Membership Representation for Detecting Block-diagonal Structure in Low-rank or Sparse Subspace Clustering

Minsik Lee[†] Jieun Lee[‡] Hyeogjin Lee[§] Nojun Kwak[§]
Division of EE, Hanyang University, Korea[†]
Department of ECE, Ajou University, Korea[‡]
Graduate School of CST, Seoul National University, Korea[§]
mleepaper@hanyang.ac.kr mokona85@ajou.ac.kr {hjinlee, nojunk}@snu.ac.kr

Abstract

Recently, there have been many proposals with state-of-the-art results in subspace clustering that take advantages of the low-rank or sparse optimization techniques. These methods are based on self-expressive models, which have well-defined theoretical aspects. They produce matrices with (approximately) block-diagonal structure, which is then applied to spectral clustering. However, there is no definitive way to construct affinity matrices from these block-diagonal matrices and it is ambiguous how the performance will be affected by the construction method. In this paper, we propose an alternative approach to detect block-diagonal structures from these matrices. The proposed method shares the philosophy of the above subspace clustering methods, in that it is a self-expressive system based on a Hadamard product of a membership matrix. To resolve the difficulty in handling the membership matrix, we solve the convex relaxation of the problem and then transform the representation to a doubly stochastic matrix, which is closely related to spectral clustering. The result of our method has eigenvalues normalized in between zero and one, which is more reliable to estimate the number of clusters and to perform spectral clustering. The proposed method shows competitive results in our experiments, even though we simply count the number of eigenvalues larger than a certain threshold to find the number of clusters.

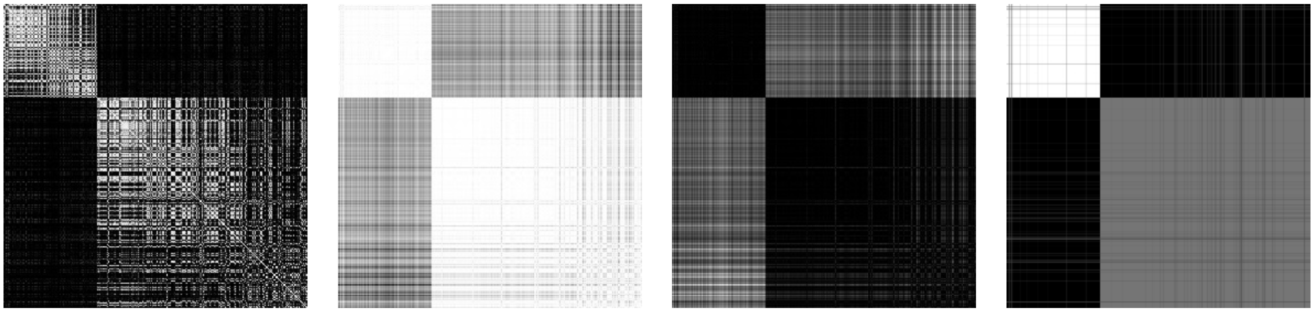
1. Introduction

Subspace clustering [6, 7, 11, 15, 25, 26] is a subfield of clustering research which has been established recently. Unlike K-means or kernel K-means clustering[1] which forms clusters based on minimizing the intra-cluster variance in the data space or in a high-dimensional kernel space, it aims to segment data into low-dimensional subspaces. Subspace clustering can be a useful tool in computer vision, because there are many types of data that can be well-represented by a low-dimensional structure such as face, motion, video segments, *etc*. Accordingly, there have been many successful applications of subspace clustering [4, 31].

Subspace clustering has been studied extensively, and there are tons of different approaches. According to [5], they can be roughly categorized into 7 different classes. Among them, the focus of this paper is the low-rank or sparse optimization approach, which is largely based on the recent advances in nonsmooth optimization [1, 2, 14] and shows state-of-the-art results. Most of the methods in this approach model the data by self-expressive dictionaries in conjunction with low-rank or sparse coefficients, *i.e*., they find a simplest way to represent the original data by using the data itself. Sparse subspace clustering (SSC) [5, 6] assumes the coefficients to be sparse, while low-rank representation (LRR) [15] assumes them to be low-rank. Low-rank subspace clustering (LRSC) [7] extends both methods and proposes an alternative non-convex formulation, which can be solved efficiently and gives a closed-form solution when there is no outlier. These methods have advantages in handling noise and outliers in data, and do not require to know the dimension of each cluster in advance. Theoretically speaking, even the number of clusters is not required if the input data is clean, *i.e*., there is neither noise nor outlier.

Unlike the well-defined theory, however, they need post-processing steps that can impact the performance significantly but are not clearly explained in theory. The outputs of these methods are matrices that have (approximately) block-diagonal structure, which we call as *latent matrices* in this paper. These latent matrices are used for spectral clustering to find the actual clusters, and to do that, they need to be transformed to affinity matrices. For example, LRR uses a heuristic post-processing step to make the ma-

[1] Since spectral clustering was shown to be equivalent to kernel K-means clustering [27, 29], spectral clustering can also be classified as this type of clustering problem.



(a) Input (latent matrix $\mathbf{W}$ = output of SSC/LRR)

(b) Output of the 1st stage: Approximation of membership matrix ($\breve{\mathbf{M}}$)

(c) Dissimilarity matrix ($\mathbf{H} = \mathbf{1}\mathbf{1}^T - \breve{\mathbf{M}}$)

(d) Output of the 2nd stage: Approximation of normalized membership matrix ($\breve{\mathbf{F}}$)

Figure 1. A typical example of the intermediate results of the proposed algorithm (Hopkins155, $K = 2$, $n = 276$).

trix symmetric and nonnegative [15]. Later, LRR-PSD [17] has been proposed to resolve this issue by enforcing the latent matrix to be positive semidefinite (PSD). However, the performance is not very different from that of LRR. On the other hand, the solution of SSC has zero-diagonal entries to avoid the trivial solution (so it is obviously not PSD) and they construct the affinity matrix by absoluting the elements and dividing them by the maximum element in each row. Intuitively, it makes sense that the elements with large magnitudes in the latent matrix will represent stronger affinity. However, it is ambiguous how the way of constructing affinity matrix will affect the performance. Moreover, since all of these methods use spectral clustering, they need to know the number of clusters in advance or to estimate it heuristically.

In this paper, we propose the membership representation (MR), which detects the block-diagonal structures from these latent matrices, as an alternative. Our motivation is that this detection problem can be formulated as another self-expressive system with a semidefinite constraint. Our formulation represents a latent matrix by a Hadamard product of the latent matrix itself and a membership matrix, which frequently appears in correlation clustering [3, 16, 20]. Handling a membership matrix, however, is a difficult problem, because it is a discrete matrix with a special structure. We resolve this issue by dividing the problem into two-fold: First, we solve a convex relaxation of this problem to find an approximate solution. Second, we propose another problem to transform the representation to a normalized membership matrix, which is closely related to spectral clustering, and also solve the convex relaxation of the second problem. Figure 1 shows the intermediate results of the proposed algorithm, which can give a general sense of the method. The final output matrix of our method has eigenvalues in the range of $[0, 1]$ regardless of the input data, which is more reliable to estimate the number of clusters and perform spectral clustering. In this paper, we simply set the number of clusters K as the number of eigenvalues of this matrix above a certain threshold, which gave better or at least similar results than the plain spectral clustering approach.

This work has been largely inspired by the recent studies in correlation clustering [3, 9, 16, 20], and doubly stochastic normalization [22, 27, 28] in spectral clustering. The remainder of this paper is organized as follows: We present a brief review of SSC and LRR, and present the main idea of MR in Section 2. The details of the proposed method is introduced in Section 3, and the experimental results follow in Section 4. Finally, we conclude the paper in Section 5.

2. Detecting block-diagonal structure

2.1. A brief review of SSC and LRR

SSC and LRR try to represent the original data by using the data itself as a dictionary. Let $\mathbf{X} \in \mathbb{R}^{d \times n}$ be the data matrix to be clustered, whose column vectors are sample features. For ease of explanation, we will assume $\mathbf{X} = [\mathbf{X}_1 \ \ldots \ \mathbf{X}_K]$ where $\mathbf{X}_k$ $(1 \leq k \leq K)$ is a matrix that contains the samples belonging to the kth cluster, which resides on a low-dimensional subspace. Note that, however, all the formulas in this paper hold after arbitrary column and/or row permutations. Roughly speaking, SSC and LRR both solve the following optimization problem:

$$\begin{aligned} \min_{\mathbf{W},\mathbf{E}} \quad & \|\mathbf{W}\|_{o_W} + \lambda_E \|\mathbf{E}\|_{o_E}, \\ \text{s.t.} \quad & \mathbf{X} = \mathbf{X}\mathbf{W} + \mathbf{E}, \end{aligned} \tag{1}$$

where $\|\cdot\|_{o_W}$ and $\|\cdot\|_{o_E}$ are some norms. Here, $\mathbf{W} \in \mathbb{R}^{n \times n}$ represents the latent matrix and $\mathbf{E} \in \mathbb{R}^{d \times n}$ contains noise and outliers. If $\|\cdot\|_{o_W}$ is a low-rank or sparse norm, the solution of $\mathbf{W}$ becomes approximately block-diagonal. In LRR, the nuclear norm $\|\mathbf{W}\|_*$ (sum of singular values) is used to minimize the rank of $\mathbf{W}$, and the reason is stated in the following theorem [15]:

Theorem 1. *The minimizer of the problem*

$$\min_{\mathbf{W}} \|\mathbf{W}\|_*, \quad \text{s.t.} \quad \mathbf{X} = \mathbf{X}\mathbf{W}, \tag{2}$$

has a block-diagonal form such that the row and column indices for each block correspond to the indices of samples that belong to each cluster, i.e., $\mathbf{X}_k$*, in* $\mathbf{X}$*.*

Please refer to [15] for the proof. Note that we have slightly changed the description of the original theorem for explanation, but it essentially states the same fact. Thus, if there is no noise or outlier, $\mathbf{W}$ becomes block-diagonal. If there is some noise or outliers in $\mathbf{X}$, then $\mathbf{E}$ in (1) will contain them as far as possible, and the resulting $\mathbf{W}$ will still be approximately block-diagonal. In [15], l_1-norm and the group sparsity norm ($l_{2,1}$-norm) [24] was used for $\|\cdot\|_{o_E}$.

In SSC, l_1-norm is used for $\|\cdot\|_{o_W}$, and the diagonal entries of $\mathbf{W}$ are constrained to be zeros to avoid trivial solution $\mathbf{W} = \mathbf{I}$. Hence, SSC seeks to represent a sample by a linear combination of a few other samples. [6] gives a similar theorem to Theorem 1 for this formulation, which ensures the block-diagonal structure of $\mathbf{W}$ for a clean $\mathbf{X}$. For $\mathbf{E}$, it is divided into two terms and each of them are subjected to l_1- or l_2-norm, in order to model both noise and outliers.

After solving (1) to obtain $\mathbf{W}$, both methods construct an affinity matrix $\mathbf{W}'$ based on it. In LRR, $\mathbf{W}'$ is set as $\mathbf{W}'_{ij} = ([\mathbf{U}'\mathbf{U}'^T]_{ij})^2$ where $\mathbf{U}'$ is formed of the normalized rows of $\mathbf{U}\mathbf{S}^{\frac{1}{2}}$ from the singular value decomposition (SVD) of $\mathbf{W} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. In SSC, small elements are discarded from $\mathbf{W}$ and the absolute values of each row is divided by the largest element in the row to build $\mathbf{W}'$. After constructing $\mathbf{W}'$, spectral clustering is applied to find the final clusters. To estimate the number of clusters K, [15] proposes to use the following heuristic estimator:

$$\hat{K} = n - \text{round}\Big(\sum f_\tau(\sigma_i)\Big), \tag{3}$$

where σ_i is the singular value of the normalized Laplacian matrix $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}'\mathbf{D}^{-\frac{1}{2}}$ of $\mathbf{W}'$ where $\mathbf{D}_{ii} = \sum_j \mathbf{W}'_{ij}$ and

$$f_\tau(\sigma) = \begin{cases} 1, & \text{if } \sigma > \tau, \\ \log_2(1 + \frac{\sigma^2}{\tau^2}), & \text{otherwise.} \end{cases} \tag{4}$$

2.2. Membership representation

Now, we will introduce our MR formulation to detect block-diagonal structure. First, let us define membership matrix:

Definition 1. A membership matrix $\mathbf{M}$ is a symmetric matrix whose elements are either one or zero, which can be transformed into a block-diagonal matrix by permuting the same indices of rows and columns.

This type of matrix frequently appears in correlation clustering [3, 16, 20]. For ease of explanation, we will assume that the rows and columns of $\mathbf{M}$ are aligned as

$$\mathbf{M} = \begin{bmatrix} \mathbf{1}\mathbf{1}^T & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{1}\mathbf{1}^T \end{bmatrix}, \tag{5}$$

where the blocks of ones[2] corresponds to the "cluster blocks" in $\mathbf{W}$. Note that $\mathbf{M}$ is PSD because every block is PSD. In fact, a discrete matrix that is PSD can conversely define a membership matrix, by the following theorem:

Theorem 2. $\mathbf{M}'$ *is a membership matrix iff it is a matrix of ones and zeros with diagonal elements being ones and is PSD.*

Proof. Only-if part is based on the definition and the property of the membership matrix. To prove the if part, let us define a new matrix $\mathbf{M}^*$ by permuting the same indices of rows and columns of $\mathbf{M}'$, so that $\mathbf{M}^*$ is block-diagonal with irreducible blocks (or itself is a irreducible matrix). Then, every irreducible block must be PSD. What we need to show is that these blocks are matrices of ones. Let $\mathbf{M}''$ be one of the blocks and suppose that there are i_1, i_2, and i_3 such that $\mathbf{M}''_{i_1 i_2} = \mathbf{M}''_{i_2 i_3} = 1$. Then, the submatrix of $\mathbf{M}''$ with these indices will look like

$$\begin{bmatrix} 1 & 1 & \mathbf{M}''_{i_1 i_3} \\ 1 & 1 & 1 \\ \mathbf{M}''_{i_3 i_1} & 1 & 1 \end{bmatrix}. \tag{6}$$

In order for this matrix to be PSD, the unknown values in above expression should be $\mathbf{M}''_{i_1 i_3} = \mathbf{M}''_{i_3 i_1} = 1$.

Now, let $\{i_j\}$ be the indices from which an all-one submatrix of $\mathbf{M}''$ is formed. If $\{i_j\}$ does not contain all the indices of $\mathbf{M}''$, then we can always find another entry i' such that some of $\{i_j\}$ are connected to, *i.e.*, $\mathbf{M}''_{i' i_j} = 1$ for some j, because $\mathbf{M}''$ is irreducible. Then, it is obvious that the submatrix of $\mathbf{M}''$ with indices $\{i'\} \cup \{i_j\}$ must be filled with ones because $\mathbf{M}''_{i' i_j} = \mathbf{M}''_{i_j i_{j'}} = 1$ for all $i_{j'}$ in $\{i_j\}$. This recursively proves that $\mathbf{M}''$ is a matrix of ones. □

This theorem shows that the PSD property is a vital condition for a membership matrix. Hereafter, we will denote $\mathbb{M}$ as the set of all membership matrices.

Now, let us represent $\mathbf{W}$ self-expressively using the membership matrix $\mathbf{M}$. If $\mathbf{W}$ is a clean block-diagonal matrix without any error and $\mathbf{M}$ is a membership matrix whose all-one block locations are identical to those of diagonal blocks of $\mathbf{W}$, then obviously

$$\mathbf{W} = \mathbf{W} \odot \mathbf{M}, \tag{7}$$

[2] In this paper, we use $\mathbf{1}$ to denote vectors of ones, and $\mathbf{0}$ to denote both vectors and matrices of zeros.

where $\odot$ is the Hadamard product (element-wise product). However, note that the trivial solution $\mathbf{M} = \mathbf{1}\mathbf{1}^T$ also satisfies this equation, hence we have to minimize the number of ones in $\mathbf{M}$ to find a valid answer. The following theorem shows that minimizing the number of ones in $\mathbf{M}$ gives the desired solution:

Theorem 3. *Let $\mathbf{W}$ be a block-diagonal matrix whose blocks are irreducible. Then, the minimizer of the problem*

$$\min_{\mathbf{M}} N(\mathbf{M}), \quad \text{s.t.} \quad \mathbf{W} = \mathbf{W} \odot \mathbf{M}, \quad \mathbf{M} \in \mathbb{M}, \tag{8}$$

where $N(\cdot)$ is the number of non-zero entries, is the membership matrix that indicates the block-diagonal structure of $\mathbf{W}$. Moreover, this holds even if we replace $N(\cdot)$ to any entry-wise norm.

Proof. To satisfy $\mathbf{W} = \mathbf{W} \odot \mathbf{M}$, all the elements of $\mathbf{M}$ that correspond to the non-zero entries in $\mathbf{W}$ should be ones. Then, because a diagonal block of $\mathbf{W}$ is irreducible, the corresponding block of $\mathbf{M}$ should be filled with ones, based on the proof of Theorem 2. To minimize $N(\mathbf{M})$, all the other elements should be zeros. Minimizing an entry-wise norm by fixing some elements will make the other elements zeros, hence this theorem also holds for any entry-wise norm. □

Note that one might think of using the nuclear norm for $\mathbf{M}$, based on the fact that the rank of $\mathbf{M}$ is K. However, it is meaningless since $\|\mathbf{M}\|_* = n$ for any $\mathbf{M} \in \mathbb{M}$.

In reality, there can be errors in $\mathbf{W}$, hence we consider the following alternative problem:

$$\begin{aligned} \min_{\mathbf{M}} \quad & \|\mathbf{W} - \mathbf{W} \odot \mathbf{M}\|_{o'_W} + \lambda_M \|\mathbf{M}\|_{o_M}, \\ \text{s.t.} \quad & \mathbf{M} \in \mathbb{M}, \end{aligned} \tag{9}$$

where $\|\cdot\|_{o'_W}$ and $\|\cdot\|_{o_M}$ are entry-wise norms, of which choices will be discussed later. However, this problem is NP-hard because it has both discrete and PSD constraints. Instead, we can relax the constraints as

$$\text{diag}(\mathbf{M}) = \mathbf{1}, \quad \mathbf{M} \succeq \mathbf{0}, \quad \mathbf{M} \geq 0, \tag{10}$$

where $\succeq$ and $\geq$ are semidefinite and element-wise inequality, respectively. Note that the first two conditions also imply $\mathbf{M} \leq \mathbf{1}\mathbf{1}^T$. A similar relaxation technique has been used in correlation clustering [16, 20]. Based on this relaxation, (9) becomes a convex problem which can be efficiently solved either by a semidefinite programming (SDP) or an augmented Lagrangian method (ALM) [14]. The solution $\breve{\mathbf{M}}$ of this relaxed problem is not a membership matrix, but the elements will be close to one (zero) if it is within (outside of) a cluster block. This can be interpreted as the similarities between the samples, hence, we will call $\breve{\mathbf{M}}$ as the similarity matrix[3].

Note that $\breve{\mathbf{M}}$ is a boosted version of $\mathbf{W}$ that emphasizes the block-diagonal structure, but we still do not know the actual clusters. There can be several ways to cluster this result. First, we may use Single Linkage (SLINK) [19] for post-processing as in some work [9] in correlation or spectral clustering. The merit of this approach is that the number of clusters is not required. However, SLINK is a greedy method that is based on local information, thus it often gives a poor result. Second, since $\breve{\mathbf{M}}$ is PSD and the scale of its elements are evenly distributed, it may be regarded as an affinity matrix for the normalized cut [18]. The disadvantage is that it requires the number of clusters, but it is better in that it considers global information. In this work, we take the third approach which is closely related to the second one.

2.3. Normalized membership representation

In this section, we will explain how to transform the similarity matrix $\breve{\mathbf{M}}$ to an affinity matrix that is more adequate for spectral clustering. For this, we introduce the normalized membership matrix, which can be considered as a variant of the membership matrix. A normalized membership matrix $\mathbf{F}$ is similar to a membership matrix, except that the cluster blocks are filled with $\frac{1}{n_k}$ instead of ones, where n_k is the number of samples in the kth cluster. Formally speaking, by permuting the same indices of rows and columns, this symmetric matrix can be made to be a block-diagonal matrix whose diagonal blocks are filled with the inverse of the dimension of each block. For ease of explanation, we will also assume that the rows and columns of $\mathbf{F}$ are also aligned as in (5).

If we denote $\mathbb{F}$ as the set of all normalized membership matrices, then every membership matrix has its counterpart in $\mathbb{F}$. $\mathbf{F}$ shares some properties with $\mathbf{M}$, such as it being PSD and its elements being nonnegative. The differences between them are that (i) $\mathbf{F}$ is doubly stochastic [27], *i.e.*, $\mathbf{F1} = \mathbf{1}$ and $\mathbf{F}^T\mathbf{1} = \mathbf{1}$, and (ii) all the eigenvalues of $\mathbf{F}$ are either zero or one. Therefore, the nuclear norm gives the same result as the rank of $\mathbf{F}$, *i.e.*, rank$(\mathbf{F}) = \|\mathbf{F}\|_* = \text{tr}(\mathbf{F})$. In fact, these properties lead to another fundamental property of the normalized membership matrix: A normalized membership matrix $\mathbf{F}$ can be completely described by the following conditions[4]:

$$\mathbf{F} \geq 0, \quad \mathbf{F1} = \mathbf{1}, \quad \mathbf{F}^2 = \mathbf{F} = \mathbf{F}^T. \tag{11}$$

[3] In this paper, we distinguish the terms latent matrix, affinity matrix and similarity matrix. A latent matrix is an output of a subspace clustering method, while an affinity matrix is an input of the spectral clustering. Similarity matrix is a matrix that represents the similarity between samples, where the range of similarity is $[0, 1]$

[4] Analogous conditions were given in [27], however, they are based on the nonnegative factorization of $\mathbf{F}$. Our conditions do not need any factorization, thanks to Lemma 1 in the supplementary material.

In other words, a matrix is a normalized membership matrix if and only if it is doubly stochastic and is an orthogonal projection. Due to space limitation, the proof is shown in the supplementary material. Note that this description does not need an explicit discreteness constraint, which is a new finding as far as we know.

The importance of the normalized membership matrix is that $\mathbb{F}$ is the feasible set for kernel K-means clustering [27], which is closely related to spectral clustering. Kernel K-means clustering can be reduced to the following problem [27, 29]:

$$\max_{\mathbf{F}} \quad \mathrm{tr}(\mathbf{\Phi F}), \quad \text{s.t.} \quad \mathbf{F} \in \mathbb{F}, \tag{12}$$

where $\mathbf{\Phi}$ is a kernel matrix. Of course, this is a difficult problem to solve in exact sense, and spectral clustering can be considered as an approximate procedure in solving this problem. In recent studies [28] of spectral clustering, normalizing $\mathbf{\Phi}$ has been suggested so that it is close to a member of $\mathbb{F}$. Actually, it is shown that the normalization step in the normalized cut makes $\mathbf{\Phi}$ closer to a doubly stochastic matrix, which is a convex relaxation of $\mathbf{F}$ [27]. This can be an explanation of why the normalized cut has better chance of providing good results. Accordingly, there have been many different proposals on normalizing $\mathbf{\Phi}$ to a doubly stochastic matrix [22, 27, 28].

awsdfgasdfgdfswrtghn

Thus, it might be beneficial if we can find an efficient way to directly transform $\breve{\mathbf{M}}$ into a normalized membership matrix. An intuitive way is to utilize the equation (7) conversely: In an ideal case where $\breve{\mathbf{M}} \in \mathbb{M}$, it is obvious that

$$\mathbf{F} = \mathbf{F} \odot \breve{\mathbf{M}}, \tag{13}$$

where $\mathbf{F}$ is the normalized membership counterpart of $\breve{\mathbf{M}}$. From another perspective, this can also be expressed as

$$\mathbf{H} \odot \mathbf{F} \triangleq (\mathbf{1}\mathbf{1}^T - \breve{\mathbf{M}}) \odot \mathbf{F} = \mathbf{0}, \tag{14}$$

where $\mathbf{H}$ is a dissimilarity matrix, which has the opposite meaning of $\breve{\mathbf{M}}$. In other words, $\mathbf{F}$ fills the empty elements of $\mathbf{H}$. In real situations, $\breve{\mathbf{M}} \notin \mathbb{M}$, so we have to minimize $\|\mathbf{H} \odot \mathbf{F}\|_{o_F}$ for some entry-wise norm $\|\cdot\|_{o_F}$. However, note that there is a trivial solution $\mathbf{F} = \mathbf{I}$ to this problem, hence we need a regularization term $\|\mathbf{F}\|_* = \mathrm{tr}(\mathbf{F})$. The resulting cost function will be something like $\|\mathbf{H} \odot \mathbf{F}\|_{o_F} + \lambda_F \|\mathbf{F}\|_*$.

Yet, there is another problem. Note that $\|\mathbf{H} \odot \mathbf{F}\|_{o_F}$ is roughly proportional to the errors in $\mathbf{H}$, but $\mathrm{tr}(\mathbf{F})$ is directly related to the number of clusters in $\mathbf{F}$. Which means, the appropriate value of λ_F will vary largely for different problems. Hence, we instead minimize $\mathrm{tr}(\mathbf{F})$ with a constraint $\|\mathbf{H} \odot \mathbf{F}\|_{o_F} \leq c$, where the constant c is decided based on $\mathbf{H}$. Since handling a normalized membership matrix is also difficult, we may relax the constraint to make it tractable as

$$\begin{aligned} \min_{\mathbf{F}} \quad & \mathrm{tr}(\mathbf{F}), \\ \text{s.t.} \quad & \mathbf{F} \geq 0, \ \mathbf{F}\mathbf{1} = \mathbf{1}, \ \mathbf{F} \succeq \mathbf{0}, \ \|\mathbf{H} \odot \mathbf{F}\|_{o_F} \leq c. \end{aligned} \tag{15}$$

Note that the constraint $\mathbf{F}^2 = \mathbf{F}$ in (11) is relaxed to a PSD constraint on $\mathbf{F}$.

The solution $\breve{\mathbf{F}}$ of the above problem can be considered as a doubly stochastic normalization [27] of $\breve{\mathbf{M}}$, which is more appropriate for spectral clustering. In performing spectral clustering on $\breve{\mathbf{F}}$, we do not need the normalization step of the normalized cut, because our affinity matrix is already doubly stochastic. Since all the eigenvalues of $\breve{\mathbf{F}}$ are in between zero and one regardless of the input data, it can be more reliable to infer the number of clusters from $\breve{\mathbf{F}}$ than from $\mathbf{W}'$. In our experiments, we just counted the eigenvalues above $\frac{1}{2}$ to estimate K. We observe that, in many cases, $\breve{\mathbf{F}}$ gives a solution that is already very close to $\mathbb{F}$, as shown in Fig. 1(d).

3. Design of the algorithm

In this section, we explain the detailed procedure of MR. For all the subproblems of MR, we used ALM [14] to optimize the cost functions.

3.1. Finding $\breve{\mathbf{M}}$

First of all, we will discuss the choices of $\|\cdot\|_{o'_W}$ and $\|\cdot\|_{o_M}$ in (9). For $\|\cdot\|_{o'_W}$, we use the entry-wise l_1-norm to handle outliers. For $\|\cdot\|_{o_M}$, it is natural to use l_1-norm because it is a convex surrogate of $N(\cdot)$, the number of nonzero elements. An interesting fact is that $\|\mathbf{M}\|_1$ is the same as $\|\mathbf{M}\|_F^2$ for $\mathbf{M} \in \mathbb{M}$, *i.e.*, $\|\mathbf{M}\|_1 = \|\mathbf{M}\|_F^2$, for a membership matrix but they are not for its convex relaxation, so using either one or even an elastic-net norm [32] can be considered as a valid relaxation. In this work, we use the Frobenius norm because of the following reason: $\breve{\mathbf{F}}$ is found by filling the empty elements of $\mathbf{H}$, thus, rather than having many zeros in $\breve{\mathbf{M}}$, making the elements of the non-cluster blocks in $\mathbf{H}$ evenly large is more important. Another thing to mention is that, in fact, we use $\mathbf{W}'$, the post-processing result of LRR or SSC, instead of $\mathbf{W}$ for the input of our algorithm, because the post-processing steps regulate the scale of elements. Since our algorithm is based on minimizing the element-wise errors, the performance will not be good if the scale of the elements varies largely.

Now, we reformulate (9) with an auxiliary variable as

$$\begin{aligned} \min_{\mathbf{M}_1, \mathbf{M}_2} \quad & \|\mathbf{W} - \mathbf{W} \odot \mathbf{M}_1\|_1 + \lambda_M \|\mathbf{M}_1\|_F^2, \\ \text{s.t.} \quad & \mathbf{M}_1 \geq 0, \quad \mathbf{M}_2 \succeq \mathbf{0}, \quad \mathbf{M}_1 = \mathbf{M}_2, \\ & \mathrm{diag}(\mathbf{M}_1) = \mathbf{1}. \end{aligned} \tag{16}$$

Since $\mathbf{M}_1 \leq \mathbf{1}\mathbf{1}^T$, the above cost function is equivalent to $\lambda_M \|\mathbf{M}_1 - \mathbf{B}\|_F^2$, where $\mathbf{B} = \frac{1}{2\lambda_M}\mathrm{sign}(\mathbf{W}) \odot \mathbf{W}$. The corresponding ALM problem is

$$\begin{aligned}\min_{\mathbf{M}_1,\mathbf{M}_2}\quad & \frac{1}{2}\|\mathbf{M}_1-\mathbf{B}\|_F^2\\ & +\langle\mathbf{\Pi},\mathbf{M}_1-\mathbf{M}_2\rangle+\frac{\mu}{2}\|\mathbf{M}_1-\mathbf{M}_2\|_F^2,\\ \text{s.t.}\quad & \mathbf{M}_1\geq 0,\quad \operatorname{diag}(\mathbf{M}_1)=\mathbf{1},\quad \mathbf{M}_2\succeq\mathbf{0},\end{aligned}\tag{17}$$

where $\mathbf{\Pi}$ is the Lagrange multiplier. Note that the minimizer for $\mathbf{M}_1$ ($\mathbf{M}_2$) fixing $\mathbf{M}_2$ ($\mathbf{M}_1$) can be found as a closed-form. Since this is a convex problem with two variables, it can be efficiently solved using the alternating direction method of multipliers (ADMM) [14]. Accordingly, the update formulas are given as

$$\begin{aligned}\mathbf{M}_1 &\leftarrow \max(\mu\mathbf{M}_2+\mathbf{B}-\mathbf{\Pi},\mathbf{0})/(1+\mu),\\ \operatorname{diag}(\mathbf{M}_1) &\leftarrow \mathbf{1},\\ \mathbf{M}_2 &\leftarrow P_{\succeq\mathbf{0}}(\mathbf{M}_1+\mu^{-1}\mathbf{\Pi}),\\ \mathbf{\Pi} &\leftarrow \mathbf{\Pi}+\mu(\mathbf{M}_1-\mathbf{M}_2),\\ \mu &\leftarrow \rho\mu,\end{aligned}\tag{18}$$

for a constant $\rho > 1$. Here, the $\max$ operator is an element-wise operator. $P_{\succeq\mathbf{0}}(\cdot)$ is a projection operator to the nearest PSD matrix, *i.e.*, $P_{\succeq\mathbf{0}}(\mathbf{Y}) = \mathbf{V}\max(\mathbf{S},\mathbf{0})\mathbf{V}^T$ where $\frac{1}{2}(\mathbf{Y}+\mathbf{Y}^T) = \mathbf{VSV}^T$ is an eigenvalue decomposition. We find $\breve{\mathbf{M}}$ by iterating (18) until convergence.

3.2. Finding $\breve{\mathbf{F}}$

Solving the optimization problem (15) is similar to the previous section. We use l_1-norm for $\|\cdot\|_{o_F}$, in order to make many zeros in $\breve{\mathbf{F}}$. If we assume that the ratio of error in $\mathbf{H}$ is consistent, then the amount of errors in $\|\mathbf{H}\odot\mathbf{F}\|_1$ will be roughly proportional to $\|\mathbf{H}\|_1/n$, where n is divided because the sum of all the elements of $\mathbf{F}$ is n, not n^2. Hence, we set $c = \beta\|\mathbf{H}\|_1/n$ for some constant $\beta > 0$. Since the elements of $\breve{\mathbf{M}}$ are in between zero and one, the same is also true for those of $\mathbf{H}$. Moreover, the last constraint of (15) is equivalent to $\langle\mathbf{H},\mathbf{F}\rangle \leq c$, since $\mathbf{F}$ is nonnegative.

The corresponding ALM problem is

$$\begin{aligned}\min_{\mathbf{F}_1,\mathbf{F}_2}\quad & \operatorname{tr}(\mathbf{F}_1)+\langle\mathbf{\Pi},\mathbf{F}_1-\mathbf{F}_2\rangle+\frac{\mu}{2}\|\mathbf{F}_1-\mathbf{F}_2\|_F^2,\\ \text{s.t.}\quad & \mathbf{F}_1\mathbf{1}=\mathbf{1},\ \mathbf{F}_1\succeq\mathbf{0},\ \mathbf{F}_2\geq 0,\ \langle\mathbf{H},\mathbf{F}_2\rangle\leq c.\end{aligned}\tag{19}$$

Note that we have reused the notations of $\mathbf{\Pi}$ and μ. The corresponding update formulas are

$$\begin{aligned}\mathbf{F}_1 &\leftarrow P_{\succeq\mathbf{0}}(P_{\mathbf{1}}(\mathbf{F}_2-\mu^{-1}(\mathbf{I}+\mathbf{\Pi}))),\\ \mathbf{F}_2 &\leftarrow P_{\{\geq\mathbf{0},\mathbf{H}\}}(\mathbf{F}_1+\mu^{-1}\mathbf{\Pi}),\\ \mathbf{\Pi} &\leftarrow \mathbf{\Pi}+\mu(\mathbf{F}_1-\mathbf{F}_2),\\ \mu &\leftarrow \rho\mu,\end{aligned}\tag{20}$$

where $P_{\mathbf{1}}(\mathbf{Y}) = (\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^T)\mathbf{Y}(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^T)+\frac{1}{n}\mathbf{1}\mathbf{1}^T$ [27]. $P_{\{\geq\mathbf{0},\mathbf{H}\}}(\mathbf{Z})$ is a projection operator to the nonnegative matrix that satisfies $\langle\mathbf{H},\mathbf{Z}\rangle \leq c$: If $\max(\mathbf{Z},\mathbf{0})$ already satisfies the condition, then this is the projected matrix. Otherwise, $\max(\mathbf{Z}+\alpha\mathbf{H},\mathbf{0})$ is the projected matrix, for α that makes the sum of elements equals to c. This α can be efficiently found by sorting the elements of $\mathbf{G} \triangleq \mathbf{X}\oslash\mathbf{H}$, where $\oslash$ is the Hadamard division (element-wise division). For the intervals between the consecutive (sorted) elements of $\mathbf{G}$, the variation of α will not change the "clipped" elements of $\mathbf{Z}+\alpha\mathbf{H}$ due to $\max(\cdot,\mathbf{0})$. Hence, by recursively updating α through the sorted elements of $\mathbf{G}$, we can find the solution in linear time.

Note that the first update equation of (20) simply calculates $P_{\succeq\mathbf{0}}(P_{\mathbf{1}}(\cdot))$ to find a matrix that is both PSD and doubly stochastic. It is usually not true that a projection to the intersection of two convex sets is the same as consecutive projections to each set. However, this case is an exception. A symmetric matrix that is doubly stochastic must have an eigenvector $\mathbf{1}$ with eigenvalue 1. Let us express $\mathbf{F}_1$ as a sum of two symmetric matrices, *i.e.*, $\mathbf{F}_1 \triangleq \mathbf{A}+\mathbf{B}$, so that $\mathbf{B1} = \mathbf{B}^T\mathbf{1} = \mathbf{0}$. Then, $\mathbf{A}$ must be $\mathbf{A} = \frac{1}{n}\mathbf{1}\mathbf{1}^T$ and $\mathbf{B}$ must be a PSD matrix that is closest to the component of $\mathbf{F}_2-\mu^{-1}(\mathbf{I}+\mathbf{\Pi})$ that is orthogonal to $\frac{1}{n}\mathbf{1}\mathbf{1}^T$, because projecting w.r.t. the Euclidean distance is identical to projecting each orthogonal component individually. Since $P_{\mathbf{1}}$ eliminates any components related to $\frac{1}{n}\mathbf{1}\mathbf{1}^T$ and replace them with $\frac{1}{n}\mathbf{1}\mathbf{1}^T$, $P_{\succeq\mathbf{0}}(P_{\mathbf{1}}(\cdot))$ will yield the closest doubly stochastic PSD matrix.

The computational complexity of the two subproblems in MR is dominated by the corresponding eigenvalue decomposition for each iteration ($P_{\succeq\mathbf{0}}(\cdot)$ operation in (18) and (20)). This surely requires more processing time than a plain spectral clustering approach, but it is comparable to that of LRR, of which the computational complexity is dominated by the singular value decomposition for each iteration.

3.3. Post-processing

After finding the solution $\breve{\mathbf{F}}$ of (19) and estimating K based on it, we can perform spectral clustering. By performing eigenvalue decomposition $\breve{\mathbf{F}} = \mathbf{VSV}^T$ and selecting the K largest eigenvalues $\mathbf{S}'$ and the corresponding eigenvectors $\mathbf{V}'$, we can calculate $\acute{\mathbf{V}} \triangleq \mathbf{V}'\mathbf{S}'^{\frac{1}{2}} \in \mathbb{R}^{n\times k}$. The most common way to cluster $\acute{\mathbf{V}}$ is to perform K-means clustering. However, K-means clustering is usually based on random trials, and it can show poor performance when K is large [23]. Instead, we take another approach, which is similar to the optimal discrete solution method in [23]. Our approach is also related to the theory in [27], in which they show that the nonnegative factorization of the doubly stochastic normalization of a kernel matrix will indicate the likelihood of each sample for different clusters. The optimization problem for our approach is given as

$$\min_{\mathbf{Z}}\quad \|\mathbf{Z}-\acute{\mathbf{V}}\mathbf{R}\|_F^2,\quad \text{s.t.}\quad \mathbf{Z}\geq 0,\quad \mathbf{R}^T\mathbf{R}=\mathbf{I},\tag{21}$$

where $\mathbf{Z}$ has the same dimensions of $\acute{\mathbf{V}}$. This problem can be efficiently solved by updating $\mathbf{Z}$ and $\mathbf{R}$ alternatingly. The update formulas are

$$\begin{aligned} \mathbf{Z} &\leftarrow \max(\acute{\mathbf{V}}\mathbf{R}, \mathbf{0}), \\ \mathbf{R} &\leftarrow P_{\perp}(\acute{\mathbf{V}}^T\mathbf{Z}), \end{aligned} \tag{22}$$

where $P_{\perp}(\cdot)$ is a projection operator to the nearest orthogonal matrix, *i.e.*, $P_{\perp}(\mathbf{Y}) = \mathbf{U}_1\mathbf{U}_2^T$ where $\mathbf{Y} = \mathbf{U}_1\mathbf{\Sigma}\mathbf{U}_2^T$ is SVD. The last update formula comes from the orthogonal Procrustes problem [8]. Although this is a non-convex problem, it is not sensitive to the initial choice of $\mathbf{R}$ due to the same reason mentioned in [23]. This procedure converges rapidly, within one or two dozens of iterations. After finding the optimal $\hat{\mathbf{Z}}$, we can find the clusters by finding the maximum element for each row.

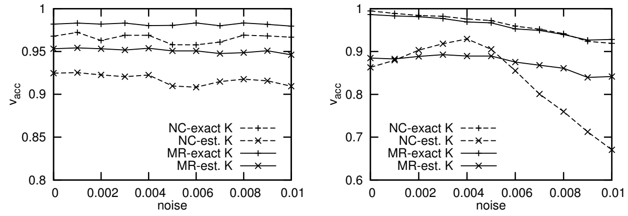
4. Experimental Results

We performed clustering experiments on synthetic toy, face, and motion databases. We have largely followed the experiments in [15] to examine the performance of MR. The parameters of SSC and LRR were set as in the face and motion experiments of [6] and [15], respectively. The performance of MR was compared to the normalized cut (NC) [18], however, we replaced the K-means clustering step with (20) in both MR and the normalized cut for fair comparison, because the random-trial-based K-means clustering often performs poorly when the number of classes was large. We tested both the exact K and that estimated from (4) by adjusting τ for the normalized cut. Note that, for SSC and LRR, we used the code provided by the authors in [6, 15]. MR has also been tested based both on the exact K and that estimated by counting the eigenvalues of $\breve{\mathbf{F}}$ above 0.5. The parameters of MR and the normalized cut has been determined for each type of data so that they yield the best performance.

We used two measures for evaluation: (i) The segmentation accuracy (v_{ACC}) and (ii) the normalized mutual information (v_{NMI}), which are given as follows:

$$\begin{aligned} v_{\text{ACC}} &= \max_{\Gamma} \frac{\sum \delta(r_i, \Gamma(s_i))}{n}, \\ v_{\text{NMI}} &= \frac{I(r_i, s_i)}{(H(r_i) + H(s_i))/2}, \end{aligned} \tag{23}$$

where s_i and r_i are ground truth and the obtained labels, respectively, $\delta(a, b)$ is the Kronecker delta function and Γ is the permutation mapping function. The optimal Γ can be found in polynomial time by the Hungarian algorithm [10]. $I(\cdot, \cdot)$ and $H(\cdot)$ are the mutual information and the entropy, respectively. The values of these measures are in the range of $[0, 1]$ and larger values indicate better performance.



(a) LRR (b) SSC

Figure 2. Experiments on synthetic toy examples: "NC" denotes the normalized cut and "MR" denotes the membership representation. "Exact K" means that the exact K was used for the final clustering step, and "est. K" means that K was estimated based on the estimation technique of each algorithm.

We empirically observed that, generally, SSC often generates "cleaner" latent matrix than LRR, but the characteristics of the generated matrix sensitively change for each data. Accordingly, SSC may yield better results when there is small noise and K is known, but is not robust and not very reliable in estimating K from the latent matrix. This will become apparent in the following experiments.

First, we performed experiments on synthetically generated toy data. We randomly generated 100 test cases, where the number of clusters for each was in the range of $[2, 10]$. The number of samples in each cluster was in the range of $[5, 50]$, and the dimension of each sample was 50. For each cluster, we randomly generated a linear subspace by applying QR decomposition to a number of Gaussian random vectors, with randomly selected rank less than half the number of samples in that cluster. We generated zero-mean Gaussian random vectors as the data samples on that linear subspace, with randomly selected standard deviation (which was also generated by a zero-mean Gaussian number with unit standard deviation) for each axis of the linear subspace. After generating the data, we added Gaussian noise with various standard deviation to examine the robustness of the schemes. The parameters of the normalized cut (for estimating K) was $\tau = 0.45$ for LRR and $\tau = 0.7$ for SSC, respectively, while those of MR was $\lambda_M = 0.01$, $\beta = 0.03$ for LRR and $\lambda_M = 0.2$, $\beta = 0.4$ for SSC.

Figure 2 shows the average clustering performance for the 100 toy data. Here, we did not show v_{NMI}, which had similar characteristics with v_{ACC}, because of the space limitation. For LRR, MR shows better performance than the normalized cut, both for the exact and estimated K. For SSC, the performance for the normalized cut and MR are very close for the exact K. For the estimated K, the normalized cut has a point that shows better performance than MR, but generally it is strongly affected by the level of noise. This is mainly due to the sensitiveness of SSC mentioned earlier. Nevertheless, MR shows steady performance regardless of the amount of noise, due to the "enhancing" effect of its nature.

For the motion clustering experiment, Hopkins155 motion database [21] was used. Hopkins155 motion database contains 156 video sequences of rigidly moving objects, in

Table 1. Clustering performance for Hopkins155 data. The numbers in parentheses are the standard deviations.

		K	v_{ACC}	v_{NMI}	p_K (%)
LRR	NC	exact	0.965 (0.077)	0.883 (0.187)	1
		est.	0.928 (0.107)	0.828 (0.236)	0.744
	MR	exact	**0.966** (0.075)	**0.891** (0.156)	0.949
		est.	**0.941** (0.103)	**0.869** (0.163)	0.814
SSC	NC	exact	**0.974** (0.073)	0.909 (0.192)	1
		est.	0.926 (0.120)	0.808 (0.320)	0.724
	MR	exact	0.970 (0.076)	**0.915** (0.158)	0.974
		est.	**0.939** (0.113)	**0.893** (0.153)	0.801

Table 2. Clustering performance for Yale-Caltech data. Here, "est." was not evaluated for the normalized cut (NC), because it is always possible to tune τ to make the estimated K be the same as the true number of clusters.

	LRR			SSC		
	NC	MR		NC	MR	
	exact	exact	est.	exact	exact	est.
v_{ACC}	**1**	**1**	**1**	0.936	**0.982**	0.950
v_{NMI}	**1**	**1**	**1**	0.961	**0.980**	0.951
K	38	38	38	38	38	40

which significant features are detected and tracked along the frames. Each sequences contains the cluster information, and the numbers of clusters are in between 2 to 5. The parameters of the normalized cut was $\tau = 0.35$ for LRR and $\tau = 0.09$ for SSC, respectively, while those of MR was $\lambda_M = 0.02$, $\beta = 0.2$ for LRR and $\lambda_M = 10^{-8}$, $\beta = 0.2$ for SSC. The reason for using such a small λ_M for SSC was that SSC generates very sparse latent matrix for this type of data, so we had to conserve the nonzero elements in $\mathbf{W}'$ as far as possible. For this experiment, we also measured the rate p_K of correct K estimation by counting the number of sequences of which K was correctly estimated. Table 1 shows the clustering performance for the Hopkins155 data. Here, we indicated the best performance in bold face. We can confirm that MR shows better performance than the normalized cut for most of the time, and even when it is worse, the performance is very similar. The performance gain gets larger when the estimated K is used, which is important because it is difficult to know the number of clusters in advance in practical situations. Note that p_K of MR is less than 100% even when the exact K is used, which is due to some cases that the number of nonzero eigenvalues in $\breve{\mathbf{F}}$ is less than K. In these cases, (mostly two) small clusters are merged into a single cluster. Nevertheless, the accuracy is very similar to that of the normalized cut, which indicates that even though the normalized cut segments the data strictly into K clusters, the accuracy may be worse.

For the face image clustering experiment, the extended Yale database B [12] was used. Following the practice of [15], we combined the images in the extended Yale database B with those in Caltech101 database [13] to form a face data set containing outliers. After finding the latent matrices, the elements corresponding to outliers were removed before applying spectral clustering or MR, as in [15]. For this experiment, the parameters of MR was set to $\lambda_M = 0.07$, $\beta = 0.2$ for LRR and $\lambda_M = 0.08$, $\beta = 0.7$ for SSC. Table 2 shows the clustering performance for the face experiment. Care should be taken that, in this case, there is only one data set unlike the case of Hopkins155, so it is always possible to find τ that gives the correct K. Hence, we did not evaluated the normalized cut for the exact K. Here, for LRR, perfect clustering is possible both for the normalized cut and MR. However, for SSC, the normalized cut with the exact K shows much worse performance than MR even with the estimated K, which indicates that MR indeed improves the latent matrix for better clustering.

We would like to mention that some of the reviewers asked whether applying the doubly stochastic normalization techniques [22, 27, 28] used in spectral clustering can replace the role of MR. We do not show the results in this paper due to the space limitation, however they were very poor. The reason is that the latent matrices from LRR or SSC are sparse, which are not suitable for the doubly stochastic normalization that are mainly designed for the kernel matrices. On the contrary, MR was designed for sparse matrices, which at this stage is not good for general spectral clustering problem.

In summary, the proposed algorithm, MR, can act as a robust post-processing step for subspace clustering. It depends much less on the characteristics of the data than the combination of a heuristic estimator and the normalized cut, and gives a more reliable form to estimate the number of clusters.

5. Conclusion

In this paper, we proposed the membership representation (MR) to detect block-diagonal structure from the output of subspace clustering. MR is based on a self-expressive system that expresses the latent matrix by a Hadamard product of a membership matrix and itself. Since this problem is NP-hard, we solve the convex relaxation of the problem, and then transform the result to a doubly stochastic matrix, which is a better form for the input of spectral clustering. The output matrix of our method is more reliable to estimate the number of clusters for spectral clustering, and a simple eigenvalue-counting method was sufficient to find a good clustering result in our experiments. Detecting block-diagonal structure from a matrix is a fundamental problem with many potential applications, thus applying MR to other problems will also be an interesting issue. For example, extending this technique to the general spectral clustering problem will be very interesting, which is not possible at this stage as mentioned earlier. On the other hand, since the proposed method may suffer scalability problems for large data sets because it requires an eigenvalue decomposition for each iteration step, finding a modification of the formulation as in [30] to achieve scalability is also an important issue, which is also left as a future work.

Acknowledgment

The work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2014R1A1A1006269).

References

[1] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Science*, 2(1):183–202, 2009.

[2] E. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *J. ACM*, 58(3), May 2011.

[3] Y. Chen, A. Jalali, S. Sanghavi, and H. Xu. Clustering partially observed graphs via convex optimization. *J. Machine Learning Research*, 15:2213–2238, June 2014.

[4] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan. Multi-task low-rank affinity pursuit for image segmentation. In *Proc. IEEE Int'l Conf. Computer Vision*, November 2011.

[5] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2009.

[6] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, January 2013.

[7] P. Favaro, R. Vidal, and A. Ravichandran. A closed form solution to robust subspace estimation and clustering. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2011.

[8] J. C. Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, March 1975.

[9] A. Jalali and N. Srebro. Clustering using max-norm constrained optimization. In *Proc. Int'l Conf. Machine Learning*, July 2012.

[10] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955.

[11] F. Lauer and C. Schnorr. Spectral clustering of linear subspaces for motion segmentation. In *Proc. IEEE Int'l Conf. Computer Vision*, September 2009.

[12] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(5):684–698, May 2005.

[13] F.-F. Li, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *Workshop of IEEE Conf. Computer Vision and Pattern Recognition*, June 2004.

[14] Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Technical Report UILU-ENG-09-2215, University of Illinois at Urbana-Champaign, November 2009.

[15] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(1):171–184, January 2013.

[16] C. Mathieu and W. Schudy. Correlation clustering with noisy input. In *Proc. ACM-SIAM Symp. Discrete Algorithms*, January 2010.

[17] Y. Ni, J. Sun, X. Yuan, S. Yan, and L.-F. Cheong. Robust low-rank subspace segmentation with semidefinite guarantees. In *IEEE Int'l Conf. Data Mining Workshops*, December 2010.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

[19] R. Sibson. SLINK: An optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1):30–34, 1973.

[20] C. Swamy. Correlation clustering: maximizing agreements via semidefinite programming. In *Proc. ACM-SIAM Symp. Discrete Algorithms*, January 2004.

[21] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2007.

[22] Y. Yan, C. Shen, and H. Wang. Efficient semidefinite spectral clustering via lagrange duality. *IEEE Trans. Image Processing*, 23(8):3522–3534, August 2014.

[23] S. X. Yu and J. Shi. Multiclass spectral clustering. In *Proc. IEEE Int'l Conf. Computer Vision*, October 2003.

[24] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *J. Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, February 2006.

[25] L. Zappella, X. Lladó, E. Provenzi, and J. Salvi. Enhanced local subspace affinity for feature-based motion segmentation. *Pattern Recognition*, 44(2):454–470, February 2011.

[26] L. Zappella, E. Provenzi, X. Lladó, and J. Salvi. Adaptive motion segmentation algorithm based on the principal angles configuration. *Computer Vision-ACCV 2010, Lecture Notes in Computer Science*, 6494:15–26, 2011.

[27] R. Zass and A. Shashua. A unifying approach to hard and probabilistic clustering. In *Proc. IEEE Int'l Conf. Computer Vision*, October 2005.

[28] R. Zass and A. Shashua. Doubly stochastic normalization for spectral clustering. In *Proc. Neural Information Processing Systems*, December 2007.

[29] H. Zha, X. He, C. Ding, M. Gu, and H. D. Simon. Spectral relaxation for k-means clustering. In *Proc. Neural Information Processing Systems*, December 2002.

[30] Y. Zheng, G. Liu, S. Sugimoto, S. Yan, and M. Okutomi. Practical low-rank matrix approximation under robust l1-norm. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2012.

[31] Y. Zhu, D. Huang, F. De la Torre, and S. Lucey. Complex non-rigid motion 3d reconstruction by union of subspaces. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2014.

[32] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *J. Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, April 2005.