terms of NMI and Purity, we have similar observations.

Conclusion

In this work, we address two problems existing in most classical spectral clustering algorithms, i.e., constructing similarity graph and relaxing discrete constraints to continuous one. To alleviate performance degradation, we propose a unified spectral clustering framework which automatically learns the similarity graph and discrete labels from the data. To cope with complex data, we develop our method in kernel space. A multiple kernel approach is proposed to solve kernel dependent issue. Extensive experiments on nine real data sets demonstrated the promising performance of our methods as compared to existing clustering approaches.

Acknowledgments

This paper was in part supported by Grants from the Natural Science Foundation of China (No. 61572111), the National High Technology Research and Development Program of China (863 Program) (No. 2015AA015408), a 985 Project of UESTC (No.A1098531023601041) and a Fundamental Research Fund for the Central Universities of China (No. A03017023701012).

References

Cai, X.; Nie, F.; Cai, W.; and Huang, H. 2013. Heterogeneous image features integration via multi-modal semi-supervised learning model. In *Proceedings of the IEEE International Conference on Computer Vision*, 1737–1744.

Cheng, B.; Yang, J.; Yan, S.; Fu, Y.; and Huang, T. S. 2010. Learning with-graph for image analysis. *IEEE transactions on image processing* 19(4):858–866.

Du, L.; Zhou, P.; Shi, L.; Wang, H.; Fan, M.; Wang, W.; and Shen, Y.-D. 2015. Robust multiple kernel k-means using 2; 1-norm. In *Proceedings of the 24th International Conference on Artificial Intelligence*, 3476–3482. AAAI Press.

Elhamifar, E., and Vidal, R. 2009. Sparse subspace clustering. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2790–2797. IEEE.

Fan, K. 1949. On a theorem of weyl concerning eigenvalues of linear transformations i. *Proceedings of the National Academy of Sciences of the United States of America* 35(11):652.

Huang, S.; Wang, H.; Li, T.; Li, T.; and Xu, Z. 2017. Robust graph regularized nonnegative matrix factorization for clustering. *Data Mining and Knowledge Discovery* 1–21.

Huang, H.-C.; Chuang, Y.-Y.; and Chen, C.-S. 2012a. Affinity aggregation for spectral clustering. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 773–780. IEEE.

Huang, H.-C.; Chuang, Y.-Y.; and Chen, C.-S. 2012b. Multiple kernel fuzzy clustering. *IEEE Transactions on Fuzzy Systems* 20(1):120–134.

Huang, J.; Nie, F.; and Huang, H. 2013. Spectral rotation versus k-means in spectral clustering. In *AAAI*.

Huang, J.; Nie, F.; and Huang, H. 2015. A new simplex sparse learning model to measure data similarity for clustering. In *Proceedings of the 24th International Conference on Artificial Intelligence*, 3569–3575. AAAI Press.

Kang, Z.; Peng, C.; and Cheng, Q. 2015. Robust subspace clustering via smoothed rank approximation. *IEEE Signal Processing Letters* 22(11):2088–2092.

Kang, Z.; Peng, C.; and Cheng, Q. 2017a. Kernel-driven similarity learning. *Neurocomputing*.

Kang, Z.; Peng, C.; and Cheng, Q. 2017b. Twin learning for similarity and clustering: A unified kernel approach. In *AAAI*, 2080–2086.

Kumar, A.; Rai, P.; and Daume, H. 2011. Co-regularized multi-view spectral clustering. In *Advances in neural information processing systems*, 1413–1421.

Mohar, B.; Alavi, Y.; Chartrand, G.; and Oellermann, O. 1991. The laplacian spectrum of graphs. *Graph theory, combinatorics, and applications* 2(871-898):12.

Ng, A. Y.; Jordan, M. I.; Weiss, Y.; et al. 2002. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems* 2:849–856.

Pavan, M., and Pelillo, M. 2007. Dominant sets and pairwise clustering. *IEEE transactions on pattern analysis and machine intelligence* 29(1):167–172.

Pei, Y.; Kim, T.-K.; and Zha, H. 2013. Unsupervised random forest manifold alignment for lipreading. In *Proceedings of the IEEE International Conference on Computer Vision*, 129–136.

Peng, C.; Kang, Z.; Yang, M.; and Cheng, Q. 2016. Feature selection embedded subspace clustering. *IEEE Signal Processing Letters* 23(7):1018–1022.

Premachandran, V., and Kakarala, R. 2013. Consensus of k-nns for robust neighborhood selection on graph-based manifolds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1594–1601.

Schölkopf, B.; Smola, A.; and Müller, K.-R. 1998. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation* 10(5):1299–1319.

Schönemann, P. H. 1966. A generalized solution of the orthogonal procrustes problem. *Psychometrika* 31(1):1–10.

Wang, J.; Chang, S.-F.; Zhou, X.; and Wong, S. T. 2008. Active microscopic cellular image annotation by superposable graph transduction with imbalanced labels. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1–8. IEEE.

Wen, Z., and Yin, W. 2013. A feasible method for optimization with orthogonality constraints. *Mathematical Programming* 142(1-2):397–434.

Yang, Y.; Xu, D.; Nie, F.; Yan, S.; and Zhuang, Y. 2010. Image clustering using local discriminant models and global integration. *IEEE Transactions on Image Processing* 19(10):2761–2773.

Yang, Y.; Ma, Z.; Yang, Y.; Nie, F.; and Shen, H. T. 2015. Multitask spectral clustering by exploring intertask correlation. *IEEE transactions on cybernetics* 45(5):1083–1094.

Yang, Y.; Shen, F.; Huang, Z.; and Shen, H. T. 2016. A unified framework for discrete spectral clustering. In *IJCAI*, 2273–2279.

Yu, S. X., and Shi, J. 2003. Multiclass spectral clustering. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 313–319. IEEE.

Zelnik-Manor, L., and Perona, P. 2004. Self-tuning spectral clustering. In *NIPS*, volume 17, 16.

Zeng, H., and Cheung, Y.-m. 2011. Feature selection and kernel learning for local learning-based clustering. *IEEE transactions on pattern analysis and machine intelligence* 33(8):1532–1547.

Zhang, C.; Nie, F.; and Xiang, S. 2010. A general kernelization framework for learning algorithms based on kernel pca. *Neurocomputing* 73(4):959–967.

Zhu, X.; Change Loy, C.; and Gong, S. 2014. Constructing robust affinity graphs for spectral clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1450–1457.