

Robust subspace clustering via symmetry constrained latent low rank representation with converted nuclear norm

Xian Fang^{a,*}, Zhixin Tie^{a,*}, Feiyang Song^a, Jialiang Yang^{b,c}

^a School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, PR China

^b School of Mathematics and Statistics, Hainan Normal University, Haikou 570100, PR China

^c Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai, New York 10029, USA

ARTICLE INFO

Article history:

Received 8 May 2018

Revised 31 January 2019

Accepted 27 February 2019

Available online 6 March 2019

Communicated by Dr Weiguo Sheng

Keywords:

Cluster analysis

Robust subspace clustering

Latent low rank representation

Symmetric constraint

Converted nuclear norm

Spectral clustering

ABSTRACT

Subspace clustering, which is devoted to classifying data samples derived from a union of linear subspaces, has been widely applied to various fields such as pattern recognition, artificial intelligence and computer vision. In this paper, we propose a symmetry constrained latent low rank representation with converted nuclear norm (SLLRR) algorithm for robust subspace clustering, which extends the original latent low rank representation (LLRR) algorithm by introducing a kind of converted nuclear norm and integrating strategy of the symmetric constraint. SLLRR both enhances the sparsity of the coefficient matrix and guarantees weight consistency for each pair of data samples when seeking the low rank representation. The symmetric coefficient matrix that will no longer be overly dense is acquired by the inexact augmented Lagrange multipliers (IALM) method. With further exploiting the angular information of the principal directions of that, the sparse affinity matrix for spectral clustering is identified. Extensive experimental results on face clustering and motion segmentation datasets confirm the availability and robustness of the proposed algorithm, and it is highly competitive compared to the state-of-the-art subspace clustering algorithms.

© 2019 Elsevier B.V. All rights reserved.

1. Introduction

As an unsupervised learning method, cluster analysis is remarkably significant in data mining and exploratory data analysis. The purpose of clustering is that dividing the objects into different clusters or classes according to the similarity of data samples. With the rapid emergence of vast amount of complex high dimensional data, traditional clustering algorithms such as density-based [1–3], grid-based [4] and model-based [5–7] methods are difficult or unable to clustering correctly due to the curse of dimensionality. Subspace clustering has aroused rising attention among researchers recently for its excellent ability of clustering high dimensional data. This sort of approach refers to classifying data samples derived from a union of linear subspaces, which has been successfully applied to pattern recognition [8,9], artificial intelligence [10], computer vision [11] and many more.

Currently, the leading subspace clustering algorithms can be normally grouped into five categories, namely, algebraic [12,13], iterative [14,15], statistical [16,17], factorization-based [18] and

spectral clustering-based [19,20] methods. Among them, spectral clustering-based method is the most popular one, and has two representative branch algorithms which are sparse subspace clustering (SSC) [21] and low rank representation (LRR) [22]. Such algorithm performs subspace clustering in two steps: first learns a representation coefficient matrix in order to further constructs an affinity matrix that encodes the subspace membership information from the given data, then gets the final clustering results by applying certain spectral clustering algorithm such as normalized cuts (NCut) [23] on affinity matrix. In the case that the data vectors are sorted column by column according to the class, affinity matrix as well as coefficient matrix will possess block diagonal structure, which reveals the subspace characteristic of data. The major distinction between SSC and LRR is how to effectively grasp the potential relationship of data points to obtain a good affinity matrix.

SSC makes a hypothesis that each data point can be represented by linear combination of other data points existing in the same subspace. Lots of modifications and developments of SSC have become active area of research. For instance, Pham et al. [24] put forward the improved weighted sparse subspace clustering (WSSC) by embedding weight into coefficient matrix relying on spatial relationship be-

* Corresponding author.

E-mail addresses: xianfangfx@163.com (X. Fang), tiezx@zstu.edu.cn (Z. Tie).

tween data points, which corrects sparse representation in a principled manner and without bringing much computational overhead. Saha et al. [25] combined Laplacian regularization term and group sparse coding to design the graph regularized group sparse subspace clustering (GRSSC), satisfactory results was achieved by this way. Chen et al. [26] implemented the active orthogonal matching pursuit-based sparse subspace clustering (AOMPSSC), which enjoys both great time efficiency and clustering accuracy by adaptively updating data points and randomly dropping data points in the orthogonal matching pursuit process. However, SSC just takes into account the sparse representation of the data points in each individual subspace and lacks the guidance of global structure of the whole data.

LRR also describes and characterizes the relationship among data points by adopting the theory of linear combination. Unlike SSC, LRR seeks low rank representation of all data points jointly that include even the data point itself. In contrast, it can better capture the global information around each sample and exhibits favorable application prospect. To improve the overall performance of LRR, researchers have spared no effort to work from many aspects. For instance, when facing with the insufficient data sample, the latent low rank representation (LLRR) was brought forward in [27]. Feng et al. [28] directly pursued the block diagonal structure by presenting a graph Laplacian constraint based formulation, the block diagonal low rank representation (BDLRR) was designed. Wang et al. [29] proposed the constrained low rank representation (CLRR), which ensures the data that share a must-link constraint or the same label have the same coordinates in the new representation and are clustered into the same subspace. Considering that the influence of symmetric components, the low rank representation with symmetric constraint (LRRSC) was presented in [30], which imposes the condition of symmetric constraint on the representation coefficient matrix and exploits the intrinsically geometrical structure of the memberships of samples. The upgraded version of LRRSC called the symmetric low rank representation (SLRR) was stated in [31], where collaborative representation combined with low rank matrix recovery techniques avoids iterative singular value decomposition (SVD) operation and computational complexity can be significantly decreased. Fang et al. [32] developed the non-negative low rank representation (NNLRR) to merge the affinity matrix construction and subspace clustering into one step for searching for the overall optimum. Although these algorithms based on low rank representation above have made impressive achievement, there are still the common inherent shortcoming. Specifically, since the rank minimization of a matrix is NP-hard problem and the nuclear norm is convex envelop of the rank function, researchers generally minimize the nuclear norm instead of the rank function to constraint representation coefficient in LRR-based algorithms. It is known that the rank function counts the number of non-zero singular values of a matrix, while the nuclear norm sums their numerical numbers. As a result, the nuclear norm may be dominated by a few very large singular values [33]. In view of this issue, a series of non-convex surrogate functions have come up to better approximate the rank function, such as Logarithm Determinant [34], Schatten- p norm [35], truncated nuclear norm [36] and others.

In this paper, we define a novel converted nuclear norm acting on LLRR framework that encourages more sparse representation coefficient. In addition, we draw advantage of symmetric constraint, which learns a low rank representation coefficient matrix in symmetric form solution by addressing the symmetric low rank optimization problem. Therefore, the symmetry constrained latent low rank representation with converted nuclear norm (SLLRRC) algorithm is proposed for robust subspace clustering.

The rest of this paper is organized as follows. Section 2 reviews the related works including basic LRR and its vari-

ants. Section 3 provides detailed description of the proposed SLLRRC. Section 4 gives the experimental results and comparative analyses followed by the conclusions are drawn in Section 5.

2. Related works

Before reviewing previous work, we give the explanation of some important notations. All matrices and vectors in this paper are represented with uppercase and lowercase symbols, respectively. For a matrix M , M_{ij} is its (i, j) th entry, $M_{i,:}$ is its i -th row and $M_{:,j}$ is its j th column. For a vector v , v_i is its i th entry. In particular, M^T and M^{-1} are the transpose and inverse for matrix M , respectively, while v^T is the transpose for vector v . I is used to denote the identity matrix, $\text{tr}(\cdot)$ is the trace of a matrix. $\|M\|_1$, $\|M\|_{2,1}$, $\|M\|_F^2$ and $\|M\|_*$ denote the ℓ_1 norm, $\ell_{2,1}$ norm, squared Frobenius norm and nuclear norm of matrix M , $\|v\|_1$ and $\|v\|_2$ denote the ℓ_1 norm and ℓ_2 norm of vector v , respectively.

2.1. Subspace clustering by low rank representation

Suppose that there is a high dimensional space dataset $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{m \times n}$ with m attributes and n samples drawn from a union of k linear low dimensional subspaces $\bigcup_{i=1}^k S_i$, where S_i ($i = 1, 2, \dots, k$) corresponds to one class ideally. Low rank representation (LRR) algorithm [22] aims to recover a low rank column coefficient matrix $Z \in \mathbb{R}^{n \times n}$ that reflects the structure of original data X under a given dictionary. In practice, it is frequent to choose the data matrix X itself as the dictionary. LRR solves the following problem in the case of data grossly corrupted by noise:

$$\min_{Z, E} \text{rank}(Z) + \lambda \|E\|_\ell \quad \text{s.t. } X = XZ + E \quad (1)$$

where $\lambda \in (0, \infty)$ is the regularization parameter, $E \in \mathbb{R}^{m \times n}$ is the sparse noise matrix and $\|\cdot\|_\ell$ denotes certain norm. The $\ell_{2,1}$ norm, ℓ_1 norm and squared Frobenius norm are candidates that are worth considering here in terms of $\|\cdot\|_\ell$.

The optimization problem (1) essentially belongs to NP-hard problem due to the discrete nature of the rank optimal solution. Related research [37] suggests that the optimization problem of the nuclear norm can be substituted for that of the rank function under certain conditions. In addition, the $\ell_{2,1}$ norm is adopt to characterize the error term E here: $\|E\|_{2,1} = \sum_{j=1}^n (\sum_{i=1}^m E_{ij}^2)^{1/2}$. Hence the tractable and equivalent relaxation of the problem (1) is given as the problem (2):

$$\min_{Z, E} \|Z\|_* + \lambda \|E\|_{2,1} \quad \text{s.t. } X = XZ + E \quad (2)$$

where the nuclear norm (i.e., the sum of the singular values σ of a matrix) is formulated as follows:

$$\|\cdot\|_* = \sum_{i=1}^{\text{rank}(\cdot)} \sigma_i \quad (3)$$

As a convex optimization problem, the problem (2) can be effectively solved by the inexact augmented Lagrange multipliers (IALM) [38], also known as the alternating direction method of multipliers (ADMM). Once coefficient matrix Z is obtained by LRR, affinity matrix $W = |Z| + |Z^T|$ can be identified to perform spectral clustering, the clustering results can be finally produced.

2.2. Variants of LRR

Although basic LRR has been proved to be powerful to discover the global intrinsic structure of datasets and less sensitive to corrupted data, it may fail to perform excellently when data samples are insufficient. To overcome this limitation, the latent low rank representation (LLRR) [27] is proposed. The core idea of LLRR is to

put low rank constraint on both column and row coefficient matrices, and the data matrix X will be reconstructed from two directions, namely, column (observed data samples) and row (unobserved data samples). LLRR adopts the ℓ_1 norm to characterize the error term E here: $\|E\|_1 = \sum_{i=1}^m \sum_{j=1}^n |E_{ij}|$, and solves the following problem in the case of data grossly corrupted by noise:

$$\min_{Z, L, E} \|Z\|_* + \|L\|_* + \lambda \|E\|_1 \quad \text{s.t. } X = XZ + LX + E \quad (4)$$

where $L \in \mathbb{R}^{m \times m}$ is the row coefficient matrix. The term XZ denotes the linear combinations of the observed data sample and the term LX denotes the linear combinations of the hidden data sample. When some data points suffer from loss, it is helpful to recover the hidden data sample used the observed data sample.

Inspired by the LRR with PSD constrains (LRRPSD) algorithm [39], Chen et al. [30] proposed the low rank representation with symmetric constraint (LRRSC). The symmetric constraint employed in LRRSC ensures the weight consistency for each pair of data points, keeps the subspace structure of data. This is a critical step towards evaluating the memberships between data points so that correlated data points of subspaces are represented together. LRRSC reserves the $\ell_{2,1}$ norm to characterize the error term E , and solves the following problem in the case of data grossly corrupted by noise:

$$\min_{Z, E} \|Z\|_* + \lambda \|E\|_{2,1} \quad \text{s.t. } X = XZ + E, Z = Z^T \quad (5)$$

3. Symmetry constrained latent low rank representation with converted nuclear norm

In this section, the proposed symmetry constrained latent low rank representation with converted nuclear norm (SLLRRC) algorithm extended from the LLRR framework is introduced. At first, we define novel converted nuclear norm to carry out nonlinear mapping on the basic nuclear norm, where the affinity matrix can become sparser for the purpose that the inter-clusters similarity is as high as possible and intra-cluster similarity is as low as possible. Then, the strategies of the targeted selection of regularization term and the symmetric constraint are simultaneously taken into consideration. At last, implementation process and computational complexity analysis of the SLLRRC algorithm are given.

3.1. The proposed SLLRRC

Nikolova [40] first presented the concept of transformed ℓ_1 penalty function in the variable selection in 2000. After that, slight modification was investigated by Lv et al. [41] in 2009. Based on that, we make further modification to our experiment, the converted function is defined as follows:

$$\xi_\tau(x) = \frac{(1 + \tau)|x|}{1 + \tau|x|} \quad (6)$$

where $\tau \in (0, \infty)$ is a tuning parameter. It is easy to see that Eq. (6) is equivalent to the following Eq. (7):

$$\xi_\tau(x) = \frac{\tau|x|}{1 + \tau|x|} + \left(\frac{1}{1 + \tau|x|}\right)|x| \quad (7)$$

This function interpolates the ℓ_0 norm and ℓ_1 norm as follows:

$$\begin{cases} \lim_{\tau \rightarrow 0^+} \xi_\tau(x) = |x| \\ \lim_{\tau \rightarrow \infty} \xi_\tau(x) = \begin{cases} [1, 1, \dots, 1]^T, & x \neq 0 \\ 0, & x = 0 \end{cases} \end{cases} \quad (8)$$

Moreover, it also satisfies the following fundamental properties:

Property 1. nonnegativity

$$\xi_\tau(x_i) \geq 0 \text{ and } \xi_\tau(x_i) = 0 \Leftrightarrow x_i = 0 \quad (9)$$

Property 2. triangle inequality

$$\xi_\tau(|x_i| + |x_j|) \leq \xi_\tau(|x_i|) + \xi_\tau(|x_j|) \quad (10)$$

Property 3. nonhomogeneity

$$\xi_\tau(c|x_i|) \begin{cases} \leq |c|\xi_\tau(|x_i|), & |c| \geq 1 \\ > |c|\xi_\tau(|x_i|), & |c| < 1 \end{cases} \quad (11)$$

Proof. (of property 2) As $\xi_\tau(x)$ is a strict monotone increasing function in positive argument, by triangle inequality $|x_i| + |x_j| \leq |x_i| + |x_j| + \tau|x_i x_j|$, we have:

$$\begin{aligned} \xi_\tau(|x_i| + |x_j|) &\leq \xi_\tau(|x_i| + |x_j| + \tau|x_i x_j|) \\ &= \frac{(1 + \tau)(|x_i| + |x_j| + \tau|x_i x_j|)}{1 + \tau(|x_i| + |x_j| + \tau|x_i x_j|)} \\ &\leq \frac{(1 + \tau)(|x_i| + |x_j| + 2\tau|x_i x_j|)}{1 + \tau(|x_i| + |x_j| + \tau|x_i x_j|)} \\ &= \xi_\tau(|x_i|) + \xi_\tau(|x_j|) \end{aligned} \quad (12)$$

□

Proof. (of property 3) For any given variable $c \in \mathbb{R}$, we have:

$$\xi_\tau(c|x_i|) = \frac{(1 + \tau)|c||x_i|}{1 + \tau|c||x_i|} = |c|\xi_\tau(|x_i|) \frac{1 + \tau|x_i|}{1 + \tau|c||x_i|} \quad (13)$$

If $|c| \geq 1$, then $\frac{1 + \tau|x_i|}{1 + \tau|c||x_i|} \leq 1$, hence $\xi_\tau(c|x_i|) \leq |c|\xi_\tau(|x_i|)$ hold. Similarly, when $|c| < 1$, we get that $\xi_\tau(c|x_i|) > |c|\xi_\tau(|x_i|)$ hold.

Thus, the function can approximate well both ℓ_0 and ℓ_1 norms with the change of parameter τ and the sparsity can be guaranteed.

The nuclear norm of the coefficient matrix in LLRR is to be minimized in objective function, which has been proven to be practical convex surrogate of the rank function. Nevertheless, the nuclear norm just adds all singular values of a matrix together, the approximation error may depend heavily on the numerical numbers of the singular values. The coefficient matrix obtained by simple nuclear norm processing may not be sparse enough, which is a bottleneck of LLRR. To this end, we make use of the properties of Eq. (6) and propose a converted nuclear norm. The novel converted nuclear norm function is defined as follows:

$$\|\cdot\|_{C*} = \sum_{i=1}^{\text{rank}(\cdot)} \xi_\tau(\sigma_i) \quad (14)$$

Comparing the converted nuclear norm and standard nuclear norm, we can find that the former carries out nonlinear mapping on the basis of the latter so that the results can be as sparse as possible under the premise of guaranteeing the low rank.

Usually, the error term E is closely related to three typical errors under the task of subspace clustering. In particular, for the sample-specific corruptions and outliers (i.e., some data vectors are corrupted and others are not), the $\ell_{2,1}$ norm should be chosen. For the random corruptions, the ℓ_1 norm is especially appropriate. For small Gaussian noise, the squared Frobenius norm is often selected. LRR assumes that the error is the first type, while LLRR prefers to meet the second one. Unlike them, we focus on the characteristics of error term E in the actual data and resort to reliable norm. For example, face clustering dataset are contaminated by sparse gross errors, the squared Frobenius norm is adopted to characterize E : $\|E\|_F^2 = \sum_{i=1}^m \sum_{j=1}^n E_{ij}^2$, motion segmentation dataset are slightly corrupted by noise, the $\ell_{2,1}$ norm is adopted. At the same time, we enforce symmetric constraint both column coefficient matrix Z and row coefficient matrix L on LLRR.

Considering comprehensively the above mentioned factors, the proposed SLLRRC solves the following problem in the case of data

grossly corrupted by noise:

$$\begin{aligned} \min_{Z, L, E} & \|Z\|_{C*} + \|L\|_{C*} + \lambda \|E\|_{\ell} \\ \text{s.t. } & X = XZ + LX + E, Z = Z^T, L = L^T \end{aligned} \quad (15)$$

3.2. Optimization

As the problem (15) is convex, there are many ways to deal with it, such as iterative thresholding (IT) [42], augmented Lagrange multipliers (ALM) [43] and accelerated proximal gradient (APG) [44]. Like LLRR, we also employ the IALM approach. Therefore, the problem (15) can be first rewritten as the problem (16) by introducing two auxiliary variable $J \in \mathbb{R}^{n \times n}$ and $S \in \mathbb{R}^{m \times m}$ as follows:

$$\begin{aligned} \min_{Z, L, E, J=S^T} & \|J\|_{C*} + \|S\|_{C*} + \lambda \|E\|_{\ell} \\ \text{s.t. } & X = XZ + LX + E, Z = J, J = J^T, L = S, S = S^T \end{aligned} \quad (16)$$

The problem (16) can be then transformed into unconstrained optimization problem by adding Lagrange multipliers as follows:

$$\begin{aligned} \min_{Z, L, E, J=S^T, S=S^T} & \|J\|_{C*} + \|S\|_{C*} + \lambda \|E\|_{\ell} \\ & + \text{tr}(Y_1^T (X - XZ - LX - E)) + \text{tr}(Y_2^T (Z - J)) + \text{tr}(Y_3^T (L - S)) \\ & + \frac{\mu}{2} (\|X - XZ - LX - E\|_F^2 + \|Z - J\|_F^2 + \|L - S\|_F^2) \end{aligned} \quad (17)$$

where $Y_1 \in \mathbb{R}^{m \times n}$, $Y_2 \in \mathbb{R}^{n \times n}$ and $Y_3 \in \mathbb{R}^{m \times m}$ are the Lagrange multipliers, and μ is the penalty parameter. The objective function can be addressed by alternately updating J , S , Z , L and E with fixing the other variables [45,46].

1. Update J with fixing the other four variables

Optimization of Eq. (17) with respect to J involves the follows:

$$J = \arg \min_{J=J^T} \|J\|_{C*} + \text{tr}(Y_2^T (Z - J)) + \frac{\mu}{2} \|Z - J\|_F^2 \quad (18)$$

Eq. (18) can be reformulated as follows:

$$J = \arg \min_{J=J^T} \frac{1}{\mu} \|J\|_{C*} + \frac{1}{2} \|J - (Z + \frac{Y_2}{\mu})\|_F^2 \quad (19)$$

Eq. (19) is convex minimization problem, it has unique closed-form solution, which can be solved using the singular value thresholding (SVT) operator [47], i.e., $J = U \varpi_{\{\frac{1}{\mu} \xi_{\tau}(\sigma)\}}(\Sigma) V^T$, where U and V are the orthonormal matrices and Σ is the diagonal element set of diagonal matrix from the SVD of symmetric matrix $\tilde{D}$, i.e., $\tilde{D} = U \Sigma V^T$, where $\tilde{D} = \frac{D + D^T}{2}$, $D = Z + \frac{Y_2}{\mu}$, and $\varpi_{(\varphi)}(\theta)$ is the soft thresholding operator:

$$\varpi_{(\varphi)}(\theta) = \begin{cases} \theta + \varphi, & \theta < -\varphi \\ 0, & -\varphi \leq \theta \leq \varphi \\ \theta - \varphi, & \theta > \varphi \end{cases} \quad (20)$$

where $\theta \in \mathbb{R}$ and $\varphi \in (0, \infty)$.

2. Update S with fixing the other four variables

Optimization of Eq. (17) with respect to S involves the follows:

$$S = \arg \min_{S=S^T} \|S\|_{C*} + \text{tr}(Y_3^T (L - S)) + \frac{\mu}{2} \|L - S\|_F^2 \quad (21)$$

Eq. (21) can be reformulated as follows:

$$S = \arg \min_{S=S^T} \frac{1}{\mu} \|S\|_{C*} + \frac{1}{2} \|S - (L + \frac{Y_3}{\mu})\|_F^2 \quad (22)$$

Similar to the above processing of updating J , Eq. (22) is also solved using SVT operator.

3. Update Z with fixing the other four variables

We drop the irrelevant terms of Z in Eq. (17), then we have:

$$\begin{aligned} Z = \arg \min_Z & \text{tr}(Y_1^T (X - XZ - LX - E)) + \text{tr}(Y_2^T (Z - J)) \\ & + \frac{\mu}{2} (\|X - XZ - LX - E\|_F^2 + \|Z - J\|_F^2) \end{aligned} \quad (23)$$

Eq. (23) is a strongly convex quadratic function, which can be solved directly. By taking the derivation of it with respect to Z and setting to be zero, we have:

$$Z = (I + X^T X)^{-1} \left(X^T (X - LX - E) + J + \frac{X^T Y_1 - Y_2}{\mu} \right) \quad (24)$$

4. Update L with fixing the other four variables

We drop the irrelevant terms of L in Eq. (17), then we have

$$\begin{aligned} L = \arg \min_L & \text{tr}(Y_1^T (X - XZ - LX - E)) + \text{tr}(Y_3^T (L - S)) \\ & + \frac{\mu}{2} (\|X - XZ - LX - E\|_F^2 + \|L - S\|_F^2) \end{aligned} \quad (25)$$

Similar to the above processing of updating Z , Eq. (25) is also solved directly. By taking the derivation of it with respect to L and setting to be zero, we have

$$L = \left((X - XZ - E) X^T + S + \frac{Y_1 X^T - Y_3}{\mu} \right) (I + X^T X)^{-1} \quad (26)$$

5. Update E with fixing the other four variables

We collect the relevant terms of E in Eq. (17), then we have

$$E = \lambda \|E\|_{\ell} + \text{tr}(Y_1^T (X - XZ - LX - E)) + \frac{\mu}{2} \|X - XZ - LX - E\|_F^2 \quad (27)$$

Eq. (27) can be reformulated as follows:

$$E = \arg \min_E \frac{\lambda}{\mu} \|E\|_{\ell} + \frac{1}{2} \|E - (X - XZ - LX + \frac{Y_1}{\mu})\|_F^2 \quad (28)$$

Eq. (28) is convex minimization problem, it has different kinds of closed-form solutions, which depends on different regularization term constraint, namely, squared Frobenius norm or $\ell_{2,1}$ norm. For squared Frobenius norm, it can be solved directly as follows:

$$E = \frac{Y_1 + \mu (X - XZ - LX)}{\mu + 2\lambda} \quad (29)$$

For $\ell_{2,1}$ norm, it can be reformulated as follows:

$$E = \arg \min_E \frac{\lambda}{\mu} \|E\|_{2,1} + \frac{1}{2} \|E - Y\|_F^2 \quad (30)$$

where $Y = X - XZ - LX + \frac{Y_1}{\mu}$. Eq. (30) can be solved by the following Lemma 1:

Lemma 1 [48]. Given a matrix $Q = [q_1, q_2, \dots, q_n] \in \mathbb{R}^{m \times n}$. If P is the optimal solution of

$$P = \arg \min_P \gamma \|P\|_{2,1} + \frac{1}{2} \|P - Q\|_F^2 \quad (31)$$

then the j th column of P is

$$P_{:,j} = \begin{cases} \frac{\|Q_{:,j}\|_2 - \gamma}{\|Q_{:,j}\|_2}, & \gamma < \|Q_{:,j}\|_2 \\ 0, & \gamma \geq \|Q_{:,j}\|_2 \end{cases} \quad (32)$$

The complete optimization process to solve the problem (17) is summarized in Algorithm 1.

Algorithm 1 Solving the problem (17) by IALM.**Input:** data matrix X , parameters λ and τ **Initialize:** $Z = J = 0$, $L = S = 0$, $E = 0$, $Y_1 = 0$, $Y_2 = 0$, $Y_3 = 0$, $\mu = 10^{-6}$, $\mu_{\max} = 10^6$, $\rho = 1.5$ and $\varepsilon = 10^{-6}$ **while** not converged **do**1. Fix the others and update J by formula

$$J = \arg \min_J \frac{1}{\mu} \|J\|_{C_*} + \frac{1}{2} \|J - \left(Z + \frac{Y_2}{\mu}\right)\|_F^2$$

2. Fix the others and update S by formula

$$S = \arg \min_S \frac{1}{\mu} \|S\|_{C_*} + \frac{1}{2} \|S - \left(L + \frac{Y_3}{\mu}\right)\|_F^2$$

3. Fix the others and update Z by formula

$$Z = (I + X^T X)^{-1} \left(X^T (X - LX - E) + J + \frac{X^T Y_1 - Y_2}{\mu} \right)$$

4. Fix the others and update L by formula

$$L = \left((X - XZ - E)X^T + S + \frac{Y_1 X^T - Y_3}{\mu} \right) (I + X^T X)^{-1}$$

5. Fix the others and update E by formula

$$E = \arg \min_E \frac{\lambda}{\mu} \|E\|_t + \frac{1}{2} \|E - \left(X - XZ - LX + \frac{Y_1}{\mu}\right)\|_F^2$$

6. Update the multipliers Y_1 , Y_2 and Y_3 by formulas

$$Y_1 = Y_1 + \mu(X - XZ - LX - E)$$

$$Y_2 = Y_2 + \mu(Z - J)$$

$$Y_3 = Y_3 + \mu(L - S)$$

7. Update the parameter μ by formula

$$\mu = \min(\mu_{\max}, \rho\mu)$$

8. Judge whether to meet the convergence conditions

$$\|X - XZ - LX - E\|_\infty < \varepsilon, \|Z - J\|_\infty < \varepsilon \text{ and } \|L - S\|_\infty < \varepsilon$$

end while**Output:** matrix Z , L and E **Algorithm 2** Subspace clustering by SLLRRC.**Input:** data matrix X , parameters λ , α and τ 1. Obtain the optimal solution (Z, L, E) by Algorithm 1 on solving the following problem:

$$\min_{Z, L, E} \|Z\|_{C_*} + \|L\|_{C_*} + \lambda \|E\|_t \quad \text{s.t. } X = XZ + LX + E, Z = Z^T, L = L^T$$

2. Implement the SVD of matrix Z by formula:

$$Z = U \Sigma V^T$$

3. Calculate the procedure matrix M or N by formula:

$$M = U(\Sigma)^{1/2} \text{ or } N = (\Sigma)^{1/2} V^T$$

4. Construct the affinity matrix W by formula:

$$W_{ij} = \left(\frac{Mr_i Mr_j^T}{\|Mr_i\|_2 \|Mr_j\|_2} \right)^{2\alpha} \text{ or } W_{ij} = \left(\frac{Nc_i^T Nc_j}{\|Nc_i\|_2 \|Nc_j\|_2} \right)^{2\alpha}$$

5. Apply W to perform spectral clustering**Output:** the clustering error

Updating both J and S require to computing the SVD of a matrix, which costs $O(n^3)$ for J with $n \times n$ matrix and $O(m^3)$ for S with $m \times m$ matrix. Updating both Z and L involve matrix multiplication, which costs $O(mn^2)$ and $O(m^2n)$, respectively. Updating E costs at most $O(n^3)$ for Frobenius norm operation. Thus, the time complexity of Algorithm 1 is $O(2n^3 + m^3 + mn^2 + m^2n)$ and Algorithm 2 is $O(t(2n^3 + m^3 + mn^2 + m^2n) + n^3)$, where t is the number of iterations.

In terms of computational costs, we want to analyze the relationships among SLLRRC and the state-of-art approaches such as SSC [21], LRR [22], LLRR [27], LRRSC [30] and FLLRR [49] from a qualitative level. The time complexity of SLLRRC can be formalized as $O(t(n^3 + m^3))$. The computation in the SLLRRC is very similar to the LLRR, except that it has an additional step of the converted norm operation. However, its dominant complexity is in the same order as with the LLRR. There is no SVD has to be performed in the FLLRR, thus its time complexity is reduced to 1/2 of the LLRR [49]. In the LRR, the time complexity is $O(t(2n^3 + mn^2) + n^3)$, which can be formalized as $O(tn^3)$ if $n \gg m$ [30]. The LRRSC is also maintained at a same order of magnitude as the LRR. For the SSC, the time complexity of which is usually slightly higher than that of the LRR [50]. Further more, the quantitative comparisons of computational costs of algorithms are given in the experiment in Scheme 2 of Section 4.2.2.

4. Experiments and discussions**4.1. Experimental setup**

All of the experiments are coded and conducted on the Matlab R2013b platform of Windows 7 operating system with Intel Core i3-2100 CPU and 4GB memory. We compared the SLLRRC algorithm performance against the state-of-the-art subspace clustering algorithms including SSC, LRR, LLRR, LRRSC and FLLRR. The capacity of those algorithms is evaluated and discussed on two types of datasets, namely, face clustering and motion segmentation datasets. As for the metric criteria, we always judge by the clustering error, which is defined as the ratio of the number of misclassifications to the total number of samples. The best results are portrayed in boldface.

Tables 1 and 2 summarize the parameters setting of different algorithms on all datasets in the experiments. For SLLRRC, there are three parameters (λ , α and τ) that need to be predetermined. For SSC, it requires one parameter (λ_e or λ_z) in advance. While for the rest of the algorithms, we must set suitable values for two parameters (λ and α). Specifically, the parameter λ , λ_e or λ_z , a priori

3.3. Constructing affinity matrix

Next, we need utilize the optimal low rank matrix Z obtained by Algorithm 1 to construct affinity matrix W of an undirected graph. Let $G = (\tilde{V}, \tilde{E})$ be the undirected graph associated with affinity matrix, where $\tilde{V} = \{v_1, v_2, \dots, v_n\}$ is the vertex set, $\tilde{E} = \{e_{ij} | i, j \in \tilde{V}\}$ is the edge group and $W = \{w_{ij} | i, j \in \tilde{V}\}$ is the adjacency matrix. The relationship among $\tilde{V}$, $\tilde{E}$ and W is that any edge e_{ij} between vertices v_i and v_j carries weight w_{ij} , which should be equivalent to w_{ji} . We no longer construct the affinity matrix in the form of $W = |Z| + |Z^T|$, but rather the form related to the angular information of the principal directions of the low rank representation. In other words, we implement the SVD of matrix Z , i.e., $Z = U \Sigma V^T$, and calculate the procedure matrix $M = U(\Sigma)^{1/2}$ or $N = (\Sigma)^{1/2} V^T$ for defining affinity matrix, in which $MN = Z$. As Z is symmetric matrix, the absolute values of the the columns of U and V are agree, and they can span the principal directions of the symmetric matrix Z . The affinity matrix is defined as follows:

$$W_{ij} = \left(\frac{Mr_i Mr_j^T}{\|Mr_i\|_2 \|Mr_j\|_2} \right)^{2\alpha} \text{ or } W_{ij} = \left(\frac{Nc_i^T Nc_j}{\|Nc_i\|_2 \|Nc_j\|_2} \right)^{2\alpha} \quad (33)$$

where Mr denotes row of matrix M , Nc denotes column of matrix N and α is a parameter. After constructing affinity matrix, the NCut is used to perform spectral clustering to product the clustering result. The complete subspace clustering algorithm of SLLRRC is summarized in Algorithm 2.

3.4. Computational complexity analysis

For a high dimensional space dataset $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{m \times n}$ with m attributes and n samples, the running time of subspace clustering by SLLRRC in Algorithm 2 is mainly occupied by updating J , S , Z , L and E variables in Algorithm 1.

Table 1

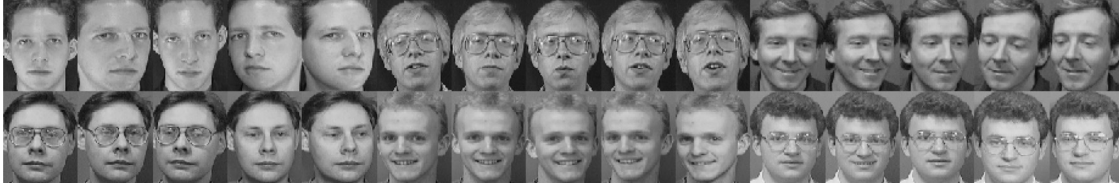
Parameter setting of different algorithms on face clustering.

Algorithm	The ORL dataset	The Extended Yale B dataset	
	Scheme 1: Divide the dataset into four groups	Scheme 1: Pick out the first 10 classes of the dataset	Scheme 2: Divide the dataset into four groups
SLLRRC	$\lambda = 5 \times 10^{-3}, \alpha = 2, \tau = 1$	$\lambda = 5 \times 10^{-3}, \alpha = 3, \tau = 1$	$\lambda = 5 \times 10^{-3}, \alpha = 2, \tau = 1$
SSC	$\lambda_e = 12/\mu_e$	$\lambda_e = 12/\mu_e$	$\lambda_e = 20/\mu_e$
LRR	$\lambda = 0.18, \alpha = 4$	$\lambda = 0.18, \alpha = 2$	$\lambda = 0.18, \alpha = 2$
LLRR	$\lambda = 5 \times 10^{-3}, \alpha = 4$	$\lambda = 5 \times 10^{-3}, \alpha = 4$	$\lambda = 5 \times 10^{-3}, \alpha = 4$
LRRSC	$\lambda = 0.1, \alpha = 3$	$\lambda = 0.2, \alpha = 4$	$\lambda = 0.1, \alpha = 3$
FLLRR	$\lambda = 5 \times 10^{-4}, \alpha = 2$	$\lambda = 5 \times 10^{-4}, \alpha = 4$	$\lambda = 5 \times 10^{-4}, \alpha = 4$

Table 2

Parameter setting of different algorithms on motion segmentation.

Algorithm	The Hopkins 155 dataset	The SegTrack dataset	
	Scheme 1: Keep the dataset in the 2F-dimensional subspace	Scheme 2: Project the dataset into 4k-dimensional subspace	Scheme 1: Pick out the two subsets of the dataset
SLLRRC	$\lambda = 3, \alpha = 4, \tau = 10$	$\lambda = 3, \alpha = 4, \tau = 10$	$\lambda = 0.05, \alpha = 3, \tau = 150$
SSC	$\lambda_z = 800/\mu_z$	$\lambda_z = 800/\mu_z$	$\lambda_z = 8/\mu_z$
LRR	$\lambda = 4, \alpha = 2$	$\lambda = 4, \alpha = 2$	$\lambda = 0.1, \alpha = 2$
LLRR	$\lambda = 1.4, \alpha = 2$	$\lambda = 1.4, \alpha = 4$	$\lambda = 0.05, \alpha = 4$
LRRSC	$\lambda = 3.3, \alpha = 2$	$\lambda = 3, \alpha = 3$	$\lambda = 0.05, \alpha = 4$
FLLRR	$\lambda = 0.05, \alpha = 2$	$\lambda = 0.08, \alpha = 2$	$\lambda = 1 \times 10^{-4}, \alpha = 2$

**Fig. 1.** The partial original sample images from the ORL dataset.

estimate, is used to measure the extent of the noise of data. The parameter α can improve the separate ability of the algorithms based on low rank representation. And the parameter τ plays the role of enhancing the sparsity of affinity matrix. From an empirical point of view, relatively larger the parameter λ , λ_e or λ_z is appropriate if the data is more clean, and the general range of parameter α is between 2 and 4. Note that the program codes of the comparison algorithms come from their respective authors, and we report the optimum results of comparison algorithms by using the parameters suggested by the respective authors or manually adjusting them to satisfactory state. The other thing to point out is that we design various schemes for dataset preprocessing according to the characteristic of different datasets to demonstrate the superiority of the proposed SLLRRC, and all experiments stop at a maximum of 500 iterations.

4.2. Face clustering

In this section, two widely used datasets are chosen for face clustering. The ORL dataset [51] contains 400 frontal face images from 40 subjects with 10 images of each, the size of each individual image is 92×112 pixel, which is available online at <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>. The Extended Yale B dataset [52,53] consists of over 2400 frontal face images from 38 subjects, the images of each individual with 168×192 pixel is shot under 9 postures and 64 lightings condition, which is available online at <http://vision.ucsd.edu/~iskwak/ExtYaleDatabase/ExtYaleB.html>. The partial original sample images from the ORL dataset and the Extended Yale B dataset are shown in Fig. 1 and Fig. 2, respectively. To ensure the efficiency of the experiments, we down sampled the images of the ORL dataset to 23×28 pixels, and each sample can be considered as 644 dimen-

Table 3

Clustering error (%) of different algorithms on scheme 1 of the ORL dataset.

Group	Algorithm					
	SLLRRC	SSC	LRR	LLRR	LRRSC	FLLRR
1	1.00	2.00	14.00	6.00	3.00	4.00
2	14.00	27.00	27.00	21.00	16.00	14.00
3	16.00	14.00	30.00	20.00	24.00	17.00
4	10.00	30.00	25.00	20.00	10.00	12.00

sional vector. The same to the Extended Yale B dataset, and each data sample is compressed to 42×48 pixel, thereby forming 2016 dimensional vector.

4.2.1. Experiments on the ORL dataset

Scheme 1: divide the dataset into four groups

The 40 subjects of the ORL dataset are evenly divided into four groups, which are subjects 1 to 10 (group 1), subjects 11 to 20 (group 2), subjects 21 to 30 (group 3) and subjects 31 to 40 (group 4). The clustering results of different algorithms on scheme 1 of the ORL dataset are listed in Table 3. As shown in the table, SSC, LRRSC and FLLRR achieves the best results in group 3, group 4 and group 2, respectively, while SLLRRC simultaneously performs best in group 1, group 2 and group 4, where SLLRRC and FLLRR tie for first place in group 2, and SLLRRC and LRRSC tie for first place in group 4. These experimental results prove that our proposed method is reasonable and feasible. In addition, LRR and LLRR are not brilliant, and there are a lot of room for progressing for them in data clustering. Fig. 3 shows the influence of the parameter λ for clustering error of SLLRRC. We investigate the effect using different values of parameter λ from $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1\}$.



Fig. 2. The partial original sample images from the Extended Yale B dataset.

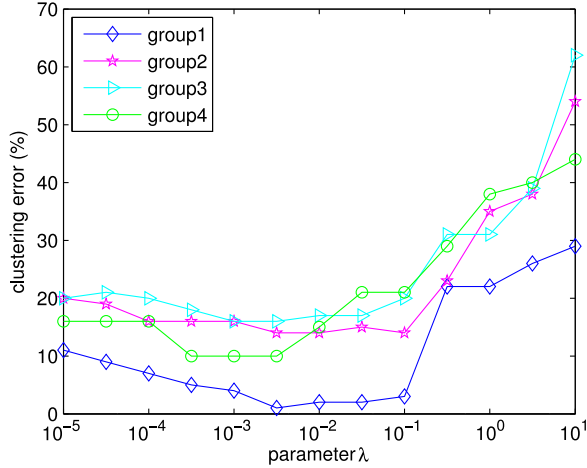


Fig. 3. The influence of the parameter λ for clustering error of SLLRRC on scheme 1 of the ORL dataset.

Table 4

Clustering error (%) of different algorithms on scheme 1 of the Extended Yale B dataset.

Dimension	Algorithm					
	SLLRRC	SSC	LRR	LLRR	LRRSC	FLLRR
100	3.91	37.81	21.09	19.84	4.38	3.91
200	3.91	39.69	17.34	9.38	26.09	14.53
300	3.91	37.19	25.31	9.22	3.91	14.06
500	3.44	44.84	21.56	19.38	3.91	14.84
2016	13.28	28.28	20.94	17.34	3.91	13.28

It is observed that the trend of the curves are almost the same, and the clustering error is relatively high when too large or too small value of λ is set, relatively low when value of λ is around 5×10^{-3} . Besides that, the clustering error varies slightly as λ ranges from 10^{-5} to 10^{-1} , which always stays within acceptable scope. The excellent reliability and robustness of SLLRRC can also be verified.

4.2.2. Experiments on the Extended Yale B dataset

There are enough and sufficient samples on this dataset, we consider two schemes to implement.

Scheme 1: pick out the first 10 classes of the dataset

The first 10 classes of the dataset are extracted as the raw data with 2016 dimensions to be studied. These data are dimensionality reduced through the principal component analysis (PCA) technology and projected into 100, 200, 300 and 500 dimensions feature space, respectively. The clustering results of different algorithms on scheme 1 of the Extended Yale B dataset are displayed in Table 4. As seen from Table 4, SLLRRC outperforms the competitors in most cases except the one which comparison with LRRSC on the 2016 dimensions data. SLLRRC is overall superior to LRRSC because that the converted nuclear norm in which provides a more sparse solution in the uniform condition of symmetric constraint. Even though on the data with 2016 dimensions, SLLRRC shows the same performance as compared with FLLRR and the lower error as compared

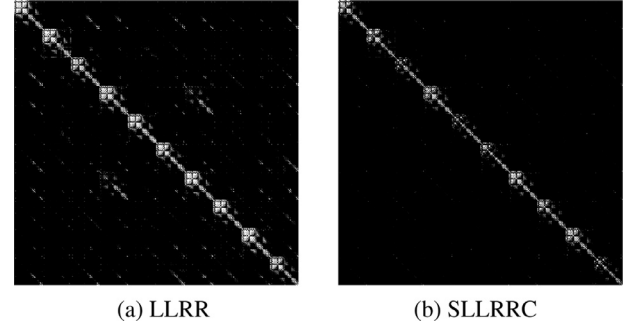


Fig. 4. The affinity matrix produced by the two algorithms on scheme 1 of the Extended Yale B dataset with 100 dimensions.

Table 5

Average clustering error (%) of different algorithms on scheme 2 of the Extended Yale B dataset with diverse subjects.

Subject	Metric	Algorithm					
		SLLRRC	SSC	LRR	LLRR	LRRSC	FLLRR
2	Mean	1.62	1.87	2.54	1.86	1.78	1.97
	Median	0.78	0	0.78	0.78	0.78	0.78
3	Mean	3.52	3.24	4.23	4.78	2.61	3.86
	Median	1.04	1.04	2.60	2.08	1.56	2.60
5	Mean	3.04	4.33	6.92	9.10	3.19	5.33
	Median	2.72	2.82	5.63	4.50	2.81	4.64
8	Mean	3.14	5.87	13.62	12.15	4.01	9.21
	Median	3.07	4.49	9.67	6.64	3.13	7.88
10	Mean	3.49	7.29	14.58	11.98	3.70	5.42
	Median	2.58	5.47	16.56	14.38	3.28	4.04

with SSC, LRR and LLRR. Fig. 4 visualizes the affinity matrix produced by two algorithms on the data with 100 dimensions. It can be seen that the block diagonal structure of the affinity matrix produced by SLLRRC is clearer, sparser and more salient than that produced by LLRR, which means each cluster becomes compact and different cluster are more easily separated. We attribute this to the fact that SLLRRC is able to learn the both low rank and sparse representation. Also, it reconstructs the dictionary both the observed and hidden data points from the same subspaces.

Scheme 2: divide the dataset into four groups

Similar to scheme 1 of the ORL dataset experiment, we also divide the 38 subjects of the Extended Yale B dataset into four groups, where subjects 1 to 10 (group 1), subjects 11 to 20 (group 2), subjects 21 to 30 (group 3) and subjects 31 to 38 (group 4) correspond to the four groups. However, the different points are that we additional trial on the first three groups of n_1 subjects and the last group of n_2 subjects on the Extended Yale B dataset, where $n_1 \in \{2, 3, 5, 8, 10\}$ and $n_2 \in \{2, 3, 5, 8\}$, instead of the all four groups of 10 subjects on the ORL dataset. As a result, there are $\{163, 416, 812, 136, 3\}$ combinations corresponding to $\{2, 3, 5, 8, 10\}$ subjects. The clustering results of different algorithms on scheme 2 of the Extended Yale B dataset are recorded in Table 5. The results illustrate that SLLRRC has the lowest mean and median clustering errors in difficult cases of 5 subjects, 8 subjects and 10 subjects, also has the ability to compete with others in the cases of

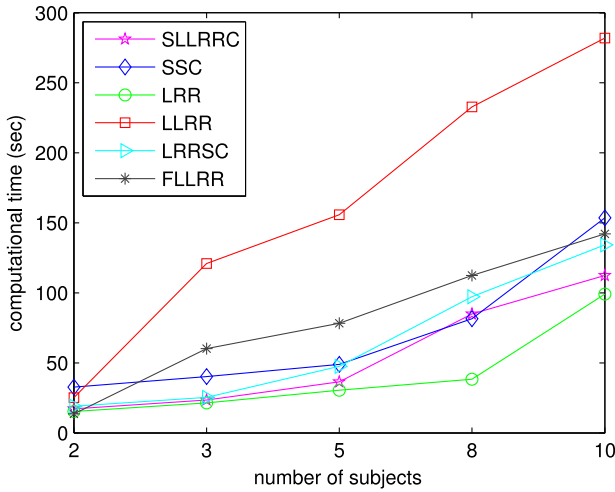


Fig. 5. Average computational time (s) of different algorithms on scheme 2 of the Extended Yale B dataset as a function of the number of subjects.

2 subjects and 3 subjects. Fig. 5 shows the average computational time of different algorithms as a function of the number of subjects just like [54,55] done. From Fig. 5, we can discover that the computational time of LLRR is drastically higher than other algorithms in general, which comes from the fact that LLRR consumes much time not only in the observed data sample, but also in the hidden data sample. SLLRRC is much more efficient than LLRR, because we have increased the learning rate ρ , which has accelerated the speed of convergence. Overall, LRR is the fastest one, SLLRRC, SSC, LRRSC and FLLRR have comparable computational times.

4.3. Motion segmentation

In this section, we also arrange two commonly used datasets for motion segmentation. The Hopkins 155 dataset [56], a well known motion segmentation database, is comprised of 120 sequences of two motions, 35 sequences of three motions and 1 sequence of five motions, a total of 156 sequences. All these sequences can be roughly divided into three categories, namely, checkerboard, traffic and non-rigid sequences, which is available online at <http://www.vision.jhu.edu/data/hopkins155>. The task of subspace clustering on such data features of the SegTrack dataset [57], in contrast, are more of difficulty and challenge. The dataset owns a variety of categories, such as cheetah, frog, hummingbird and parachute, which is available online at <https://www.cc.gatech.edu/~fli/SegTrack2/dataset.html>. Two subsets including the penguin dataset and the bmx dataset are selected as experimental materials. The former contains 6 moving objects over 42 annotated frames, while the latter contains 2 moving objects over 36 annotated frames within each video sequence. The partial original sample images from the Hopkins 155 dataset and the SegTrack dataset are shown in Fig. 6 and Fig. 7, respectively.

4.3.1. Experiments on the Hopkins 155 dataset

We design two sets of schemes for this dataset from the dimension level.

Scheme 1: keep the dataset in the 2F-dimensional subspace

The dataset is of dimension $2F \times N$, where F denotes the number of frames and N denotes the number of 2D trajectories in video. We reserve the original feature trajectories associated with each motion in an affine subspace without any preprocessing. The clustering results of different algorithms on scheme 1 of the Hopkins 155 dataset are listed in Table 6. We can see that SLLRRC performs the best in all of the max error, mean error and standard deviation

Table 6

Average clustering error (%) of different algorithms on scheme 1 of the Hopkins 155 dataset with 156 sequences.

Metric	Algorithm					
	SLLRRC	SSC	LRR	LLRR	LRRSC	FLLRR
Max.	19.34	46.97	33.33	26.26	33.33	21.18
Mean	0.77	2.65	1.71	1.07	1.50	0.85
Std.	2.58	7.61	4.86	3.25	4.36	3.08

Table 7

Average clustering error (%) of different algorithms on scheme 2 of the Hopkins 155 dataset with 156 sequences.

Metric	Algorithm					
	SLLRRC	SSC	LRR	LLRR	LRRSC	FLLRR
Max	39.69	45.50	43.38	43.38	43.38	41.42
Mean	0.86	2.65	2.17	2.12	1.56	1.24
Std.	3.73	7.53	6.58	6.15	5.48	4.62

Table 8

Clustering error (%) of different algorithms on scheme 1 of the SegTrack dataset with 2 subsets.

Subset	Number	Algorithm					
		SLLRRC	SSC	LRR	LLRR	LRRSC	FLLRR
penguin	50	6.00	44.67	28.00	11.33	8.00	7.67
	100	8.67	42.00	20.33	16.50	9.00	9.50
	150	6.67	41.11	19.56	11.11	8.22	54.44
	200	8.67	45.44	19.33	12.92	9.17	41.75
bmx	50	4.83	38.92	19.18	10.07	5.66	9.47
	100	4.79	35.07	22.18	13.19	6.43	11.00
	150	5.25	35.54	23.56	14.62	6.69	21.08
	200	6.02	37.70	23.79	11.78	7.11	25.83

followed by FLLRR, LLRR, LRRSC, LRR and SSC in turn. Fig. 8 shows the anova test of different algorithms on the Hopkins 155 dataset. In Fig. 8, SLLRRC and FLLRR are the algorithms with high degree of stability.

Scheme 2: project the dataset into 4k-dimensional subspace

With the PCA dimensionality reduction technology, the raw data will be projected into 4k-dimensional subspace, where k denotes the number of subspaces. The clustering results of different algorithms on scheme 2 of the Hopkins 155 dataset are displayed in Table 7. We can see that SLLRRC gets again the best results in terms of all these metrics including the max error, mean error and standard deviation. LRR-based methods, which use the global structure of data as dictionary to find the corresponding representation, are generally better than SSC method in this situation. Fig. 9 shows the anova test of algorithms on the Hopkins 155 dataset. In Fig. 9, the stability of SLLRRC is in really high degree, which proves the robustness of our SLLRRC again.

4.3.2. Experiments on the SegTrack dataset

Scheme 1: pick out the two subsets of the dataset

The bmx subset and the penguin subset are the two most representative units of the whole dataset. We sample four distinct numbers of points of feature trajectories from the two subsets, namely, 50, 100, 150 and 200. The clustering results of different algorithms on scheme 1 of the SegTrack dataset are recorded in Table 8. From Table 8, SLLRRC gets the lowest clustering error across all experiments that can be found. In particular, SLLRRC acquires clustering error rates of 6.00%, 8.67%, 6.67% and 8.67% in the penguin subset, 4.83%, 4.79%, 5.25% and 6.02% in the bmx subset, representing 5.33%, 7.83%, 4.44%, 4.25%, 5.24%, 8.40%, 9.37% and 5.76% improvement, respectively, over the accuracy of LLRR. This indicates that SLLRRC achieves encouraging results in improving LLRR. Further, these results confirm that it is promising to



Fig. 6. The partial original sample images from the Hopkins 155 dataset.

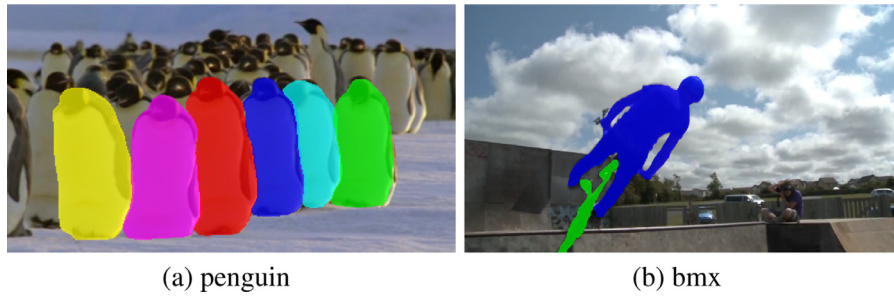


Fig. 7. The partial original sample images from the SegTrack dataset.

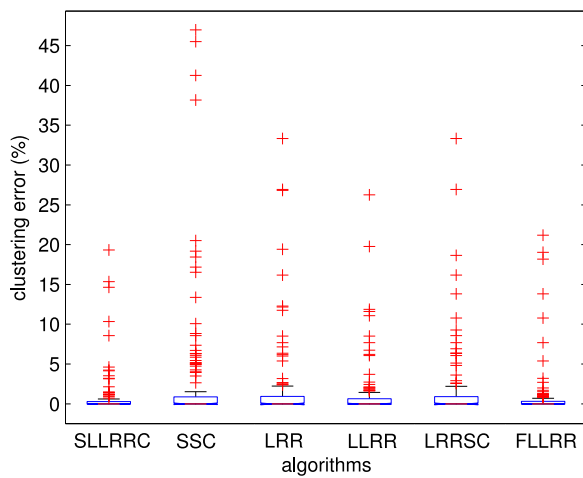


Fig. 8. The anova test of different algorithms on scheme 1 of the Hopkins 155 dataset.

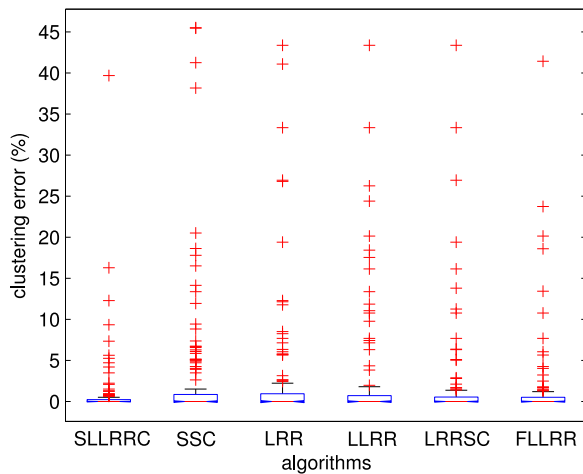
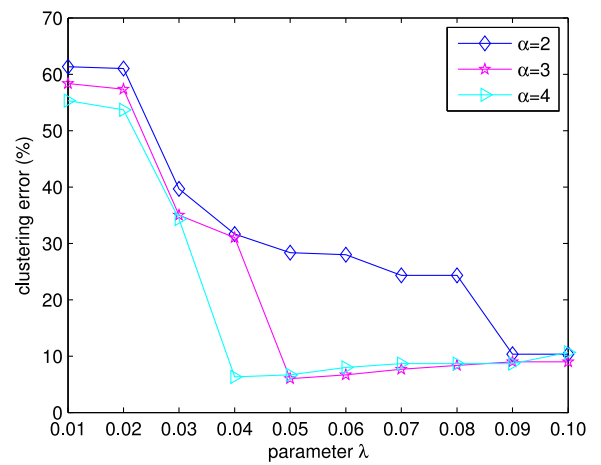
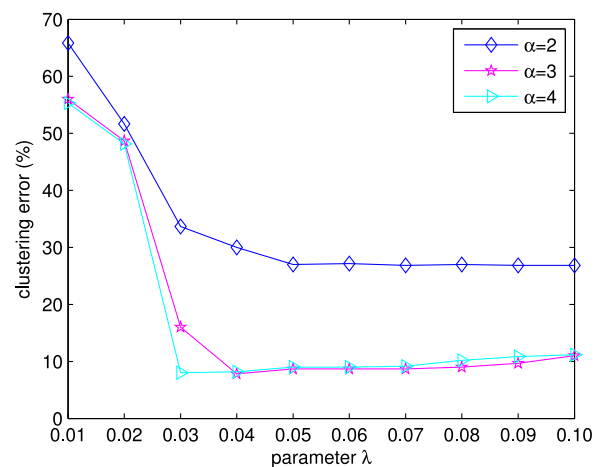


Fig. 9. The anova test of different algorithms on scheme 2 of the Hopkins 155 dataset.



(a) number=50



(b) number=100

Fig. 10. Clustering error given different λ and α combinations of SLLRRC on scheme 1 of the SegTrack dataset with different numbers of feature trajectories of the penguin subset.

incorporate the converted nuclear norm and symmetric constraint into LLRR technique. Fig. 10 shows the clustering error given different λ and α combinations of SLLRRC with different numbers of feature trajectories of the penguin subset. It is easily discovered that the curves shakes so badly when set $\alpha=2$, especially in Fig. 10(a). When set $\alpha=3$ or $\alpha=4$, the curves is smoother, which is more benefit to robust subspace clustering, and SLLRRC performs well for a large range of λ .

5. Conclusions

A novel low rank representation subspaces clustering algorithm called SLLRRC is proposed in this paper. SLLRRC extends the original LLRR algorithm, which addresses the problem of insufficient sampling of the dictionary. Thus, both the observed and hidden data sample are employed for building the dictionary. On the other hand, it incorporates the converted nuclear norm and symmetric constraint to attempt to learn the both low rank and sparse coefficient matrix with weight consistency for each pair of data samples, which can be acquired by IALM. That is for the purpose of constructing desired affinity matrix to robust subspace clustering. We conduct considerable amount of experiments such as face clustering and motion segmentation to illustrate that our method is of strong competitiveness compared to these state-of-the-art subspace clustering algorithms. There are, however, several underlying unsolved problems. For example, parameters in SLLRRC are still set by empirical estimation accumulated from a large number of experiments although the algorithm is robust, and developing methods to automatically get these optimal parameters is very meaningful. In addition, we intend to apply the algorithm to clustering analysis on single-cell RNA-seq.

Compliance with ethical standards

Conflict of interest The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable and insightful comments. This work is partially supported by the National Natural Science Foundation of China under Grants 61170110 and 61772027, and the Zhejiang Provincial Natural Science Foundation of China under Grant LY13F020043.

References

- [1] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (5) (2002) 603–619.
- [2] A. Rodriguez, A. Laio, Clustering by fast search and find of density peaks, *Science* 344 (6191) (2014) 1492–1496.
- [3] R. Mehmood, G. Zhang, R. Bie, H. Dawood, H. Ahmad, Clustering by fast search and find of density peaks via heat diffusion, *Neurocomputing* 208 (2016) 210–217.
- [4] S. Yue, M. Wei, J.S. Wang, H. Wang, A general grid-clustering approach, *Pattern Recognit. Lett.* 29 (9) (2008) 1372–1384.
- [5] G.A. Carpenter, S. Grossberg, A massively parallel architecture for a self-organizing neural pattern recognition machine, *Comput. Vis. Graph. Image. Process.* 37 (1) (1987) 54–115.
- [6] G.A. Carpenter, S. Grossberg, The ART of adaptive pattern recognition by a self-organizing neural network, *Computer* 21 (3) (1988) 77–88.
- [7] T. Kohonen, The self-organizing map, *Neurocomputing* 21 (1998) 1–6.
- [8] L. Zhuang, H. Gao, Z. Lin, Y. Ma, Non-negative low rank and sparse graph for semi-supervised learning, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, 2012, pp. 2328–2335.
- [9] B. McWilliams, G. Montana, Subspace clustering of high-dimensional data: a predictive approach, *Data Min. Knowl. Discov.* 28 (3) (2014) 736–772.
- [10] L. Zappella, X. Lladó, E. Provenzi, J. Salvi, Enhanced local subspace affinity for feature-based motion segmentation, *Pattern Recognit.* 44 (2) (2011) 454–470.
- [11] L. Lu, R. Vidal, Combined central and subspace clustering for computer vision applications, in: *Proceedings of the International Conference on Machine Learning, (ICML)*, 2006, pp. 593–600.
- [12] C.W. Gear, Multibody grouping from motion images, *Int. J. Comput. Vis.* 29 (2) (1998) 133–150.
- [13] R. Vidal, Y. Ma, S. Sastry, Generalized principal component analysis (GPCA), *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (12) (2005) 1945–1959.
- [14] P.S. Bradley, O.L. Mangasarian, K-plane clustering, *J. Glob. Optim.* 16 (1) (2000) 23–32.
- [15] J. Ho, M.H. Yang, J. Lim, K.C. Lee, D. Kriegman, Clustering appearances of objects under varying illumination conditions, in: *Proceedings of the IEEE International Conference on Computer Vision, (ICCV)*, 2003, pp. 11–18.
- [16] M.E. Tipping, C.M. Bishop, Mixtures of probabilistic principal component analyzers, *Neural Comput.* 11 (2) (1999) 443–482.
- [17] Y. Ma, H. Derksen, W. Hong, J. Wright, Segmentation of multivariate mixed data via lossy data coding and compression, *IEEE Trans. Pattern Anal. Mach. Intell.* 29 (9) (2007) 1546–1562.
- [18] T.E. Boult, L.G. Brown, Factorization-based segmentation of motions, in: *Proceedings of the IEEE Workshop on Visual Motion*, 1991, pp. 179–186.
- [19] U.V. Luxburg, A tutorial on spectral clustering, *Stat. Comput.* 17 (4) (2007) 395–416.
- [20] F. Lauer, C. Schnörr, Spectral clustering of linear subspaces for motion segmentation, in: *Proceedings of the IEEE International Conference on Computer Vision, (ICCV)*, 2009, pp. 678–685.
- [21] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (11) (2013) 2765–2781.
- [22] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Robust recovery of subspace structures by low-rank representation, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (1) (2013) 171–184.
- [23] S. J. J. Malik, Normalized cuts and image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (8) (2000) 888–905.
- [24] D.S. Pham, S. Budhaditya, D. Phung, S. Venkatesh, Improved subspace clustering via exploitation of spatial constraints, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, 2012, pp. 550–557.
- [25] B. Saha, D.S. Pham, D. Phung, S. Venkatesh, Sparse subspace clustering via group sparse coding, in: *Proceedings of the SIAM International Conference on Data Mining, (SDM)*, 2013, pp. 130–138.
- [26] Y. Chen, G. Li, Y. Gu, Active orthogonal matching pursuit for sparse subspace clustering, *IEEE Signal Process. Lett.* 25 (2) (2018) 164–168.
- [27] G. Liu, S. Yan, Latent low-rank representation for subspace segmentation and feature extraction, in: *Proceedings of the IEEE International Conference on Computer Vision, (ICCV)*, 2011, pp. 1615–1622.
- [28] J. Feng, Z. Lin, H. Xu, S. Yan, Robust subspace segmentation with block-diagonal prior, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, 2014, pp. 3818–3825.
- [29] J. Wang, X. Wang, F. Tian, C. Liu, H. Yu, Constrained low-rank representation for robust subspace clustering, *IEEE Trans. Cybern.* 47 (12) (2017) 4534–4546.
- [30] J. Chen, H. Mao, Y. Sang, Z. Yi, Subspace clustering using a symmetric low-rank representation, *Knowl.-Based Syst.* 127 (2017) 46–57.
- [31] J. Chen, H. Zhang, H. Mao, Y. Sang, Z. Yi, Symmetric low-rank representation for subspace clustering, *Neurocomputing* 173 (2016) 1192–1202.
- [32] X. Fang, Y. Xu, X. Li, Z. Lai, W. Wong, Robust semi-supervised subspace clustering via non-negative low-rank representation, *IEEE Trans. Cybern.* 46 (8) (2016) 1828–1838.
- [33] Z. Kang, C. Peng, Q. Cheng, Robust subspace clustering via tighter rank approximation, in: *Proceedings of the ACM International Conference on Information and Knowledge Management, (CIKM)*, 2015, pp. 393–401.
- [34] M. Fazel, Matrix rank minimization with applications, Stanford University, 2002 (Ph.D. thesis).
- [35] K. Mohan, M. Fazel, Iterative reweighted algorithms for matrix rank minimization, *J. Mach. Learn. Res.* 13 (11) (2012) 3441–3473.
- [36] Y. Hu, D. Zhang, J. Ye, X. Li, X. He, Fast and accurate matrix completion via truncated nuclear norm regularization, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (9) (2013) 2117–2130.
- [37] E.J. Candès, T. Tao, The power of convex relaxation: near-optimal matrix completion, *IEEE Trans. Inf. Theory* 56 (5) (2010) 2053–2080.
- [38] Z. Lin, R. Liu, Z. Su, Linearized alternating direction method with adaptive penalty for low-rank representation, in: *Proceedings of the Advances in Neural Information Processing Systems, (NIPS)*, 2011, pp. 612–620.
- [39] Y. Ni, J. Sun, X. Yuan, S. Yan, L. Cheong, Robust low-rank subspace segmentation with semidefinite guarantees, in: *Proceedings of the IEEE International Conference on Data Mining Workshops, (ICDMW)*, 2010, pp. 1179–1188.
- [40] M. Nikolova, Local strong homogeneity of a regularized estimator, *SIAM J. Appl. Math.* 61 (2) (2000) 633–658.
- [41] J. Lv, Y. Fan, A unified approach to model selection and sparse recovery using regularized least squares, *Ann. Stat.* 37 (6A) (2009) 3498–3528.
- [42] M. Fornasier, H. Rauhut, Iterative thresholding algorithms, *Appl. Comput. Harmon. Anal.* 25 (2) (2008) 187–208.
- [43] Z. Lin, M. Chen, L. Wu, Y. Ma, The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices, in: *University of Illinois at Urbana-Champaign (UIUC) Technical Report UILU-ENG-09-2215*, 2009.
- [44] Z. Lin, A. Ganes, J. Wright, L. Wu, M. Chen, Y. Ma, Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix, *J. Mar. Biol. Assoc. U. K.* 56 (3) (2009) 707–722.
- [45] S. Du, Y. Ma, Y. Ma, Graph regularized compact low rank representation for subspace clustering, *Knowl.-Based Syst.* 118 (2017) 56–69.

- [46] W. He, J.X. Chen, W. Zhang, Low-rank representation with graph regularization for subspace clustering, *Soft Comput.* 21 (6) (2017) 1569–1581.
- [47] J.F. Cai, E.J. Candès, Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Optim.* 20 (4) (2010) 1956–1982.
- [48] J. Yang, W. Yin, Y. Zhang, Y. Wang, A fast algorithm for edge-preserving variational multichannel image restoration, *SIAM J. Imag. Sci.* 2 (2) (2009) 569–592.
- [49] Y. Song, Y. Wu, Subspace clustering based on latent low rank representation with Frobenius norm minimization, *Neurocomputing*. 275 (2018) 2479–2489.
- [50] C. Peng, Z. Kang, Q. Cheng, Integrating feature and graph learning with low-rank representation, *Neurocomputing*. 249 (2017) 106–116.
- [51] F.S. Samaria, A.C. Harter, Parameterisation of a stochastic model for human face identification, in: *Proceedings of the IEEE Workshop on Applications of Computer Vision, (WACV)*, 1994, pp. 138–142.
- [52] A.S. Georghiades, P.N. Belhumeur, D.J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, *IEEE Trans. Pattern Anal. Mach. Intell.* 23 (6) (2001) 643–660.
- [53] K.-C. Lee, J. Ho, D.J. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (5) (2005) 684–698.
- [54] X.J. Zhang, C. Xu, X.L. Sun, G. Baciú, Schatten-q regularizer constrained low rank subspace clustering model, *Neurocomputing* 182 (2015) 36–47.
- [55] H.Z. Chen, W.W. Wang, X.C. Feng, R.Q. He, Discriminative and coherent subspace clustering, *Neurocomputing* 284 (2018) 177–186.
- [56] R. Tron, R. Vidal, A benchmark for the comparison of 3-D motion segmentation algorithms, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, 2007, pp. 1–8.
- [57] F. Li, T. Kim, A. Humayun, D. Tsai, J.M. Rehg, Video segmentation by tracking many figure-ground segments, in: *Proceedings of the IEEE International Conference on Computer Vision, (ICCV)*, 2013, pp. 2192–2199.



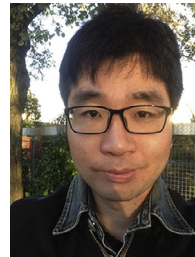
Xian Fang received the B.S. degree from the Anhui Xinhua University, Hefei, China, in 2016. Now, he is pursuing the M.S. degree in the School of Information Science and Technology, Zhejiang Sci-Tech University, China. His current research interests include machine learning and bioinformatics.



Zhixin Tie received his Ph.D. degree in computer science from Zhejiang University, China, in 2000. He is currently an Associate Professor in the School of Information Science and Technology at Zhejiang Sci-Tech University (ZSTU), China. His current research interests include mobile agent systems, embedded systems, data mining and power automation systems.



Feiyang Song received the B.S. degree from the Anhui University of Science and Technology, Huainan, China, in 2016. Now, he is pursuing the M.S. degree in the School of Information Science and Technology, Zhejiang Sci-Tech University, China. His current research interests include machine learning and data mining.



Jialiang Yang received his Ph.D. degree in the Department of Mathematics, National University of Singapore in 2009. From 2010 to 2011, he was an assistant professor in CAS-MPG Partner Institute for Computational Biology in China. After that, he moved to USA and became a postdoctoral fellow at the Department of Basic Sciences, Mississippi State University. Since 2013, he has become a postdoctoral fellow and senior scientist at the Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai. Dr. Yang has published more than 50 peer-reviewed articles. His main research areas include bioinformatics, machine learning, aging and evolutionary biology.