Noise-Aware Subspace Clustering

Baohua Li, Huchuan Lu, Ying Zhang, Zhouchen Lin

***Abstract*—The existing popular clustering self-reconstruction based models generally assume the corruption stems from a unimodal Gaussian noise, which makes the model insufficiently describe the noise. In fact, the realistic noise is much more complex beyond our assumption. Therefore, it may be unsuitable to consider the noises come from one source. Motivated by this, we propose the Mixture of Gaussian Regression (MoG Regression) for subspace clustering. The MoG Regression can provide an effective way to model the unknown noise distribution which approaches the real one as far as possible. As a result, the desired affinity matrix is better at characterizing the structure of data in real world, furthermore, improving the performance. We prove the diagonal entries of coefficient matrix are all zeros and our method holds the grouping effect. Borrowing the ideal from the minimum message length, a model selection strategy is proposed to estimate the numbers of the Gaussian components. In addition, we proved the asymptotic property of our model. In order to highlight the ability of modelling the complex noises we evaluate the MoG Regression model for clustering on the challenge datasets. The experimental results on multiple datasets demonstrate that the proposed MoG Regression model significantly outperforms state-of-the-art subspace clustering methods.**

***Index Terms*—Subspace clustering, Mixture of Gaussian Regression, Minimum Message Length.**

I. INTRODUCTION

Subspace clustering aims to model the data as samples drawn from a union of multiple low-dimensional affine subspaces. In a nut shell, the mission of subspace clustering is to gather the given data points into disparate group which contains the data points that come from the same underlying subspace. It has been attracting more and more attention in recent years and has found many applications in computer vision and image processing, such as image segmentation [1], motion segmentation [2], face clustering [3], and image representation and compression [4].

There exists numbers of subspace clustering approaches which have been proposed in the past two decades. These methods may be roughly divided into four main categories: algebraic methods [5]–[7], iterative methods [8], [9], statistical methods [10]–[12], and spectral-clustering-based methods [13]–[19]. It should be noted specially that the subspace clustering self-reconstruction based methods [15]–[19], which take root in the elegant spectral graph theory [20], have shown excellent performance in many real applications.

Generally, the subspace clustering self-reconstruction based methods consist of two steps. Firstly, building an affinity matrix which is used to capture the similarity between pairs of sample points. Secondly, graph cut is applied to a undirected graph, whose vertices are the samples and whose weights are prescribed by the affinity matrix, for segmenting the sample points. Building a "good" affinity matrix is key to guarantee a good clustering result. Therefore, some subspace clustering methods focus on how to build a good affinity matrix.

Based on the fact that each datum point in a union of several subspaces can be represented as a linear or affine combination of other points, the Sparse Subspace Clustering (SSC) algorithm [15] utilizes the ℓ_1-norm regularization to find the sparsest representation of a data point, where points come from the same subspace correspond to the nonzero representation coefficients. Low-Rank Representation (LRR) [16] aims to get a low rank reconstruction coefficient for robust subspace recovery of the data containing corruptions, the $\ell_{2,1}$ is used to make the algorithm more robust to outliers. Least Squares Regression (LSR) [17] employs the Frobenius norm regularization to speed up the clustering process, while still ensuring the grouping effect of the representation matrix. However, the reconstruction coefficient of SSC may be too sparse to encode the data correlation, while the reconstruction coefficient derived by both LRR and LSR may result in dense connections between-clusters besides the within-clusters. In order to achieve a good balance between within-cluster density (which we call *grouping effect* afterwards) and between-cluster sparsity, Correlation Adaptive Subspace Segmentation (CASS) [19] adopts trace Lasso norm regularization, which is adaptive to the data correlation, to trade-off the representation matrix.

As pointed out by Liu et al. [16], the noises that always exist in data can perturb the subspace structures, which leads to unreliable subspace clustering result. To cluster the real subspaces when the data are corrupted by noises, SSC, LRR, LSR, and CASS employ different norms to select the solution of various properties, respectively.

Given a data matrix $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N) \in \mathbb{R}^{M\times N}$ with N samples in $\mathbb{R}^M$, here, we denote $\boldsymbol{E} \in \mathbb{R}^{M\times N}$ and $\boldsymbol{Z} \in \mathbb{R}^{N\times N}$ as the noise matrix and the representation matrix, respectively, where the component Z_{ij} of $\boldsymbol{Z}$ measures the similarity between points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the data matrix. In this paper, we use $\|\cdot\|_F$, $\|\cdot\|_1$, $\|\cdot\|_2$, $\|\cdot\|_{2,1}$, and $\|\cdot\|_*$ to denote Frobenius norm, the ℓ_1-norm (sum of absolute values), the ℓ_2-norm, the $\ell_{2,1}$-norm (sum of the ℓ_2-norm of columns of a matrix), and the nuclear norm (sum of singular values), respectively. The mathematical models of mentioned subspace clustering methods are listed as follows.

Sparse Subspace Clustering (SSC) [15]:

$$\min_{\boldsymbol{Z},\boldsymbol{E}} \| \boldsymbol{E} \|_1 + \lambda \| \boldsymbol{Z} \|_1 \\ s.t.\ \boldsymbol{X} = \boldsymbol{X}\boldsymbol{Z} + \boldsymbol{E}, \operatorname{diag}(\boldsymbol{Z}) = \boldsymbol{0}.$$

Low-Rank Representation (LRR) [16]:

$$\min_{\boldsymbol{Z},\boldsymbol{E}} \| \boldsymbol{E} \|_{2,1} + \lambda \| \boldsymbol{Z} \|_* \\ s.t.\ \boldsymbol{X} = \boldsymbol{X}\boldsymbol{Z} + \boldsymbol{E}.$$

Least Squares Regression (LSR) [17]:

$$\min_{\boldsymbol{Z},\boldsymbol{E}} \parallel \boldsymbol{E} \parallel_F^2 + \lambda \parallel \boldsymbol{Z} \parallel_F^2 \quad s.t.\ \boldsymbol{X} = \boldsymbol{X}\boldsymbol{Z} + \boldsymbol{E}, \mathrm{diag}\left(\boldsymbol{Z}\right) = \boldsymbol{0}.$$

Correlation Adaptive Subspace Segmentation (CASS) [19]:

$$\min_{\boldsymbol{Z},\boldsymbol{E}} \parallel \boldsymbol{E} \parallel_F^2 + \lambda \sum_{n=1}^{N} \parallel \boldsymbol{X}\mathrm{diag}\left(\boldsymbol{z}_i\right) \parallel_* \quad s.t.\ \boldsymbol{X} = \boldsymbol{X}\boldsymbol{Z} + \boldsymbol{E}.$$

In the above mentioned formulations, $\boldsymbol{z}_i$ is represented the i-th column of $\boldsymbol{Z}$, $\mathrm{diag}\left(\boldsymbol{z}_i\right)$ is a diagonal matrix with entries of $\boldsymbol{z}_i$ on its diagonal, and $\lambda > 0$ is a regularization parameter to balance the effects of two terms. $\boldsymbol{E}$ denotes the reconstruction error. $\parallel \boldsymbol{E} \parallel_F^2$ is utilized to model Gaussian noise, $\parallel \boldsymbol{E} \parallel_{2,1}$ is for sample-specific corruptions, and $\parallel \boldsymbol{E} \parallel_1$ is for entry-wise corruptions.

All the clustering algorithms mentioned above rely on specific norms on $\boldsymbol{Z}$ and $\boldsymbol{E}$ to encourage either the between-cluster sparsity and within-cluster density or grouping effect of the representation matrix which makes the model be valid. However, they all use a relatively simple norm for the data fidelity term to describe the noises. If the data are corrupted by noise, then the subspace structures, the grouping effect, and the data similarity are all to be corrupted as well. In fact, the real noise scenario in applications often exhibits very complex statistical distributions, rather than simply being Gaussian or Laplace [21]. If the noise be described by a simple norm like the Frobenious norm, ℓ_1-norm, or $\ell_{2,1}$-norm, the obtained affinity matrix may largely deviate from the desired one. Therefore, how to properly model the noise for subspace clustering problem is a crucial issue.

To alleviate this problem, we borrow a fundamental result from the probability theory that almost any distribution can be well approximated by a mixture of a suitable number of Gaussian type distributions. Namely, we employ the mixture of Gaussian (MoG) model to describe the real noise accurately, rather than assuming some specific distribution for the noise. As for the regularization the term, we simply choose the Frobenius norm which means that we select the minimal Frobenius norm solution among the candidates. The reasons are two-fold. First, we want to demonstrate the effect of noise modeling on subspace clustering. So a simple regularization on $\boldsymbol{Z}$ can better exhibit such an effect. Second, it makes the computational procedure much easier with the Frobenious norm on $\boldsymbol{Z}$. For example, we can use the traditional Expectation Maximization (EM) algorithm to find the solution from our new subspace clustering model. We prove that the prosed model holds the grouping effect [22] for correlated data points, which encourages the coefficients of correlated data pints are approximately equal. How to determine the number of Gaussians K is anther crux problem. Aside from empirically fixing K, we proposed a model selection strategy to estimate K inspired by [23], [24]. Besides, we prove the asymptotic properties of our model for fixed M and K in the sprit of [25].

In summery, we list the outline of the contributions as follows:

- We present a novel subspace clustering approach called Mixture of Gaussian Regression (MoG Regression), which is used to find a good affinity matrix.
- We prove that MoG Regression has the grouping effect, which is important for subspace clustering.
- We provide a model selection method based on the minimum message length(MML) criterion to estimate the numbers of Gaussian components.
- The asmptotic properties of our model is shown.

The remainder of the paper is organized as follows. In Section II, we motivate and introduce the MoG Regression method in detail for clustering data. In Section III we prove that the proposed model possesses the grouping effect. The asymptotic property is shown in section IV. Based on MML, we show how to estimate K in section V. Section VI provides the experimental results on motion segmentation, hand-written digits clustering, and complex face clustering to evaluate and demonstrate the superiority of MoG Regression. We relegate the main steps of proof to section VIII. Partial results of this paper appear in our conference version [26].

II. Subspace Clustering via MoG Regression

As described in [27], we model the subspace clustering issue as the following optimization problem:

$$\begin{aligned} &\min_{\boldsymbol{Z},\boldsymbol{E}} \mathcal{L}\left(\boldsymbol{E}\right) + \mathcal{R}\left(\boldsymbol{Z}\right) \\ &s.t.\ \boldsymbol{X} = \boldsymbol{X}\boldsymbol{Z} + \boldsymbol{E}, \end{aligned} \tag{1}$$

where $\mathcal{L}\left(\boldsymbol{E}\right)$ is to be described the noise in the loss function, and $\mathcal{R}\left(\boldsymbol{Z}\right)$ is the regularization term to impose some desired properties on the representation matrix $\boldsymbol{Z}$.

In fact, the noise is a nuisance, that may spoil the ability of (1) to cluster the real subspaces. So, it becomes significant importance that how to describe noise in subspace clustering problems when we face the unknown complex noise. Lu et al. [27] proposed Correntropy Induced L2 (CIL2) graph, which uses correntropy to process non-Gaussian and impulsive noise for robust subspace clustering, and the effectiveness is demonstrated by experiments of face clustering under various types of corruptions and occlusions. In fact, the variation of the width of kernel function makes the behavior of Correntropy Induced Metric changes between ℓ_0, ℓ_1, and ℓ_2 norms, which is effective for many types of noise but not for general noise anyway. However, as Liu et al. [28] pointed that the correntropy strategy is more suitable for the impulsive noise environment, which may cripple the performance of clustering by correntropy induced metric [27] if the noise is out of specified type.

Inspired by the probability theory that almost any continuous density can be approximated by using s sufficient number of Gaussians to arbitrary accuracy. We therefore propose a novel clustering method called MoG Regression, which employs MoG to characterize the general noise for robust subspace clustering.

We assume that each column $\boldsymbol{e}_n\left(n = 1, \ldots, N\right)$ of $\boldsymbol{E}$ follows the MoG distribution, i.e.,

$$p\left(\boldsymbol{e}_n\right) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{e}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right),$$

where K is the number of Gaussian components and π_k denotes the mixing weight which is satisfy with the constrain $\pi_k \geq 0$ and $\sum_{k=1}^{K} \pi_k = 1$. $\mathcal{N}\left(\boldsymbol{e}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)$ is denoted the zero-mean multivariate Gaussian distribution, with $\boldsymbol{\Sigma}_k (k = 1, 2, \ldots, K)$ representing the invertible and symmetrical covariance matrix.

It is analogous to the classical regression analysis that the columns of $\boldsymbol{E}$ are assumed to be independently and identically distributed in the MoG Regression setting. Thus we have

$$p\left(\boldsymbol{E}\right) = \prod_{n=1}^{N} \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{e}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right). \quad (2)$$

In the general MoG model, our mission is to find $\boldsymbol{\pi} = \left(\pi_1, ..., \pi_K\right)^{\top}$ and $\boldsymbol{\Sigma} = \left(\boldsymbol{\Sigma}_1, ..., \boldsymbol{\Sigma}_K\right)$ that maximize $p\left(\boldsymbol{E}\right)$, which is also equivalent to minimizing the negative log likelihood function that is defined as

$$-\ln p\left(\boldsymbol{E}\right) = -\sum_{n=1}^{N} \ln \left(\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{e}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)\right). \quad (3)$$

If we use $\mathcal{L}\left(\boldsymbol{E}\right) = -\ln p\left(\boldsymbol{E}\right)$ to replace the Frobenius norm that is related to the reconstruction error term in the LSR model, then the proposed MoG Regression method can be formulated as follows:

$$\begin{aligned} \min_{\boldsymbol{Z}, \boldsymbol{E}, \boldsymbol{\pi}, \boldsymbol{\Sigma}} \; & -\sum_{n=1}^{N} \ln \left(\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{e}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)\right) + \lambda \parallel \boldsymbol{Z} \parallel_F^2 \\ & s.t. \; \boldsymbol{X} = \boldsymbol{X}\boldsymbol{Z} + \boldsymbol{E}, \operatorname{diag}\left(\boldsymbol{Z}\right) = \mathbf{0}, \\ & \pi_k \geq 0, \boldsymbol{\Sigma}_k \in \mathbb{S}^{+}, k = 1, ..., K, \sum_{k=1}^{K} \pi_k = 1, \end{aligned} \quad (4)$$

where $\lambda > 0$ is the regularization parameter, $\mathbb{S}^{+}$ is denoted the set of symmetrical positive definite (SPD) matrices and the constraint $\operatorname{diag}\left(\boldsymbol{Z}\right) = \mathbf{0}$ discourages using a sample to represent itself. Here we simply choose the Frobenius norm of $\boldsymbol{Z}$ as the regularization term. As declared before, we chose the Frobenious norm on $\boldsymbol{Z}$ that can not only reduce the computation cost but also expose the the effect of MoG regression based noise modeling on subspace clustering.

A natural way to capture the solution of (4) may be the powerful EM algorithm [29], [30], which finds the maximum-likelihood estimate of the parameters iteratively. Its procedure starts from an initial guess and iteratively runs an expectation (E) step, which evaluates the posterior probabilities using currently known parameters, and a maximization (M) step, which will re-estimate the parameters based on the probabilities calculated in the E step. The iterations will stop until some convergence criteria are satisfied [31]–[33]. Integrating the traditional processes of the EM algorithm, we can obtain the solution of problem (4) in the following three main steps.

First, we initialize the representation matrix $\boldsymbol{Z}$, mixing weighting π_k, and covariance matrices $\boldsymbol{\Sigma}_k$, for $k = 1, ..., K$.

In the E-step, we compute the posterior probabilities based on the current parameters:

$$\gamma_{n,k} = \sum_{n=1}^{N} \frac{\pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_j\right)},$$

where $\widetilde{\boldsymbol{e}}_n = \widetilde{\boldsymbol{X}}_n \boldsymbol{z}_n - \boldsymbol{x}_n$ and $\widetilde{\boldsymbol{X}}_n$ is a copy of $\boldsymbol{X}$ except that the n-th column is $\mathbf{0}$.

In the M-step, we want to minimize the log likelihood with respect to the parameters, using the current posterior probabilities.

To find $\boldsymbol{\Sigma}_k, \; k = 1, 2, \ldots, K$, we should solve the following optimization problem

$$\begin{aligned} \min_{\boldsymbol{\Sigma}_k} - \sum_{n=1}^{N} \ln \left(\textstyle\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)\right) \\ s.t. \; \boldsymbol{\Sigma}_k \in \mathbb{S}^{+}. \end{aligned}$$

Letting the derivative of the objective function with respect to $\boldsymbol{\Sigma}_k$ to be zero, we obtain

$$\boldsymbol{\Sigma}_k = \frac{1}{\gamma_{n,k}} \left(\sum_{n=1}^{N} \frac{\pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_j\right)} \widetilde{\boldsymbol{e}}_n \cdot \widetilde{\boldsymbol{e}}_n^{\top} + \epsilon \boldsymbol{I}\right), \quad (5)$$

where $\epsilon > 0$ is a small regularization parameter to avoid that the determinant of $\boldsymbol{\Sigma}_k$ equals to zero.

Each mixing weighting $\pi_k, \; k = 1, 2, \ldots, K$, is updated by solving

$$\min_{\pi_k \geq 0} - \sum_{n=1}^{N} \ln \left(\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)\right) + \beta \left(\sum_{k=1}^{K} \pi_k - 1\right),$$

where $\beta > 0$ is the Lagrangian multiplier. We find $\beta = N$ and accordingly

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} \frac{\pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_j\right)}. \quad (6)$$

Each column of $\boldsymbol{Z}$ is found by solving the following problem:

$$\min_{\boldsymbol{z}_n} - \sum_{n=1}^{N} \ln \left(\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right)\right) + \lambda \parallel \boldsymbol{z}_n \parallel_F^2 . \quad (7)$$

By setting the derivative of above object function with respect to $\boldsymbol{z}_n$ to zero, we obtain

$$\boldsymbol{z}_n = \left(\frac{\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right) \widetilde{\boldsymbol{X}}_n^{\top} \boldsymbol{\Sigma}_k^{-1} \widetilde{\boldsymbol{X}}_n}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_j\right)} + 2\lambda \boldsymbol{I}\right)^{-1} \boldsymbol{b}_n,$$

where

$$\boldsymbol{b}_n = \frac{\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_k\right) \widetilde{\boldsymbol{X}}_n \boldsymbol{\Sigma}_k^{-1}}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \mathbf{0}, \boldsymbol{\Sigma}_j\right)} \boldsymbol{x}_n.$$

Then we plug the renewed $\boldsymbol{\Sigma}_k, \; \pi_k \; (k = 1, 2, \cdots, K)$, and $\boldsymbol{Z}$ in (4) for the next round iteration. The optimization procedure for solving (4) is summarized in Algorithm 1.

A. MoG Regression for Subspace Clustering

Similar to the previous methods [15]–[17], our clustering method belongs to the based spectral clustering theory [20], [34] as well. After solving the MoG Regression problem (4), the desired representation matrix $\boldsymbol{Z}$ is found. We define the affinity matrix as

$$\boldsymbol{C} = \mid \boldsymbol{Z} \mid + \mid \boldsymbol{Z}^{\top} \mid,$$

where each value of entry C_{ij} in $\boldsymbol{C}$ measures the similarity between data points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

We illustrate the affinity matrices of 10 subjects clustering derived by SSC, LRR, LSR, CASS, CIL2, and the proposed MoG Regression, respectively with Figure 1 on the AR database, where the facial variations, illumination variations, and occlusions can be regarded as complex noise added to the original images. We notice that the affinity matrices derived by SSC and CASS are sparse, which due to the sparsity regularization, but the correlations within clusters are weak. So they may be less ability of grouping data points in the same cluster. On contrast, the affinity matrices derived by LRR, LSR, CIL2, and MoG Regression are very dense. The value of representation coefficients within clusters are large, which indicates the relevant clustering method to be good ability to group correlated data together. Meanwhile, we can see that the contrast between diagonal blocks and non-diagonal parts of MoG Regression is much higher than those of LRR, LSR, and CIL2.

TABLE I
THE CONTRAST (%) OF AFFINITY MATRICES IN FIGURE 1

SSC	LRR	LSR	CASS	CIL2	Ours
73.51	75.41	52.10	75.18	76.35	**80.32**

In order to quantitatively evaluate the contrast of the diagonal blocks against the non-diagonal parts of affinity matrices derived by each method, we define the contrast by $(S_d - S_{nd})/\|\boldsymbol{C}\|_1$, where S_d and S_{nd} are the sums of absolute values of entries in diagonal and non-diagonal parts, respectively. Table I lists the contrast of the affinity matrices from different methods. We notice that the contrast value of MoG Regression precedes other approaches. This demonstrates that, with complex noise corruption the data, our method is suitable for describing the distribution of noise, thus presenting stronger grouping effect and greater ability to recover the true subspace structures.

In the end, we employ the famous Normalize Cut [18] strategy on the affinity matrix $\boldsymbol{C}$ to produce the final clustering results.

III. THE GROUPING EFFECT

In this section we will theoretically expound the validity of the proposed MOG regression model for subspace clustering. A regression method shows the grouping effect if the coefficients of a group of correlated data tend to be equal. In [22], [35] the grouping effect is detailed studied. The validity of clustering come from the grouping effect for the models in [17], [19], [27] has been proved. In this section we will show that our proposed MoG Regression model also possesses the groping effect for correlated data. Now, we declare the grouping effect of MoG Regression as follows.

***Theorem** 3.1:* Given a sample point $\boldsymbol{x} \in \mathbb{R}^M$, the normalized data matrix $\boldsymbol{X}$ and the regularization parameter λ, let $\widehat{\boldsymbol{z}}$ be the optimal solution to

$$\min_{\boldsymbol{z}} \; -\ln\left(\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{X}\boldsymbol{z} - \boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)\right) + \lambda \parallel \boldsymbol{z} \parallel^2, \quad (8)$$

then there exists a constant a such that

$$\mid \widehat{z}^i - \widehat{z}^j \mid \leq \frac{a}{\lambda}\sqrt{\frac{1-\rho}{2}},$$

where $\rho = \cos\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$. Here we denote $\widehat{z}^i$ and $\widehat{z}^j$ as the i-th and j-th entries of vector $\widehat{\boldsymbol{z}}$, and $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ as the i-th and j-th columns of $\boldsymbol{X}$, respectively.

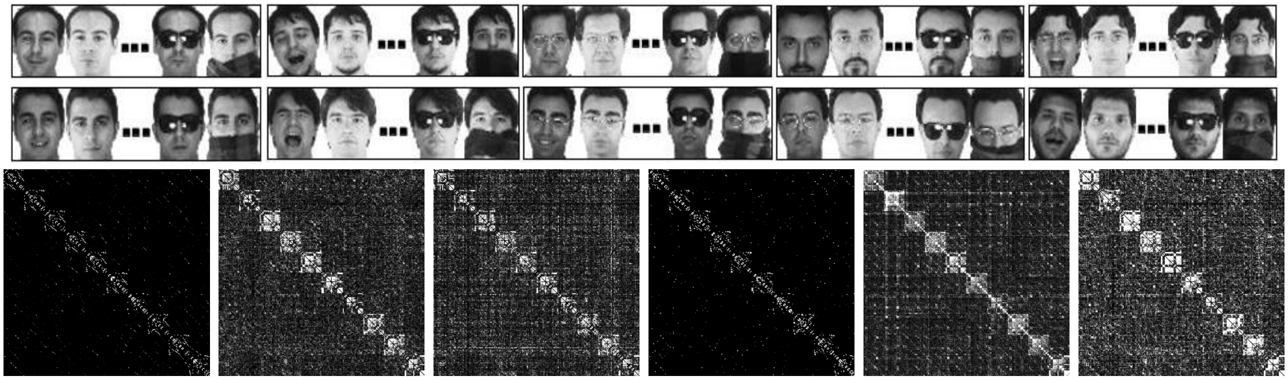
Algorithm 1: Finding the solution of (4) **by EM**

```
***Initialize:*** data matrix $\boldsymbol{X}$, covariance matrices $\boldsymbol{\Sigma}_k$, parameter $\lambda$, threshold value $\varepsilon$, initial representation matrix $\boldsymbol{Z}$, and the components number $K$.
***Repeat :***
**1:** Compute $\gamma_{n,k}$:
    $\gamma_{n,k} = \sum_{n=1}^{N} \frac{\pi_k \mathcal{N}(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_j)},$
    where $\widetilde{\boldsymbol{e}}_n^{old} = \widetilde{\boldsymbol{X}}_n \boldsymbol{z}_n^{old} - \boldsymbol{x}_n$.
**2:** Update the $\boldsymbol{\Sigma}_k$, $\pi_k$, and $\boldsymbol{Z}$:
    $\boldsymbol{\Sigma}_k^{new} = \frac{1}{\gamma_{n,k}}\left(\sum_{n=1}^{N} \frac{\pi_k \mathcal{N}(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_j)} \widetilde{\boldsymbol{e}}_n^{old}\left(\widetilde{\boldsymbol{e}}_n^{old}\right)^{\top} + \epsilon \boldsymbol{I}\right),$
    $\pi_k^{new} = \frac{1}{N}\sum_{n=1}^{N} \gamma_{n,k},$
    $\boldsymbol{z}_n^{new} = \left(\frac{\sum_{k=1}^{K} \xi_k \widetilde{\boldsymbol{X}}_n^{\top} \left(\boldsymbol{\Sigma}_k^{new}\right)^{-1} \widetilde{\boldsymbol{X}}_n}{\sum_{j=1}^{K} \xi_j} + 2\lambda \boldsymbol{I}\right)^{-1} \boldsymbol{b}_n,$
    where
    $\boldsymbol{b}_n = \frac{\sum_{k=1}^{K} \xi_k \widetilde{\boldsymbol{X}}_n \left(\boldsymbol{\Sigma}_k^{new}\right)^{-1}}{\sum_{j=1}^{K} \xi_j} \boldsymbol{x}_n,$
    and
    $\xi_k = \pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n^{old} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k^{new}\right), \quad \xi_j = \pi_j \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n^{old} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_j^{new}\right).$
**3:**
    $\boldsymbol{\Sigma}_k^{old} \longleftarrow \boldsymbol{\Sigma}_k^{new}, \quad \pi_k^{old} \longleftarrow \pi_k^{new}, \quad \boldsymbol{z}_n^{old} \longleftarrow \boldsymbol{z}_n^{new}.$
***Until :***
    $\parallel \boldsymbol{Z}^{old} - \boldsymbol{Z}^{new} \parallel_F \leq \varepsilon$ and $\parallel \boldsymbol{\Sigma}^{old} - \boldsymbol{\Sigma}^{new} \parallel_F \leq \varepsilon$.
***Output:*** The coefficient matrix $\boldsymbol{Z}$.
```

From the above **Theorem** 3.1 we can see that, if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are highly correlated, i.e. ρ is close to 1, then the upper bound of the difference between $\widehat{z}^i$ and $\widehat{z}^j$ approaches 0. In this case, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ would be grouped into the same cluster due to the grouping effect, which encourages the clustering performance.

In order to show the relevance level between a appointed datum and other points, we impose a constrain that diag $(\boldsymbol{Z}) = 0$ which means each datum is represented as a linear combination of other data rather than by itself. Next, we will show that the solution of (4) obeys this constrain.

***Theorem** 3.2:* Given the data matrix $\boldsymbol{X}$, for $i = 1, \cdots, n$, let $\boldsymbol{X}_{-i} = (\boldsymbol{x_1}, \cdots, \boldsymbol{x_{i-1}}, \boldsymbol{0}, \boldsymbol{x_{i+1}}, \cdots, \boldsymbol{x_n})$, the solution of optimal problem (8) in **Theorem** 3.1 satisfies $\boldsymbol{z}_{ii} = 0$. Furthermore, we get $diag\,(\boldsymbol{Z}) = \boldsymbol{0}$ for the problem (4).



(a) SSC [15] (b) LRR [16] (c) LSR [17] (d) CASS [19] (e) CIL2 [27] (f) Ours

Fig. 1. The affinity matrices of 10 objects obtained by different methods on the AR database.

Before the demonstration is shown, it is necessary to introduce a lemma [36], [37] which will be used during the process of proof.

***Lemma** 3.1:* Assuming that the matrix $\boldsymbol{R}_1$ is nonsingular, then the matrix

$$\boldsymbol{R} = \begin{pmatrix} \boldsymbol{R}_1 & \boldsymbol{R}_2 \\ \boldsymbol{R}_3 & \boldsymbol{R}_4 \end{pmatrix}$$

is invertible if and only if the Schur complement $\boldsymbol{R}_4 - \boldsymbol{R}_3\boldsymbol{R}_1^{-1}\boldsymbol{R}_2$ of $\boldsymbol{R}_1$ is invertible, and the following matrix

$$\begin{pmatrix} \tilde{\boldsymbol{R}}_1 & \tilde{\boldsymbol{R}}_2 \\ \tilde{\boldsymbol{R}}_3 & \tilde{\boldsymbol{R}}_4 \end{pmatrix}$$

is the inverse of $\boldsymbol{R}$. Where

$$\begin{cases} \tilde{\boldsymbol{R}}_1 = \boldsymbol{R}_1^{-1} + \boldsymbol{R}_1^{-1}\boldsymbol{R}_2(\boldsymbol{R}_4 - \boldsymbol{R}_3\boldsymbol{R}_1^{-1}\boldsymbol{R}_2)^{-1}\boldsymbol{R}_3\boldsymbol{R}_1^{-1} \\ \tilde{\boldsymbol{R}}_2 = -\boldsymbol{R}_1^{-1}\boldsymbol{R}_2(\boldsymbol{R}_4 - \boldsymbol{R}_3\boldsymbol{R}_1^{-1}\boldsymbol{R}_2)^{-1} \\ \tilde{\boldsymbol{R}}_3 = -(\boldsymbol{R}_4 - \boldsymbol{R}_3\boldsymbol{R}_1^{-1}\boldsymbol{R}_2)^{-1}\boldsymbol{R}_3\boldsymbol{R}_1^{-1} \\ \tilde{\boldsymbol{R}}_4 = (\boldsymbol{R}_4 - \boldsymbol{R}_3\boldsymbol{R}_1^{-1}\boldsymbol{R}_2)^{-1} \end{cases}$$

We list the proof of **Theorem** 3.2 in the appendix. This result shows that our operation is valid which meets the constrain $diag\left(\boldsymbol{Z}\right) = 0$ in problem (4).

IV. Asymptotic property

In fact, our method is similar to [15]–[17], [19], [27] belong to the self reconstruction category which needs the data points is sufficient. On the other hand, due to the non-convex of our model, it is necessary to analysis the asymptotic property of $\boldsymbol{Z}$ in (4). We assume that the number of mixture Gaussian components k and the dimension of each data point M are fixed. Let $\theta_{\boldsymbol{Z}}^{ini}$ and $\theta_{\boldsymbol{Z}}^{t}$ denote the initial value and the true parameter value of $\boldsymbol{Z}$, where $\theta_{\boldsymbol{Z}}^{ini} = \left(\boldsymbol{z}_1^{ini}, \dots, \boldsymbol{z}_N^{ini}\right)$ and $\theta_{\boldsymbol{Z}}^{t} = (\boldsymbol{z}_1^{t}, \dots, \boldsymbol{z}_N^{t})$. In the sprit of [25], [38], we obtain the following result.

***Theorem** 4.1:* Let $\boldsymbol{Z} \in \mathbb{R}^{M\times N}$ be the err matrix with independent and identically distributed $(i.i.d)$ in model (4), and keep k and M invariably. If $\frac{\lambda}{\sqrt{N}} = o\left(1\right)$, $\theta_{\boldsymbol{Z}}^{ini} - \theta_{\boldsymbol{Z}}^{t} = \mathcal{O}_p\left(N^{\frac{-1}{2}}\right)$, then, under the regularity conditions $(A) - (C)$ [25] on (7), then the local minimizer $\theta_{\boldsymbol{Z}}^{lm}$ of model (7) holds

$$\sqrt{n}\left(\boldsymbol{z}_i^{lm} - \boldsymbol{z}_i^{t}\right) \xrightarrow{d} \mathcal{N}\left(0, I_s\left(\boldsymbol{z}_i^{t}\right)^{-1}\right),\ i = 1,\ \dots\ ,N \quad (9)$$

Where the notions $\mathcal{O}_p$ and $\xrightarrow{d}$ denote the order to be equal in probability, and convergence in distribution [39] respectively, and $I_s\left(\cdot\right)$ denotes the information matrix [39].

V. A strategy For Finding the Number of Mixture Components

So far, we assume the number of the components K is fixed by empirical value. Inspired by [23], [24], [40] we provide a strategy to estimation the number of components K based on the minimum message length(MML) criterion [41], [42]. For the sake of self-contained, we give a glance of MML criterion.

To formalize the MML ideal, we notice that the equation (2) can be viewed as $p\left(\boldsymbol{E}|\Theta\right)$ which agrees with parameter

$$\Theta = (\pi_1, \cdots \pi_K, \Sigma_1, \cdots, \Sigma_K) \quad (10)$$

In the sprit of [41], [42], the parameter estimation issue boils down a transmission encoding problem. If you can find a short code for the provide data, you will obtain a good data generation mode [41], [43], [44]. This leads to

$$length\left(\boldsymbol{E}, \Theta\right) = length\left(\boldsymbol{E} \mid \Theta\right) + length\left(\Theta\right) \quad (11)$$

where $length\left(\boldsymbol{E}, \Theta\right) = -\ln p\left(\boldsymbol{E}, \Theta\right)$. In this context, the finite code length can only be obtained by quantizing the parameter Θ to finite precision after we undergo a loop of finding $\boldsymbol{Z}$. In fact, a fine precision is truncated, $length\left(\Theta\right)$ may be large, but $length\left(\boldsymbol{E} \mid \Theta\right)$ will be small because Θ closes to the optimal value. Conversely, a coarse precision is used, $length\left(\Theta\right)$ may be small, but $length\left(\boldsymbol{E} \mid \Theta\right)$ will be large because Θ departs from the optimal value [24]. In [24], the Taylor approximation method is used to balance the optimal quantization. In this case the optimal Θ is found by

$$\begin{aligned} \widehat{\Theta} = &\arg\min_{\Theta}\{-\ln p\left(\Theta\right) - \ln p\left(\boldsymbol{E} \mid \Theta\right) \\ &+ \frac{1}{2}\ln \mid \mathrm{F}\left(\Theta\right) \mid + \frac{D\left(\Theta\right)}{2}\left(1 + \ln\frac{1}{12}\right)\} \end{aligned} \quad (12)$$

where $\mid \mathrm{F}(\Theta) \mid$ denotes the determinant of the expected Fisher information matrix and $D(\Theta)$ denotes the dimension of the parameter Θ.

We adopt the approach in [24], which allows equation (12) to be rewritten as the following equivalent problem:

$$\begin{aligned}\widehat{\Theta} = & \arg\min_{\Theta}\{-\ln p(\Theta) - \ln p(\boldsymbol{E} \mid \Theta) \\ & + \frac{1}{2}\ln \mid \mathrm{F}_c(\Theta) \mid + \frac{D(\Theta)}{2}\left(1+\ln\frac{1}{12}\right)\}\end{aligned} \tag{13}$$

where $\mathrm{F}_c(\Theta)$ denotes the expected complete Fisher information matrix that is shown

$$\mathrm{F}_c(\Theta) = diag\left(\pi_1\sum_{i=1}^{n}\mathrm{F}_i(\Sigma_1), \cdots, \pi_K\sum_{i=1}^{n}\mathrm{F}_i(\Sigma_K), n\mathrm{M}\right)$$

$\mathrm{F}_i(\Sigma_k)$ is the Fisher information matrix of the ith observation for the kth component, and M is the Fisher information matrix of the multinomial distribution, that is $\mathrm{M} = diag(\pi_1, \cdots, \pi_K)^{-1}$ [45]. We adopt the same setting with [24] for the distributions $p(\Sigma_k)$ $(k = 1, \cdots, K)$ and $p(\pi_1, \cdots, \pi_K)$. Furthermore, let K_{no} denote the number of non-zero components and D denote the dimensionality of covariance matrix Σ_k $(k = 1, \cdots, K)$. We have

$$\begin{aligned}\widehat{\Theta} = & \arg\min_{\Theta}\mathcal{L}(\Theta, \boldsymbol{Z}) \\ = & \arg\min_{\Theta}\{\frac{D}{2}\sum_{k:\pi_k>0}\ln\left(\frac{n\pi_k}{12}\right) + \frac{\mathrm{K}_{no}}{2}\ln\left(\frac{n}{12}\right) \\ & + \frac{\mathrm{K}_{no}(D+1)}{2} - \sum_{n=1}^{N}\ln\left(\sum_{k=1}^{K}\pi_k\mathcal{N}(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k)\right)\} \\ & s.t.\ \ \pi_k \geq 0, \boldsymbol{\Sigma}_k \in \mathbb{S}^{+}, k = 1, ..., K, \sum_{k=1}^{K}\pi_k = 1\end{aligned} \tag{14}$$

in our particular case. Θ is the same as (10).

Based on this preparatory work, we will discuss how to determine the numbers of the Gaussian components. Notice that the optimal problem (14) is equal to use a symmetric improper Dirichlet type prior which is conjugate to multinomial likehoods [46]. Therefore, the modified estimation mixing weight is

$$\pi_k^{mnew} = \frac{\max\{0, \sum_{n=1}^{N}\gamma_{n,k} - \frac{D}{2}\}}{\sum_{j=1}^{K}\max\{0, \sum_{n=1}^{N}\gamma_{n,j} - \frac{D}{2}\}} \tag{15}$$

From the (16), it shows whether the kth component is annihilated. Once the required support $\frac{D}{2}$ can not be reached from the provide data, the mixing weight π_k equals to zero. In this scenario, the corresponding component is removed. Thus we can estimate the number of the components. As discussed in [24], [40], we should provide both the maximum K and the minimum K. If we begin with a large K, which leads to several empty components. To avoid this singular case, we use the component-wise EM procedure [24], [47]. The crucial difference between the modified EM algorithm and the older version in that the formulation (6) is replaced by formulation (16). We list the modified EM in algorithms 2. Each iterative t runs the component-wise E and M step. If one of the components is removed, the parameters are updatad accordingly, until $\mid \triangle Lenth \mid$ is below a given threshold. In this case, if the current length is not more than the $Lenth_{\min}$, the current parameters, coefficient matrix, and length are assigned to $\Theta_{\min}$, $\boldsymbol{Z}_{\min}$ and $\mathcal{L}(\Theta^{mne}, \boldsymbol{Z}^{new})$ respectively. For sake of exploring the full range of K^{+}, the less populated component is artificially removed, run the component-wise EM procedure again in sprit of [24].

We evaluate this tentative EM strategy on the contaminated Extended Yale Face Dataset B, and compare it with the algorithm 1. Although the modified EM algorithm 2 provides a way for automatically selecting the number of Gaussian components rather than by empirical value. The price it pays is more heavier computational burden than the common EM algorithm [24].

VI. Experiments

In this section, we evaluate the proposed MoG Regression model for clustering on the Hopkins 155 database [48], the Rotated MNIST Dataset [49], the AR database [50], and the Extended Yale Face Dataset B [51]. Experimental results demonstrate that the proposed method is valid and robust to noise in motion segmentation, handwritten digits clustering, and complex face clustering.

We also run SSC [15], LRR [16], LSR [17], CASS [19], and CIL2 [27] on these datasets. Meanwhile, we tune the parameters of each method so that every model achieves its best performance, and the clustering accuracy [15] is employed in quantitative evaluation.

$$clustering\ accuracy = 1 - \frac{missclustered\ points}{total\ data\ points} \times 100\%$$

The comparison results show that our approach outperforms the mentioned five state-of-the-art methods.

A. Hopkins 155 Database

The Hopkins 155 motion segmentation database [48] that contains altogether 155 video sequences. Thereinto, 120 of the videos contain 2 motions, and the rest of the videos have 3 motions. On average, there are 266 feature trajectories and 30 frames in the each sequence of the 2 motions, and 398 feature trajectories and 29 frames in each sequence of the 3 motions. For each video sequence, the task of subspace clustering is to group the the point trajectories. In this situation, we will recorder the average accuracy over 2 motions and 3 motions videos respectively.

Firstly, we use PCA method to reduce the dimensionality of the data matrix. Then we test the proposed MoG Regression method on each video sequence. Some motion segmentation visually results of our approach are shown in Figure 2, where motions of different objects and background trajectories can be accurately segmented.

Table II lists the average clustering accuracies of different methods. We can see that MoG Regression achieves significantly higher accuracies than the other methods.

Algorithm 2: Finding the solution of (4) by modified EM based on MML

```
Initialize: input data matrix $\boldsymbol{X}$, covariance matrices $\boldsymbol{\Sigma}_k$, $\pi_k$, parameter $\lambda$, threshold value $\varepsilon$, initial representation matrix $\boldsymbol{Z}^{old}$, and the components number $K_{\min}$ and $K_{\max}$
output: The minimum length mixture model: The coefficient matrix $\boldsymbol{Z}$, $K^+$
Set $t \longleftarrow 0$, $K^+ \longleftarrow K_{\max}$, $Lenth_{\min} \longleftarrow +\infty$
while $K^+ \geq K_{\min}$ do
repeat
t = t + 1;
for k = 1 to $K_{\max}$ do
E–step: Compute $\gamma_{n,k}$:
$$\gamma_{n,k} = \sum_{n=1}^{N} \frac{\pi_k \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K_{\max}} \pi_j \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_j\right)},$$
where $\widetilde{\boldsymbol{e}}_n^{old} = \widetilde{\boldsymbol{X}}_n \boldsymbol{z}_n^{old} - \boldsymbol{x}_n$.
M–step:
$$\pi_k^{mnew} = \frac{\max\{0, \sum_{n=1}^{N} \gamma_{n,k} - \frac{D}{2}\}}{\sum_{j=1}^{K} \max\{0, \sum_{n=1}^{N} \gamma_{n,j} - \frac{D}{2}\}}$$
if $\pi_k^{mnew} > 0$
$$\boldsymbol{\Sigma}_k^{mnew} = \frac{1}{\gamma_{n,k}} \left( \sum_{n=1}^{N} \frac{\pi_k^{mnew} \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K_{\max}} \pi_j^{mnew} \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n \mid \boldsymbol{0}, \boldsymbol{\Sigma}_j\right)} \widetilde{\boldsymbol{e}}_n^{old} \left(\widetilde{\boldsymbol{e}}_n^{old}\right)^{\top} + \epsilon \boldsymbol{I} \right)$$
else
$K^+ = K^+ - 1$
end if
end for
updata $\boldsymbol{Z}$
$$\boldsymbol{z}_n^{new} = \left( \frac{\sum_{k=1}^{K_{\max}} \xi_k \widetilde{\boldsymbol{X}}_n^{\top} \left(\boldsymbol{\Sigma}_k^{mnew}\right)^{-1} \widetilde{\boldsymbol{X}}_n}{\sum_{j=1}^{K_{\max}} \xi_j} + 2\lambda \boldsymbol{I} \right)^{-1} \boldsymbol{b}_n,$$
where
$$\boldsymbol{b}_n = \frac{\sum_{k=1}^{K_{\max}} \xi_k \widetilde{\boldsymbol{X}}_n \left(\boldsymbol{\Sigma}_k^{mnew}\right)^{-1}}{\sum_{j=1}^{K_{\max}} \xi_j} \boldsymbol{x}_n,$$
and
$$\xi_k = \pi_k^{mnew} \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n^{old} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k^{new}\right), \;\; \xi_j = \pi_j^{mnew} \mathcal{N}\left(\widetilde{\boldsymbol{e}}_n^{old} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_j^{mnew}\right).$$
$\boldsymbol{Z}^{old} \leftarrow \boldsymbol{Z}^{new}$
Compute the optimal length using the new parameters and $\boldsymbol{Z}$
Compute the length $\mathcal{L}\left(\Theta^{mne}, \boldsymbol{Z}^{new}\right)$ via (14)
until
$\mid \triangle Lenth \mid \leq \varepsilon$
if $\mathcal{L}\left(\Theta^{mne}, \boldsymbol{Z}^{new}\right)$ < $Lenth_{\min}$
$\mathcal{L}\left(\Theta^{mne}, \boldsymbol{Z}^{new}\right) \leftarrow Lenth_{\min}$
$\Theta_{\min} \leftarrow \Theta^{mne}$
$\boldsymbol{Z}_{\min} \leftarrow \boldsymbol{Z}^{new}$
end if
k = $\arg\min_k \{\pi_k^{mnew}\}_+$
$\pi_k^{mnew} \leftarrow 0$
$K^+ \leftarrow K^+ - 1$
end while
Using the representation matrix $\boldsymbol{Z}$ to cluster
```

TABLE II
THE CLUSTERING ACCURACIES (%) ON THE HOPKINS 155 DATABASE.

	SSC	LRR	LSR	CASS	CIL2	Ours
2 motions	95.69	96.43	97.48	97.01	97.63	**98.76**
3 motions	91.97	92.35	93.21	94.06	94.34	**95.03**

B. MNIST-Back-Rand Dataset

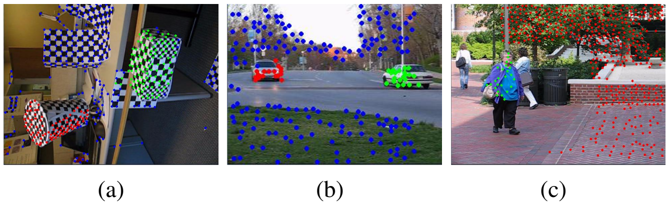

(a) (b) (c)

Fig. 2. Exemplar results of motion segmentation on the Hopkins 155 Database. (a) Checkerboard. (b) Cars. (c) People.

The MNIST-back-rand database contains 50000 images of hand-written digits from 0 to 9. It is first selected from the MNIST dataset [52] and then transformed them into more challenging images by adding some random noises into the original pictures. Figure 3 shows some example images from the dataset.

In order to alleviate memory consumption in our experiments, we randomly select 10 images for each digit to build a subset, thus the candidate dataset contains 100 samples. The experiment results are reported in Table III, which declares that the advantage of our method is notable. This experiment also shows that when the data are corrupted with non-Gaussian or complex noise, the proposed method is more capable of clustering the underlying subspaces with the help of MoG.

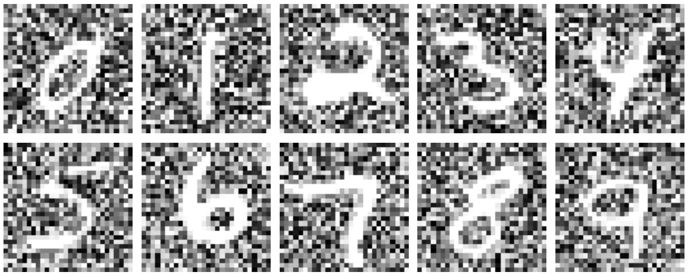


Fig. 3. Examples of the MNIST-back-rand database, where digits are corrupted with random noise.

TABLE III
THE CLUSTERING ACCURACIES (%) ON THE MNIST-BACK-RAND DATABASE

SSC	LRR	LSR	CASS	CIL2	Ours
33.56	22.85	20.55	29.05	36.50	**51.98**

C. AR Dataset

The AR database [50] consists of over 4,000 facial images corresponding to 126 subjects which include 70 men and 56 women. For each subject, there exist 26 facial images be taken in two separate sessions. These images suffer severely different facial variations, including various facial expressions such as neutral, smile, anger, and scream, illumination variations (left light on, right light on, and all side lights on), and varying degrees of occlusion by sunglasses or scarf.

We design two subspace clustering tasks by selecting first 5 and 10 subjects based on this dataset, respectively. The clustering results on the AR database of different algorithms are recorded in Table IV. We can see that the performance of MoG Regression method for subspace clustering is superior to the other methods in both clustering tasks. This is because MoG Regression has both a strong grouping effect on this

TABLE IV
THE CLUSTERING ACCURACIES (%) ON THE AR DATABASE.

	SSC	LRR	LSR	CASS	CIL2	Ours
5 subjects	83.05	84.41	87.69	78.46	85.38	**93.85**
10 subjects	75.06	78.54	63.07	77.69	80.39	**88.85**

challenging database and reasonably model the noise, which can be seen in Figure 1.

D. *Extended Yale Face Dataset B*

There are total 2,414 frontal face images of 38 subjects in the Extended Yale Face Dataset B [51], and each subject consists of 64 images under various lighting, poses, and illumination conditions. In order to further show the ability of MOG model for describing the noises, we add the noise on each image by replacing random its pixels with samples from a uniform distribution on the interval from 0 to 255 [27], and the percentage of corrupted pixels range from 10% to 100%. In order to reduce the computational cost and memory requirements, we tailor the grayscale images to a resolution of 32×32 pixels.

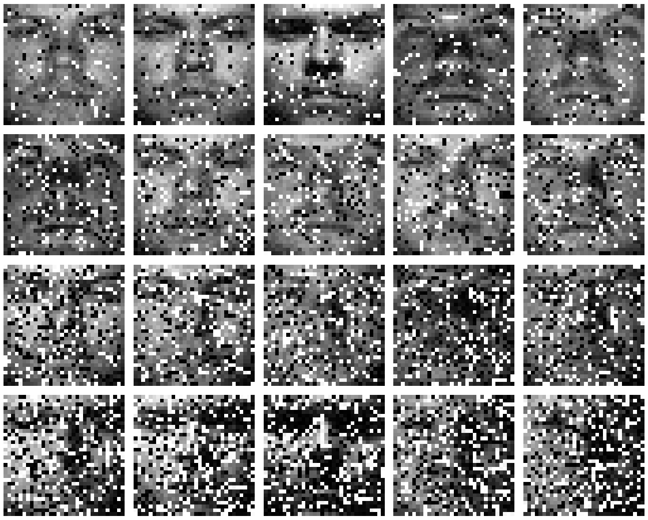


Fig. 4. Examples of the corrupted face images from the Yale dataset database, each row pictures show 10%, 20% 30%, 40% of pixels corrupted from top to bottom, respectively.

The clustering accuracies of all methods on the corrupted Extended Yale B database are reported by Figure 5.

From Figure 5 we can see that the proposed method performs much better when face images are randomly contaminated at a level from 10% to 40%, exhibiting better adaptability and greater robustness in noise situation. When the percentage of corrupted pixels is over than 60%, the discriminative information are destructive damaged, thus will weaken the performance of all methods.

If the pixels are corrupted over 40% the accuracies are less than 40%. As a consequence, we compare the K, accuracies obtained by algorithm 1 with algorithm 2 on the data set that the pixels are corrupted not more than 40%. Fig 4 shows some examples. Fig 6 reveals the experimental results. We take $K_{\min} = 1$ and $K_{\max} = 25$ for iteration. It can be seen that, the number of Gaussian components of both two algorithms are increase when the level of the pixels corruption is high. While the accuracies obtained by these two algorithms go down when the pixels corruption has high level. In the case of 10% and 20% pixels are corrupted, the estimated K is 2 that equals to the empirical value. In case of 30% and 40% pixels are corrupted, the estimated values of K are 3 and 4 respectively. In this case the empirical values of K are 4 and 5. The accuracies that obtained by empirical values K have relatively small deviations from the accuracies that obtained by MML based. If the data contains simple unknown noise we advocate to use the empirical value K. When the data surfers from serious corruption, we need more Gaussian components to approximate the unknown noises. In this scenario using the explorative method is seems blind. Notice that the accuracies between MML based EM algorithm and the common EM algorithm are different under the optimal K estimated by MML based EM algorithm. The reason is that the update mechanism of mixing weights and covariance matrixes MML based is different from the common EM algorithm.

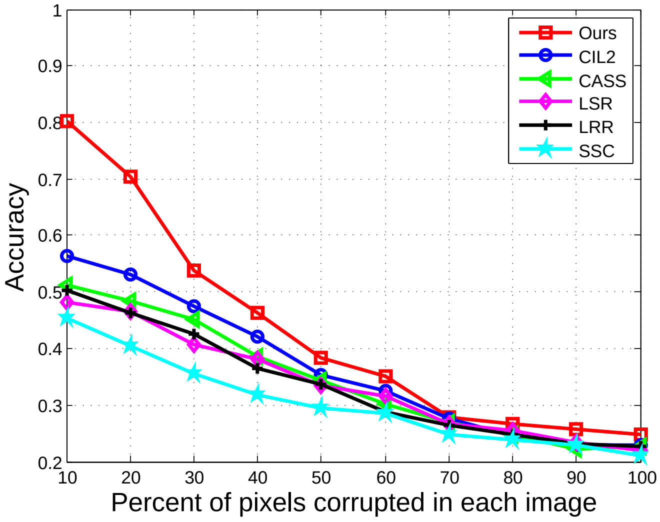


Fig. 5. The clustering accuracies (%) with pixel corruption on the Extended Yale B database.

VII. CONCLUSIONS

In this paper, we propose a new subspace clustering method by employing the MoG model to describe the distribution of complex noise. In fact, the SSC, LRR, LSR, CASS, CIL2, and MOG are all reconstruction based methods for subspace clustering by computing a reconstruction matrix which is also called coefficient matrix. Using the model (1) can be written in a unified form. The models SSC, LRR, LSR, and CASS describe the noise as the unimodal Gaussian or sparse type, while the CIL2 borrows the ideal of [28] which deals with the non-Gaussian noise especially for impulsive noise. In real scenario, the noise goes beyond the Gaussian or impulsive types. Inspired by the property of mix Gaussian distribution, we use MOG model to group the subspaces. One hand,

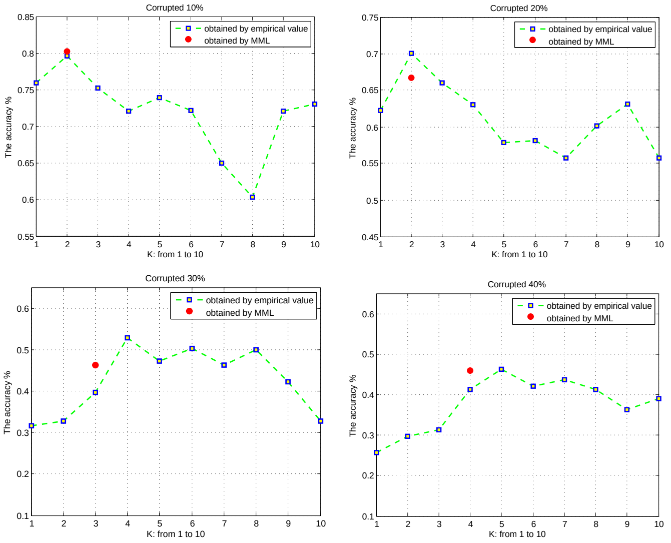


Fig. 6. Comparison of the empirical value K with the estimated K based on MML, and their clustering accuracies

our proposed model can character the complex noise by the property mix Gaussian model, on the other hand we give the theoretical analysis shows that the MoG Regression model maintains the grouping effect. The experiments on motion segmentation, handwritten digits clustering, and complex face clustering demonstrate the superiority of our proposed method, regarding stability and robustness in handling general noise, over the state-of-the-art subspace clustering methods, SSC, LRR, LSR, CASS, and CIL2 which assume Gaussian or sparse noise or impulsive type. In the future, we will deal with the accelerating aspect of the solution for MoG Regression.

VIII. Appendix

Proof of **theorem** 3.1:

Proof: Let

$$f\left(\boldsymbol{z}\right) = -\ln\left(\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{X}\boldsymbol{z} - \boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)\right) + \lambda \parallel \boldsymbol{z} \parallel^2 .$$

Since $\widehat{\boldsymbol{z}} = \mathop{\textbf{argmin}}_{\boldsymbol{z}} f\left(\boldsymbol{z}\right)$, we have

$$\left.\frac{\partial f\left(\boldsymbol{z}\right)}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}=\widehat{\boldsymbol{z}}} = 0.$$

This gives

$$\frac{\boldsymbol{x}_i^{\top}\left(\sum_{k=1}^{K}\xi_k\boldsymbol{\Sigma}_k^{-1}\right)\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right)}{\sum_{k=1}^{K}\xi_k} + 2\lambda\widehat{z}^i = 0,$$

and

$$\frac{\boldsymbol{x}_j^{\top}\left(\sum_{k=1}^{K}\xi_k\boldsymbol{\Sigma}_k^{-1}\right)\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right)}{\sum_{k=1}^{K}\xi_k} + 2\lambda\widehat{z}^j = 0,$$

where $\xi_k = \pi_k\mathcal{N}\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)$.

From the above two equations for $\widehat{z}^i$ and $\widehat{z}^j$ we deduce that

$$\widehat{z}^i - \widehat{z}^j = \frac{\left(\boldsymbol{x}_i^{\top} - \boldsymbol{x}_j^{\top}\right)\left(\sum_{k=1}^{K}\xi_k\boldsymbol{\Sigma}_k^{-1}\right)\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right)}{2\lambda\sum_{k=1}^{K}\xi_k}.$$

Note that

$$\begin{aligned}\left\|\sum_{k=1}^{K}\xi_k\boldsymbol{\Sigma}_k^{-1}\right\|_2 &\leq \sum_{k=1}^{K}\xi_k \parallel \boldsymbol{\Sigma}_k^{-1} \parallel_2 \\ &\leq \left(\max_k \parallel \boldsymbol{\Sigma}_k^{-1} \parallel_2\right)\sum_{k=1}^{K}\xi_k.\end{aligned}$$

So we get

$$\mid \widehat{z}^i - \widehat{z}^j \mid \leq \frac{\parallel \boldsymbol{x}_i - \boldsymbol{x}_j \parallel_2 \cdot \parallel \boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x} \parallel_2 \cdot \left(\max_k \parallel \boldsymbol{\Sigma}_k^{-1} \parallel_2\right)}{2\lambda}.$$

Note that $\widehat{\boldsymbol{z}}$ is a minimizer of $f(\boldsymbol{z})$. So we have

$$f\left(\widehat{\boldsymbol{z}}\right) \leq f\left(\boldsymbol{0}\right),$$

which yields

$$\ln\left(\sum_{k=1}^{K}\pi_k\mathcal{N}\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)\right) \geq \ln\left(\sum_{k=1}^{K}\pi_k\mathcal{N}\left(\boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)\right).$$

On the other hand, if we define

$$V = \mathop{\textbf{argmax}}_{k} \pi_k$$

and

$$S = \mathop{\textbf{argmax}}_{k} \mathcal{N}\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right),$$

then we get

$$\ln\left(K\pi_V\mathcal{N}\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_S\right)\right) \geq \ln\left(\sum_{k=1}^{K}\pi_k\mathcal{N}\left(\boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)\right),$$

which is equivalent to

$$\begin{aligned}&\frac{\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right)^{\top}\boldsymbol{\Sigma}_S^{-1}\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right)}{2} \\ &\leq \ln\left(\frac{K\pi_V}{\sqrt[M]{2\pi} \mid \boldsymbol{\Sigma}_S \mid^{-1}\left(\sum_{k=1}^{K}\pi_k\mathcal{N}\left(\boldsymbol{X}\boldsymbol{z} - \boldsymbol{x} \mid \boldsymbol{0}, \boldsymbol{\Sigma}_k\right)\right)}\right).\end{aligned}$$

Since $\boldsymbol{\Sigma}_S^{-1}$ is a symmetric positive definite matrix, whose unitary similar matrix is a diagonal matrix, we can list the diagonal entries in descending order. Then We have

$$\mid \boldsymbol{\Sigma}_S \mid^{-1} = U^{\top}\begin{pmatrix}\lambda_1 & \cdots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \lambda_{\min}\end{pmatrix}U,$$

Where U is the unitary matrix, $\lambda_1 \geq \cdots\cdots \geq \lambda_{\min}$ denote the eigenvalues of $\boldsymbol{\Sigma}_S^{-1}$. Thus we get

$$\begin{aligned}&\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right)^{\top}\boldsymbol{\Sigma}_S^{-1}\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right) \\ &= \left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right)^{\top}U^{\top}\begin{pmatrix}\lambda_1 & \cdots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \lambda_{\min}\end{pmatrix}U\left(\boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x}\right) \\ &\geq \lambda_{\min} \parallel \boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x} \parallel^2 .\end{aligned}$$

It yields to

$$\parallel \boldsymbol{X}\widehat{\boldsymbol{z}} - \boldsymbol{x} \parallel^2 \leq Q,$$

where

$$Q=\frac{1}{\lambda_{\min}}\ln\left(\frac{K\pi_V}{\sqrt[M]{2\pi}\mid\boldsymbol{\Sigma}_S\mid^{-1}\left(\sum\limits_{k=1}^{K}\pi_k\mathcal{N}\left(\boldsymbol{X}\boldsymbol{z}-\boldsymbol{x}\mid\boldsymbol{0},\boldsymbol{\Sigma}_k\right)\right)}\right) \tag{16}$$

Then we get

$$\mid\widehat{z}^i-\widehat{z}^j\mid\leq\frac{a}{\lambda}\sqrt{\frac{1-\rho}{2}}, \tag{17}$$

where

$$a=\left(\max_k\parallel\boldsymbol{\Sigma}_k^{-1}\parallel_2\right)\sqrt{Q}$$

is a constant.

Proof of **theorem** 3.2:

Proof: The formulation of $\boldsymbol{z}_n$ in **Algorithm 1** can be rewritten as the the following compact form

$$\boldsymbol{z}_n^{new}=(\widetilde{\boldsymbol{X}}_n^\top\boldsymbol{D}_n\widetilde{\boldsymbol{X}}_n+2\lambda\boldsymbol{I})^{-1}\widetilde{\boldsymbol{X}}_n\widetilde{\boldsymbol{b}}_n,$$

where

$$\boldsymbol{D}_n=\frac{\sum_{k=1}^{K}\pi_k\mathcal{N}(\widetilde{\boldsymbol{X}}_n\boldsymbol{z}_n-\boldsymbol{x}_n\mid\boldsymbol{0},\boldsymbol{\Sigma}_k^{new})\,(\boldsymbol{\Sigma}_k^{new})^{-1}}{\sum_{j=1}^{K}\pi_j\mathcal{N}(\widetilde{\boldsymbol{X}}_n\boldsymbol{z}_n-\boldsymbol{x}_n\mid\boldsymbol{0},\boldsymbol{\Sigma}_j^{new})}$$

and

$$\widetilde{\boldsymbol{b}}_n=\frac{\sum_{k=1}^{K}\pi_k\mathcal{N}(\widetilde{\boldsymbol{X}}_n\boldsymbol{z}_n-\boldsymbol{x}_n\mid\boldsymbol{0},\boldsymbol{\Sigma}_k^{new})\,(\boldsymbol{\Sigma}_k^{new})^{-1}}{\sum_{j=1}^{K}\pi_j\mathcal{N}(\widetilde{\boldsymbol{X}}_n\boldsymbol{z}_n-\boldsymbol{x}_n\mid\boldsymbol{0},\boldsymbol{\Sigma}_j^{new})}\boldsymbol{x}_n.$$

Since the n-th column of $\widetilde{\boldsymbol{X}}_n$ is $\boldsymbol{0}$, the invertible symmetry matrix $(\widetilde{\boldsymbol{X}}_n^\top\boldsymbol{D}_n\widetilde{\boldsymbol{X}}_n+2\lambda\boldsymbol{I})^{-1}$ can be represented as the partitioned matrix form

$$\left(\widetilde{\boldsymbol{X}}_n^\top\boldsymbol{D}_n\widetilde{\boldsymbol{X}}_n+2\lambda\boldsymbol{I}\right)^{-1}=\begin{pmatrix}\boldsymbol{B}_{11}&\boldsymbol{B}_{12}\\\boldsymbol{B}_{21}&\boldsymbol{B}_{22}\end{pmatrix}^{-1},$$

where $\boldsymbol{B}_{11}$ and $\boldsymbol{B}_{22}$ are both invertible matrix.

Furthermore, $\boldsymbol{B}_{11}$ can be written as the block matrix

$$\boldsymbol{B}_{11}=\begin{pmatrix}\mathbb{B}_{11}&\boldsymbol{0}\\\boldsymbol{0}^\top&2\lambda\end{pmatrix}.$$

Similarly, $\boldsymbol{B}_{12}$ and $\boldsymbol{B}_{21}$ has the following form:

$$\boldsymbol{B}_{12}=\begin{pmatrix}\mathbb{B}_{12}\\\boldsymbol{0}^\top\end{pmatrix},\quad\boldsymbol{B}_{21}=\begin{pmatrix}\mathbb{B}_{21}&\boldsymbol{0}\end{pmatrix}.$$

On one hand, we obtain the inverse matrix $(\widetilde{\boldsymbol{A}}_n^\top\boldsymbol{D}_n\widetilde{\boldsymbol{A}}_n+2\lambda\boldsymbol{I})^{-1}$ as

$$\begin{pmatrix}\widehat{\boldsymbol{B}}_{11}&\boldsymbol{0}&\widehat{\boldsymbol{B}}_{12}\\\boldsymbol{0}^\top&\frac{1}{2\lambda}&\boldsymbol{0}^\top\\\widehat{\boldsymbol{B}}_{21}&\boldsymbol{0}&\widehat{\boldsymbol{B}}_{22}\end{pmatrix}$$

where $\widehat{\boldsymbol{B}}_{11}$, $\widehat{\boldsymbol{B}}_{12}$, $\widehat{\boldsymbol{B}}_{21}$ and $\widehat{\boldsymbol{B}}_{22}$ can be computed by lemma 3.1 using $\boldsymbol{B}_{11}$, $\boldsymbol{B}_{12}$, $\boldsymbol{B}_{21}$ and $\boldsymbol{B}_{22}$. Here we omit the computation process.

On the other hand $\widetilde{\boldsymbol{X}}_n\widetilde{\boldsymbol{b}}_n$ can be written as

$$\begin{pmatrix}\boldsymbol{x}_1^\top\widetilde{\boldsymbol{b}}_n&\cdots&\boldsymbol{x}_{n-1}^\top\widetilde{\boldsymbol{b}}_n&0&\boldsymbol{x}_{n+1}^\top\widetilde{\boldsymbol{b}}_n&\cdots&\boldsymbol{x}_M^\top\widetilde{\boldsymbol{b}}_n\end{pmatrix}^\top,$$

Here, M is denoted the dimension of $\widetilde{\boldsymbol{X}}_n\widetilde{\boldsymbol{b}}_n$.

2 Then we get the n-th element of $\boldsymbol{x}_n^{new}$,

$$\begin{aligned}\boldsymbol{x}_{nn}^{new}&=\boldsymbol{0}^\top\cdot\begin{pmatrix}\boldsymbol{x}_1^\top\boldsymbol{b}_n\\\cdots\\\boldsymbol{x}_{n-1}^\top\boldsymbol{b}_n\end{pmatrix}+0\cdot\frac{1}{2\lambda}+\boldsymbol{0}^\top\cdot\begin{pmatrix}\boldsymbol{x}_{n+1}^\top\boldsymbol{b}_n\\\cdots\\\boldsymbol{x}_M^\top\boldsymbol{b}_n\end{pmatrix}\\&=0,\end{aligned}$$

which draws the conclusion that $diag(\boldsymbol{Z})=\boldsymbol{0}$.

Proof of **theorem** 4.1:

Proof: let

$$Q\left(\boldsymbol{z}_i\right)=-\mathcal{L}\left(\boldsymbol{z}_i\right)+\lambda\parallel\boldsymbol{z}_i\parallel_F^2.$$

where $-\mathcal{L}\left(\boldsymbol{z}_i\right)$ replaces $-\sum_{n=1}^{N}\ln\left(\sum_{k=1}^{K}\pi_k\mathcal{N}\left(\widetilde{\boldsymbol{e}}_n\mid\boldsymbol{0},\boldsymbol{\Sigma}_k\right)\right)$ for simplification.

Since $\boldsymbol{z}_i^{lm}$ is the local minimizer of $Q\left(\boldsymbol{z}_i\right)$, which reads

$$\boldsymbol{0}=\frac{\partial Q\left(\boldsymbol{z}_i\right)}{\partial\boldsymbol{z}_i}\mid_{\boldsymbol{z}_i=\boldsymbol{z}_i^{lm}}=-\frac{\partial\mathcal{L}\left(\boldsymbol{z}_i\right)}{\partial\boldsymbol{z}_i}\mid_{\boldsymbol{z}_i=\boldsymbol{z}_i^{lm}}+2\lambda\boldsymbol{z}_i^{lm} \tag{18}$$

We let $\frac{\partial\mathcal{L}\left(\boldsymbol{z}_i^{lm}\right)}{\partial\boldsymbol{z}_i}$ denote $\frac{\partial\mathcal{L}\left(\boldsymbol{z}_i\right)}{\partial\boldsymbol{z}_i}\mid_{\boldsymbol{z}_i=\boldsymbol{z}_i^{lm}}$. Thanks for the Taylor expansion, we obtain

$$\begin{aligned}\frac{\partial\mathcal{L}\left(\boldsymbol{z}_i^{lm}\right)}{\partial\boldsymbol{z}_i}&=\frac{\partial\mathcal{L}\left(\boldsymbol{z}_i^t\right)}{\partial\boldsymbol{z}_i}+\left(\frac{\partial^2\mathcal{L}\left(\boldsymbol{z}_i^t\right)}{\partial\boldsymbol{z}_i^2}\right)^\top\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\\&+\frac{1}{2}\left(I\otimes\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\right)^\top\left(\frac{\partial^3\mathcal{L}\left(\boldsymbol{z}_i^{nt}\right)}{\partial\boldsymbol{z}_i^3}\right)^\top\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\end{aligned} \tag{19}$$

where $\boldsymbol{z}_i^{nt}$ belongs to the ball neighbourhood of $\boldsymbol{z}_i^t$ and I is a identity matrix of size $M\times M$. Using (19), we rearrange the equation (18) that yields to

$$\begin{aligned}\boldsymbol{0}=&-\frac{1}{n}\frac{\partial\mathcal{L}\left(\boldsymbol{z}_i^t\right)}{\partial\boldsymbol{z}_i}-\frac{1}{n}\left(\frac{\partial^2\mathcal{L}\left(\boldsymbol{z}_i^t\right)}{\partial\boldsymbol{z}_i^2}\right)^\top\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\\&-\frac{1}{2n}\left(I\otimes\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\right)^\top\left(\frac{\partial^3\mathcal{L}\left(\boldsymbol{z}_i^{nt}\right)}{\partial\boldsymbol{z}_i^3}\right)^\top\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\\&+\frac{2\lambda}{n}\boldsymbol{z}_i^{lm}\end{aligned} \tag{20}$$

Now employing the regularity conditions $(A)-(C)$ [25], which reads

$$\frac{1}{n}\left(\frac{\partial^2\mathcal{L}\left(\boldsymbol{z}_i^t\right)}{\partial\boldsymbol{z}_i^2}\right)^\top=-I\left(\boldsymbol{z}_i^t\right)+o_p\left(1\right)$$

where $I\left(\boldsymbol{z}_i^t\right)$ is the information matrix at $\boldsymbol{z}_i^t$.

$$\frac{1}{n}\left(\frac{\partial^3\mathcal{L}\left(\boldsymbol{z}_i^t\right)}{\partial\boldsymbol{z}_i^3}\right)^\top=\mathcal{O}_p\left(1\right)$$

Notice the consistency of $\theta_{\boldsymbol{Z}}^{ini}-\theta_{\boldsymbol{Z}}^t=\mathcal{O}_p\left(N^{\frac{-1}{2}}\right)$, which means that $\theta_{\boldsymbol{Z}}^{ini}-\theta_{\boldsymbol{Z}}^t=o_p\left(1\right)$. Thus

$$\frac{1}{2n}\left(I\otimes\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\right)^\top\left(\frac{\partial^3\mathcal{L}\left(\boldsymbol{z}_i^{nt}\right)}{\partial\boldsymbol{z}_i^3}\right)^\top=o_p\left(1\right)$$

After rearrange (20), we get

$$\begin{aligned}-\frac{1}{\sqrt{n}}\frac{\partial\mathcal{L}\left(\boldsymbol{z}_i^t\right)}{\partial\boldsymbol{z}_i}&=\left(-I\left(\boldsymbol{z}_i^t\right)+o_p\left(1\right)\right)\sqrt{n}\left(\boldsymbol{z}_i^{lm}-\boldsymbol{z}_i^t\right)\\&+\frac{2\lambda}{\sqrt{n}}\boldsymbol{z}_i^{lm}\end{aligned} \tag{21}$$

Note that $\frac{\lambda}{\sqrt{N}} = o(1)$ by assumed condition, $I(\boldsymbol{z}_i^t)$ is the mentioned information matrix. Using the center limit theory, we get $-\frac{1}{\sqrt{n}}\frac{\partial \mathcal{L}(\boldsymbol{z}_i^t)}{\partial \boldsymbol{z}_i} \xrightarrow{d} \mathcal{N}(0, I(\boldsymbol{z}_i^t))$, which leads to the conclusion.

References

[1] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 212–225, 2008.

[2] R. Vidal and R. Hartley, "Motion segmentation with missing data using powerfactorization and GPCA," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2004, pp. II–310.

[3] J. Ho, M. H. Yang, J. Lim, K. C. Lee, and D. Kriegman, "Clustering appearances of objects under varying illumination conditions," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2003, pp. I–11.

[4] K. H. Wei Hong, John Wright and Y. Ma, "Multi-scale hybrid linear models for lossy image representation," in *IEEE Transactions on Image Processing*, vol. 15, 2006, pp. 3655–3671.

[5] Y. Ma, A. Y. Yang, H. Derksen, and R. Fossum, "Estimation of subspace arrangements with applications in modeling and segmenting mixed data," *SIAM review*, vol. 50, no. 3, pp. 413–458, 2008.

[6] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1945–1959, 2005.

[7] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *International Journal of Computer Vision*, vol. 29, no. 3, pp. 159–179, 1998.

[8] P. S. Bradley and O. L. Mangasarian, "K-plane clustering," *Journal of Global Optimization*, vol. 16, no. 1, pp. 23–32, 2000.

[9] P. K. Agarwal and N. H. Mustafa, "K-means projective clustering," in *Proceedings of the Twenty-third ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 14-16, 2004, Paris, France*, 2004, pp. 155–165.

[10] Y. Ma, H. Derksen, W. Hong, and J. Wright, "Segmentation of multivariate mixed data via lossy data coding and compression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1546–1562, 2007.

[11] S. R. Rao, R. Tron, R. Vidal, and Y. Ma, "Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[12] A. Y. Yang, S. R. Rao, and Y. Ma, "Robust statistical estimation and segmentation of multiple subspaces," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2006, pp. 99–99.

[13] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[14] J. Yan and M. Pollefeys, "A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate," in *Proceedings of European Conference on Computer Vision*, 2006, pp. 94–106.

[15] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[16] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.

[17] C. Y. Lu, H. Min, Z. Q. Zhao, L. Zhu, D. S. Huang, and S. Yan, "Robust and efficient subspace segmentation via least squares regression," in *Proceedings of European Conference on Computer Vision*, 2012, pp. 347–360.

[18] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[19] C. Lu, J. Feng, Z. Lin, and S. Yan, "Correlation adaptive subspace segmentation by trace lasso," in *Proceedings of IEEE International Conference on Computer Vision*, 2013, pp. 1345–1352.

[20] F. R. Chung, *Spectral graph theory*. American Mathematical Soc., 1997, vol. 92.

[21] Q. Zhao, D. Meng, Z. Xu, W. Zuo, and L. Zhang, "Robust principal component analysis with complex noise," in *Proceedings of International Conference on Machine Learning*, 2014.

[22] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[23] J. J. Oliver, R. A. Baxter, and C. S. Wallace, "Unsupervised learning using mml," in *ICML*, 1996, pp. 364–372.

[24] M. A. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 3, pp. 381–396, 2002.

[25] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001.

[26] B. Li, Y. Zhang, Z. Lin, H. Lu, and C. M. I. Center, "Subspace clustering by mixture of gaussian regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2094–2102.

[27] C. Lu, J. Tang, M. Lin, L. Lin, S. Yan, and Z. Lin, "Correntropy induced l2 graph for robust subspace clustering," in *Proceedings of IEEE International Conference on Computer Vision*, 2013, pp. 1801–1808.

[28] W. Liu, P. P. Pokharel, and J. C. Príncipe, "Correntropy: properties and applications in non-gaussian signal processing," *Signal Processing, IEEE Transactions on*, vol. 55, no. 11, pp. 5286–5298, 2007.

[29] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[30] G. McLachlan and T. Krishnan, *The EM algorithm and extensions*. John Wiley & Sons, 2007, vol. 382.

[31] C. J. Wu, "On the convergence properties of the EM algorithm," *The Annals of statistics*, pp. 95–103, 1983.

[32] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for gaussian mixtures," *Neural Computation*, vol. 8, no. 1, pp. 129–151, 1996.

[33] D. Nettleton, "Convergence properties of the EM algorithm in constrained parameter spaces," *Canadian Journal of Statistics*, vol. 27, no. 3, pp. 639–648, 1999.

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

[35] M. R. Segal, K. D. Dahlquist, and B. R. Conklin, "Regression approaches for microarray data analysis," *Journal of Computational Biology*, vol. 10, no. 6, pp. 961–980, 2003.

[36] S. Puntanen and G. P. Styan, "Schur complements in statistics and probability," in *The Schur Complement and Its Applications*. Springer, 2005, pp. 163–226.

[37] Y. Tian and Y. Takane, "The inverse of any two-by-two nonsingular partitioned matrix and three matrix inverse completion problems," *Computers & Mathematics with Applications*, vol. 57, no. 8, pp. 1294–1304, 2009.

[38] N. ST ADLER, P. Buhlmann, and S. van de Geer, "l1-penalization for mixture regression models (with discussion)," *Test*, vol. 19, pp. 209–285, 2010.

[39] E. L. Lehmann, *Elements of large-sample theory*. Springer Science & Business Media, 1999.

[40] I. Gebru, X. Alameda-Pineda, F. Forbes, and R. Horaud, "Em algorithms for weighted-data clustering with application to audio-visual scene analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2016.

[41] C. S. Wallace and P. R. Freeman, "Estimation and inference by compact coding," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 240–265, 1987.

[42] C. S. Wallace and D. L. Dowe, "Mml clustering of multi-state, poisson, von mises circular and gaussian distributions," *Statistics and Computing*, vol. 10, no. 1, pp. 73–83, 2000.

[43] J. Rissanen, *Stochastic complexity in statistical inquiry*. World scientific, 1998, vol. 15.

[44] C. S. Wallace and D. L. Dowe, "Minimum message length and kolmogorov complexity," *The Computer Journal*, vol. 42, no. 4, pp. 270–283, 1999.

[45] D. M. Titterington, "Statistical analysis of finite mixture distributions," 1985.

[46] J. Bernardo, "Bayesian theory." *Wiley Series in Probability and Statistics. 23 cm. 586 p.*, 2000.

[47] G. Celeux, S. Chrétien, F. Forbes, and A. Mkhadri, "A component-wise em algorithm for mixtures," *Journal of Computational and Graphical Statistics*, 2012.

[48] R. Tron and R. Vidal, “A benchmark for the comparison of 3-d motion segmentation algorithms,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[49] H. Larochelle, D. Erhan, A. Courville, J. Bergstra, and Y. Bengio, “An empirical evaluation of deep architectures on problems with many factors of variation,” in *Proceedings of International Conference on Machine Learning*, 2007, pp. 473–480.

[50] A. M. Martinez, “The AR face database,” *CVC Technical Report*, vol. 24, 1998.

[51] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[52] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.