Electronic Journal of Statistics
Vol. 0 (0000)
ISSN: 1935-7524
DOI: 10.1214/154957804100000000

Model-based clustering with envelopes

Wenjing Wang, Xin Zhang and Qing Mai

Department of Statistics, Florida State University, Tallahassee, FL, 32306
e-mail: wenjing.wang@stat.fsu.edu; henry@stat.fsu.edu; mai@stat.fsu.edu

Abstract: Clustering analysis is an important unsupervised learning technique in multivariate statistics and machine learning. In this paper, we propose a set of new mixture models called CLEMM (in short for Clustering with Envelope Mixture Models) that is based on the widely used Gaussian mixture model assumptions and the nascent research area of envelope methodology. Formulated mostly for regression models, envelope methodology aims for simultaneous dimension reduction and efficient parameter estimation, and includes a very recent formulation of envelope discriminant subspace for classification and discriminant analysis. Motivated by the envelope discriminant subspace pursuit in classification, we consider parsimonious probabilistic mixture models where the cluster analysis can be improved by projecting the data onto a latent lower-dimensional subspace. The proposed CLEMM framework and the associated envelope-EM algorithms thus provide foundations for envelope methods in unsupervised and semi-supervised learning problems. Numerical studies on simulated data and two benchmark data sets show significant improvement of our propose methods over the classical methods such as Gaussian mixture models, K-means and hierarchical clustering algorithms. An R package is available at https://github.com/kusakehan/CLEMM.

Keywords and phrases: Clustering; Computational Statistics; Dimension Reduction; Envelope Methods; Gaussian Mixture Models.

1. Introduction

Cluster analysis (or clustering) is a cornerstone of multivariate statistics and unsupervised learning. The goal of clustering is to divide the observed samples into groups or clusters according to their similarity/dissimilarity (see, for example, Hartigan [22], Jain et al. [25], Kaufman and Rousseeuw [28], Chi and Lange [8] for backgrounds on clustering). Among various clustering approaches, two of the most widely used algorithms are the K-means clustering [33] and the hierarchical clustering [26]. Both methods and their variations are iterative algorithms with different convergence criteria (e.g. minimized dissimilarity within clusters) and starting points (e.g. start the algorithm by assigning all observations to one cluster or assigning each observation as its own cluster). On the other hand, to facilitate statistical interpretation and inference, clustering analysis can also be based upon probabilistic models [e.g. 4, 20, 34]. By assuming mixture distributions, clusters can be determined based on the maximum likelihood estimators in the model. See Lindsay [32] and Yao and Lindsay [44] for general backgrounds on mixture models. In this paper, we focus on the Gaussian mixture

models (GMM) because of their popularity and effectiveness in approximating (non-Gaussian) multimodal distributions (see [48, 7] for example).

Clustering algorithms suffer from the curse of dimensionality, as distance measures and parameter estimation become more challenging with increased dimensions [38]. Dimension reduction of the data thus may often improve clustering accuracy and also provide informative visualization. However, due to the unsupervised nature of clustering, many supervised dimension reduction methods for classification [e.g., 13, 40, 37, 47] or for regression [see 30, for an overview] are not directly applicable to clustering. As a widely used unsupervised dimension reduction technique, principal component analysis (PCA) is often used as a pre-processing step in clustering, especially when variables are highly correlated. For example, Ding and He [18] explains the connections between PCA and K-means clustering by showing that principal components are the continuous solutions to the cluster membership indicators in K-means clustering. In model-based clustering, there have been several proposals that share the same spirit of modeling clusters by constrained estimation in a lower-dimensional subspace. In particular, our parsimonious modeling approach is conceptually similar to the factor analysis approaches in Rubin and Thayer [35] and Baek et al. [3], but notably differs in several ways. First, nearly all the existing latent-subspace or factor analysis methods are built upon the idea that covariance matrix in each cluster has a "low-rank plus diagonal" structure, where the low-rank structure is driven by a low-dimensional latent variable. In contrast, our method is built on the envelope principle that is more flexible and general. It assumes that there exists a subspace such that observations projected onto this subspace would share a common structure that is invariant as the underlying cluster varies. In other words, data projected onto this subspace contains no information about the cluster differences and is thus immaterial to clustering. Therefore, clustering becomes more efficient if we eliminate these extraneous variations. Secondly, the subspace learning in our method is completely data-driven and integrated into the likelihood framework and EM-algorithms. The targeted dimension reduction subspace in our approach, i.e. the envelope, always exist and is a natural inferential and estimative object for dimension reduction in clustering. Finally, unlike in PCA and factor analysis, the envelope method is more adaptive and direct. The components useful for clustering are not necessarily the leading components that are identified by PCA and factor analysis. In presence of highly correlated variables, it is likely that some components with large variability are actually not useful for clustering. The envelope, on the other hand, is a more targeted dimension reduction subspace whose goal is to improve efficiency in Gaussian mixture model parameter estimation and thus to obtain better clustering result.

Our proposed Clustering with Envelope Mixture Models (CLEMM) framework advances the recent development of envelope methodology that was first proposed in the context of multivariate linear regression by Cook et al. [12] and then further developed in a series of regression problems [e.g., 39, 9, 45, 19, 29] and general multivariate parameter estimation problems [14]. See [11] for an overview on envelopes and [10] for more detailed backgrounds. Whilst all existing envelope methods concentrate on supervised learning, particularly for regres-

sion problems, our work differs obviously by tackling a unsupervised learning problem. Such an extension is far from trivial, and new techniques are required throughout its development. Moreover, our development for envelope-EM algorithms enriches the envelope computational techniques [15, 16]. The CLEMM approach, which is essentially a subspace-regularized clustering, also complements the sparse penalized solutions in the literature [e.g., 42, 6].

The rest of the article is structured as follows. In Section 2 we formally introduce the definition of CLEMM and illustrate the working mechanism of envelope in clustering. We also connect with the recent study on envelope discriminant analysis [47]. In Section 3, we derive the maximum likelihood estimators and develop the envelope-EM algorithms for CLEMM. In Section 4, we explore an important special case of CLEMM that further assumes shared covariance structure across clusters. Under this shared covariance assumption, the envelope-EM algorithm can be even faster than the standard EM estimation in Gaussian mixture models, which does not require subspace estimation. Model selection is discussed in Section 5. Numerical analysis includes simulations and two benchmark datasets are given in Section 6. Finally, Section 7 contains a summary and a short discussion on some future research directions. Proofs and technical details are given in the Appendix.

2. Models

2.1. Notation and Definitions

We first introduce the following notation and definitions to be used in this paper. For a matrix $\mathbf{B} \in \mathbb{R}^{p\times q}$, the subspace of $\mathbb{R}^p$ spanned by the columns of $\mathbf{B}$ is denoted as $\mathcal{B} = \text{span}(\mathbf{B})$. When $\mathbf{B}^T\mathbf{B}$ is positive definite, we use $\mathbf{P}_\mathbf{B} = \mathbf{P}_\mathcal{B} = \mathbf{B}(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$ to denote the orthogonal projection onto the subspace $\mathcal{B} = \text{span}(\mathbf{B})$. The orthogonal complement subspace $\mathcal{B}^\perp$ of $\mathcal{B}$ is constructed with respect to the usual inner product and that $\mathcal{B}\oplus\mathcal{B}^\perp = \mathbb{R}^p$ and $\mathcal{B}\cap\mathcal{B}^\perp = \mathbf{0}$. The projection onto $\mathcal{B}^\perp$ is then written as $\mathbf{Q}_\mathbf{B} = \mathbf{Q}_\mathcal{B} = \mathbf{I}_p - \mathbf{P}_\mathbf{B}$.

We will use the following definitions of a *reducing subspace* and an *envelope*. The definitions are equivalent to that given by Cook et al. [12] and contain constructive properties of reducing subspaces and envelopes.

Definition 1. *A subspace $\mathcal{S} \subseteq \mathbb{R}^p$ is a reducing subspace of a symmetric matrix $\mathbf{M} \in \mathbb{R}^{p\times p}$ if and only if $\mathbf{M}$ can be decomposed as $\mathbf{M} = \mathbf{P}_\mathcal{S}\mathbf{M}\mathbf{P}_\mathcal{S} + \mathbf{Q}_\mathcal{S}\mathbf{M}\mathbf{Q}_\mathcal{S}$. The $\mathbf{M}$-envelope of $\mathcal{B}$, denoted as $\mathcal{E}_\mathbf{M}(\mathcal{B})$ is the intersection of all reducing subspaces of $\mathbf{M}$ that contains $\mathcal{B}$.*

In the following sections, we provide an intuitive construct of envelope in clustering, where the matrix $\mathbf{M}$ in the above definition is replaced by the covariance of predictor $\mathbf{X} \in \mathbb{R}^p$ and the subspace $\mathcal{B}$ will be the subspace that captures the location and shape changes across clusters.

2.2. CLEMM: Clustering with Envelope Mixture Models

In a multivariate Gaussian mixture model (GMM), the observed data $\mathbf{X}_i \in \mathbb{R}^p, i = 1, \ldots, n$ are assumed to be i.i.d. following the finite mixture Gaussian distribution as

$$\mathbf{X} \sim \sum_{k=1}^{K} \pi_k N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \tag{2.1}$$

where $\pi_k \in (0, 1)$ and $\sum_{i=1}^{K} \pi_k = 1$ are the mixing weights, $N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ denotes the multivariate normal distribution with mean $\boldsymbol{\mu}_k \in \mathbb{R}^p$ and positive definite covariance matrix $\boldsymbol{\Sigma}_k$. The key to model-based clustering with GMM is to estimate the parameters $\boldsymbol{\theta} \equiv (\pi_1, \ldots, \pi_K, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$. The expectation-maximization (EM) algorithm [17] is a popular and standard approach for estimating these parameters. Specifically, the maximum likelihood estimator (MLE) for $\boldsymbol{\theta}$ is obtained by iteratively updating within the EM algorithm. We will discuss more about the EM algorithm and the estimation procedure in Section 3.

Motivated by envelope modeling techniques in regression and classification, we assume that there exists a low-dimensional subspace that fully captures the variation of data across all clusters. Let $(\boldsymbol{\Gamma}, \boldsymbol{\Gamma}_0) \in \mathbb{R}^{p\times p}$ be an orthogonal matrix, where $\boldsymbol{\Gamma} \in \mathbb{R}^{p\times u}$, $u \leq p$, is the semi-orthogonal basis for the subspace of interest. In particular, we refer to $\mathbf{X}_{\mathrm{M}} = \boldsymbol{\Gamma}^T\mathbf{X} \in \mathbb{R}^u$ as the material part of $\mathbf{X}$ – the part that contains all the information about clusters; and we refer to $\mathbf{X}_{\mathrm{IM}} = \boldsymbol{\Gamma}_0^T\mathbf{X} \in \mathbb{R}^{p-u}$ as the immaterial part of $\mathbf{X}$ – the part that is homogeneous and does not vary across clusters. Without loss of generality, we assume $\mathrm{E}(\mathbf{X}) = 0$ and propose the CLEMM as follows,

$$\mathbf{X}_{\mathrm{M}} = \boldsymbol{\Gamma}^T\mathbf{X} \sim \sum_{k=1}^{K} \pi_k N(\boldsymbol{\alpha}_k, \boldsymbol{\Omega}_k), \quad \mathbf{X}_{\mathrm{IM}} = \boldsymbol{\Gamma}_0^T\mathbf{X} \sim N(\mathbf{0}, \boldsymbol{\Omega}_0), \quad \mathbf{X}_{\mathrm{M}} \perp\!\!\!\perp \mathbf{X}_{\mathrm{IM}}, \tag{2.2}$$

where $\boldsymbol{\alpha}_k \in \mathbb{R}^u$, $\pi_k \in (0, 1)$ is defined previously, $\boldsymbol{\Omega}_k \in \mathbb{R}^{u\times u}$ and $\boldsymbol{\Omega}_0 \in \mathbb{R}^{(p-u)\times(p-u)}$ are symmetric positive definite matrices. The above model assumes that $\mathbf{X}_{\mathrm{M}}$, which follows the GMM with parameters π_k, $\boldsymbol{\alpha}_k$ and $\boldsymbol{\Omega}_k$, $k = 1, \ldots, K$, is multimodal and heterogeneous. In contrast, $\mathbf{X}_{\mathrm{IM}}$ is unimodal and follows the multivariate normal distribution. In other words, the distribution of the material part $\mathbf{X}_{\mathrm{M}}$ changes in both mean and covariance across different clusters while the immaterial part $\mathbf{X}_{\mathrm{IM}}$ does not vary. Furthermore, the last statement in (2.2) implies that the material part $\mathbf{X}_{\mathrm{M}}$ and the immaterial part $\mathbf{X}_{\mathrm{IM}}$ are independent of each other. This ensures that the immaterial part is not associated with the material part and can be eliminated completely.

To better understand the connections between CLEMM and GMM, we note that the CLEMM in (2.2) is equivalent (the proof is given in the Appendix) to

the following parsimonious parameterization in the original GMM setting,

$$\mathbf{X} \sim \sum_{k=1}^{K} \pi_k N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad \boldsymbol{\mu}_k = \boldsymbol{\Gamma}\boldsymbol{\alpha}_k, \quad \boldsymbol{\Sigma}_k = \boldsymbol{\Gamma}\boldsymbol{\Omega}_k\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T, \quad k = 1, ..., K. \tag{2.3}$$

From (2.3), it can be seen that the centers of each clusters lie within the low-dimensional subspace span($\boldsymbol{\Gamma}$), which is a reducing subspace of each $\boldsymbol{\Sigma}_k$. As a direct consequence, the marginal covariance of $\mathbf{X}$ can also be written as $\boldsymbol{\Sigma}_{\mathbf{x}} = \boldsymbol{\Gamma}\boldsymbol{\Omega}_{\mathbf{x}}\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T$, where $\boldsymbol{\Omega}_{\mathbf{x}}$ is the marginal covariance of $\mathbf{X}_{\mathrm{M}} = \boldsymbol{\Gamma}^T\mathbf{X}$. Therefore, the subspace span($\boldsymbol{\Gamma}$) is not only a reducing subspace of the intra-cluster covariance $\boldsymbol{\Sigma}_k$ but also reduces the marginal covariance $\boldsymbol{\Sigma}_{\mathbf{x}}$. These observations will help us construct CLEMM estimation: CLEMM-Shared in Section 4. The parameterization in (2.3) also links the two-part (material and immaterial parts) model in (2.2) with the original GMM in (2.1), and helps deriving the EM algorithms for CLEMM in Section 3. Similar to the envelope discriminant subspace in Zhang and Mai [47], the smallest such subspace span($\boldsymbol{\Gamma}$) is uniquely defined and always exists. We establish properties of the smallest such span($\boldsymbol{\Gamma}$) in the following section.

2.3. Envelope in clustering: a latent variable interpretation

In this section, we recast the smallest subspace span($\boldsymbol{\Gamma}$) that satisfies (2.3) as an envelope (cf. Definition 1). First, we introduce the latent variable $Y \in \{1, \ldots, K\}$ as the cluster indicator, then the GMM (2.1) can be expressed as,

$$\Pr(Y = k) = \pi_k, \quad \mathbf{X} \mid (Y = k) \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \tag{2.4}$$

where Y is latent and unobservable in clustering.

When the variable Y is observed class labels, (2.4) is commonly known as the quadratic discriminant analysis (QDA) model; and if we further assume shared covariance structure across classes, $\boldsymbol{\Sigma}_1 = \cdots = \boldsymbol{\Sigma}_K$, then (2.4) becomes the linear discriminant analysis (LDA) model. In classification, the ultimate goal is to obtain the Bayes' rule for classification defined as $\phi(\mathbf{X}) = \arg\max_{k=1,\ldots,K} \Pr(Y = k \mid \mathbf{X})$, which achieves the lowest possible error rate for any classifier (i.e. the Bayes error rate). Zhang and Mai [47] introduced the envelope discriminant subspace as the smallest subspace that is a reducing subspace of $\boldsymbol{\Sigma}_{\mathbf{x}}$ and also retains the Bayes' rule if we project the data onto it. With observable Y in (2.4), the envelope discriminant subspace leads to the same parameterization for $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ as CLEMM in (2.3), where span($\boldsymbol{\Gamma}$) is the envelope discriminant subspace. This connection leads to the following properties of CLEMM that are straightforward derivations from Proposition 3 in Zhang and Mai [47] (and hence we omitted the proof).

Let $\mathcal{L} = \operatorname{span}\{(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K - \boldsymbol{\mu}_1)\} = \operatorname{span}\{(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)\}$ (recall that we have assumed $\mathrm{E}(\mathbf{X}) = \sum_{k=1}^{K} \pi_k\boldsymbol{\mu}_k = 0$); and let $\mathcal{Q} = \operatorname{span}\{(\boldsymbol{\Sigma}_2 - \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K - \boldsymbol{\Sigma}_1)\}$. Then $\mathcal{L}$ contains the location changes across clusters and $\mathcal{Q}$ contains the spectral

changes in cluster-specific covariance matrices. The smallest subspace span($\mathbf{\Gamma}$) defined in CLEMM (2.2), or equivalently (2.3), is the envelope $\mathcal{E}_{\mathbf{\Sigma}_\mathrm{x}}(\mathcal{L}+\mathcal{Q})$. The subspace $\mathcal{S}=\mathrm{span}(\mathbf{\Gamma})$ in CLEMM (2.3) can be defined equivalently using the following coordinate-free statements,

$$\mathcal{L}+\mathcal{Q}\subseteq\mathcal{S}, \mathbf{\Sigma}_\mathrm{x}=\mathbf{P}_\mathcal{S}\mathbf{\Sigma}_\mathrm{x}\mathbf{P}_\mathcal{S}+\mathbf{Q}_\mathcal{S}\mathbf{\Sigma}_\mathrm{x}\mathbf{Q}_\mathcal{S}. \tag{2.5}$$

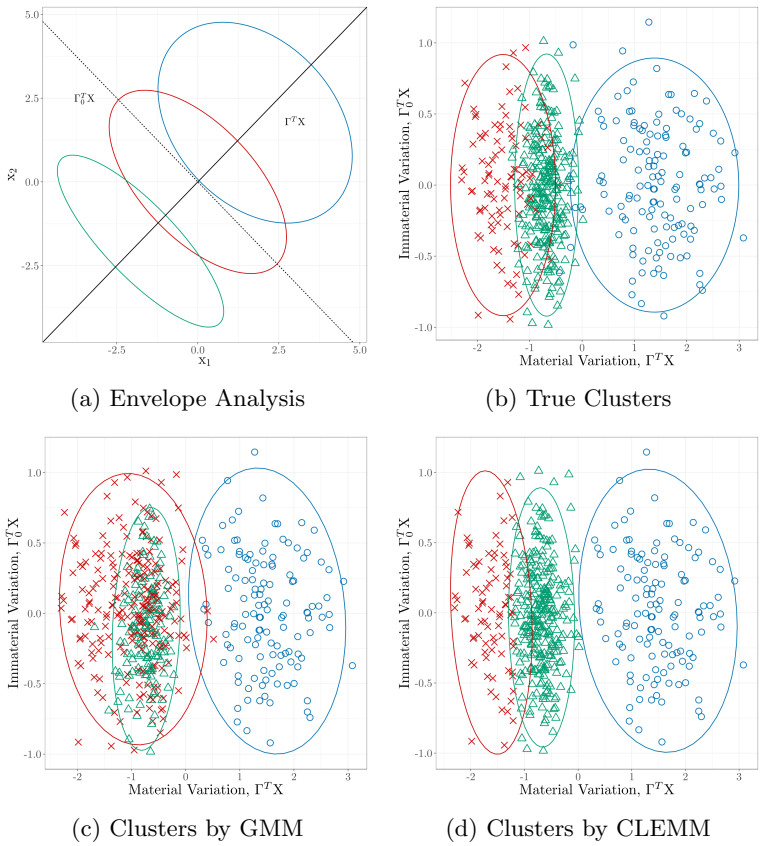
The intersection of any two subspaces that satisfy (2.5) is a subspace that still satisfies (2.5). Therefore, the intersection of all such subspaces is uniquely defined, minimal dimensional and satisfies (2.5). It is in fact the $\mathbf{\Sigma}_\mathrm{x}$-envelope of $\mathcal{L}+\mathcal{Q}$, denoted as $\mathcal{E}_{\mathbf{\Sigma}_\mathrm{x}}(\mathcal{L}+\mathcal{Q})$. With a bit abuse of notation, we henceforth use $\mathbf{\Gamma}\in\mathbb{R}^{p\times u}$ as a semi-orthogonal basis for the envelope $\mathcal{E}_{\mathbf{\Sigma}_\mathrm{x}}(\mathcal{L}+\mathcal{Q})$, which has dimension u.

2.4. Working mechanism of CLEMM

Clearly, CLEMM can help reduce the total number of free parameters. For GMM, there are $(K-1)p+Kp(p+1)/2$ free parameters in $\{\boldsymbol{\mu}_k,\mathbf{\Sigma}_k\}$, $k=1,\ldots,K$, where the factor $(K-1)$ is due to that we have assume $\mathrm{E}(\mathbf{X})=0$. For CLEMM (2.3), the number of free parameters in $\{\mathbf{\Gamma},\boldsymbol{\alpha}_k,\mathbf{\Omega}_k,\mathbf{\Omega}_0\}$ is $(p-u)u+(K-1)u+Ku(u+1)/2+(p-u)(p-u+1)/2$. The total reduction in the number of free parameters is thus $(K-1)[(p-u)+\{p(p+1)-u(u+1)\}/2]$, where $[(p-u)+\{p(p+1)-u(u+1)\}/2]$ is the difference between the mixture distribution of full data $\mathbf{X}\sim\sum_{k=1}^K\pi_kN(\boldsymbol{\mu}_k,\mathbf{\Sigma}_k)$ and the mixture distribution of the material part of data $\mathbf{X}_\mathrm{M}=\mathbf{\Gamma}^T\mathbf{X}\sim\sum_{k=1}^K\pi_kN(\boldsymbol{\alpha}_k,\mathbf{\Omega}_k)$. By reducing the number of free parameters and thus the model complexity, the CLEMM parameterization leads to potential efficiency gain in parameter estimation with EM algorithms. To provide more intuition about the working mechanism of CLEMM and its potential advantages over the classical GMM, we next consider some visualizations and an illustrative simulation example.

Figure 1a is a schematic plot of the envelope on a bivariate tri-cluster data. Specifically, $\mathbf{X}=(X_1,X_2)^T\sim\sum_{k=1}^3\pi_kN(\boldsymbol{\mu}_k,\mathbf{\Sigma}_k)$ and the envelope dimension $u=1$ where $\mathbf{\Gamma}=(1,1)^T/\sqrt{2}$ and $\mathbf{\Gamma}_0=(1,-1)^T/\sqrt{2}$. From this plot, we see clearly that the centers of clusters varies along the envelope direction, and that the heteroscedasticity is also captured by this direction. On the other hand, if we project the data onto $\mathbf{\Gamma}_0$, the three clusters become one. By eliminating the immaterial variation from $\mathbf{\Gamma}_0^T\mathbf{X}$, or equivalently, by projecting the data onto $\mathbf{\Gamma}$, we expect a substantial improvement in distinguishing the three clusters.

To further verify the actual efficiency gain by CLEMM, we consider a simulation model (M1) in our numerical studies (see Section 6 for more details), where $\mathbf{X}\sim\sum_{k=1}^3\pi_kN(\boldsymbol{\mu}_k,\mathbf{\Sigma}_k)$ has $p=15$ variables and three clusters of relative sizes $(\pi_1,\pi_2,\pi_3)=(0.3,0.2,0.5)$. The envelope has dimension $u=1$ and each $\mathbf{\Sigma}_k$ has a relatively complicated format such that the predictors are all highly correlated with each other. Figure 1b plots the simulated data after being projected onto a two-dimensional plane consists of the true envelope and an arbitrary direction



(a) Envelope Analysis (b) True Clusters

(c) Clusters by GMM (d) Clusters by CLEMM

Fig 1: CLEMM working mechanism. Figure (a) and (b) are the true clusters and the true distributions of the data. Figure (c) shows the clustering result by GMM and Figure (d) illustrates the clustering result of CLEMM. The ellipses in each plots represent the true or the estimated multivariate normal distributions.

from the orthogonal complement of the envelope. Clearly, we see the distinctions among the three clusters lie within the envelope (horizontal axis), while the distributions are the same along the immaterial direction (vertical axis). Figure 1c and Figure 1d show that actual estimated results from the classical GMM and our proposed CLEMM. The parameter estimation of $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are reflected by the three ellipses in the plot. Compared to the true distribution (eclipses) in Figure 1b, CLEMM clearly improves the parameter estimation substantially. Not surprisingly, if we compare the cluster labels by GMM and by CLEMM, the mis-clustering error rate is also reduced drastically by CLEMM.

2.5. Connections with factor analyzers approaches

Baek et al. [3] proposed the Mixture of Common Factor Analyzers (MCFA) model, which is a popular approach in the factor analysis-type clustering methods and is thus included as a competitor in our numerical studies (Section 6). Using our notation, the MCFA model can be summarized as,

$$\mathbf{X} \sim \sum_{k=1}^{K} \pi_k N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad \boldsymbol{\mu}_k = \boldsymbol{\Gamma}\boldsymbol{\alpha}_k, \quad \boldsymbol{\Sigma}_k = \boldsymbol{\Gamma}\boldsymbol{\Phi}_k\boldsymbol{\Gamma}^T + \mathbf{D}, \quad k = 1, \ldots, K, \tag{2.6}$$

where $\mathbf{D}$ is a $p \times p$ diagonal matrix (e.g. $\mathbf{D} = \sigma^2\mathbf{I}_p$). Therefore, this model can be viewed as a special case of our CLEMM model (2.3). If we let $\boldsymbol{\Omega}_k = \boldsymbol{\Phi}_k + \sigma^2\mathbf{I}_u$ and $\boldsymbol{\Omega}_0 = \sigma^2\mathbf{I}_{p-u}$, then (2.3) reduces to (2.6). It assumes that the common covariance is isotropic and also restricts the shared subspace to contain the leading eigenvalues, since $\boldsymbol{\Omega}_k = \boldsymbol{\Phi}_k + \sigma^2\mathbf{I}_u$ now has larger eigenvalues than $\boldsymbol{\Omega}_0 = \sigma^2\mathbf{I}_{p-u}$. The CLEMM approach is therefore more flexible than the factor analyzer approaches. The flexibility of CLEMM, however, leads to a more complicated EM algorithm as we carefully derives in the following section.

3. Estimation

3.1. A brief review of the EM algorithm for GMM

Dempster et al. [17] introduced the EM algorithm which later become the most popular technique to solve GMM. In this section, we first give a brief review of the EM algorithm for GMM. To make our envelope-EM algorithm easier to comprehend, we present it in a way that is parallel to the classical EM algorithm for fitting GMM.

By introducing a latent variable Y, the GMM can be written as (2.4), $\Pr(Y = k) = \pi_k$, $\mathbf{X} \mid (Y = k) \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, $k = 1, \ldots, K$. The log-likelihood of the "observed data" $\{\mathbf{X}_i\}_{i=1}^n$ can be written as $\ell_o(\boldsymbol{\theta}) = \sum_{i=1}^n \log\{\sum_{k=1}^K \pi_k \phi(\mathbf{X}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}$, where $\phi(\mathbf{X}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is the density function of $N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ and $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_K, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$ is the set of all parameters. Directly solving this log-likelihood is difficult and the EM algorithm iteratively updates the estimator by treating Y_i as missing data. Let $\{\mathbf{X}_i, Y_i\}_{i=1}^n$ be the "complete data", where Y_i is unobserved. Then we have the complete data log-likelihood $\ell_c(\boldsymbol{\theta}) = \sum_{i=1}^n \sum_{k=1}^K y_{ik} \log\{\pi_k \phi(\mathbf{X}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}$, where $y_{ik} = 1$ if the i-th observation belongs to the k-th cluster and 0 otherwise. The EM algorithm then estimates $\boldsymbol{\theta}$ by iteratively maximizing the conditional expectation of the complete log-likelihood on the observed data. The EM algorithm for GMM is summarized as follows.

Initialization: Choose an initial value $\boldsymbol{\theta}^{(0)}$ and set iteration number $m = 0$. We can simply choose the clustering result from K-means and hierarchical clustering as starting value. See [27] for more discussion on the choice of initial values for GMM.

Iterating over the E-step and the M-step below to generate a sequence of estimators $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \ldots$.

E-step: Compute the expectation $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) = \mathrm{E}\{\ell_c(\boldsymbol{\theta}) \mid \boldsymbol{\theta}^{(m)}, \mathbf{X}_i, i = 1, 2, \ldots, n\}$ for $m = 1, 2 \ldots$, which is equivalent to

$$Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) \simeq \sum_{i=1}^{n}\sum_{k=1}^{K} \eta_{ik}^{(m)}\{\log \pi_k - \frac{1}{2}\log|\boldsymbol{\Sigma}_k| - \frac{1}{2}(\mathbf{X}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\mathbf{X}_i - \boldsymbol{\mu}_k)\},$$

where $\eta_{ik}^{(m)} = \Pr(Y_i = k \mid \mathbf{X} = \mathbf{X}_i, \boldsymbol{\theta}^{(m)})$ is the membership weight for each data point and it satisfies $\sum_{k=1}^{K}\eta_{ik}^{(m)} = 1$, and the symbol "$\simeq$" means equal up to an additive constant. Specifically, we have,

$$\eta_{ik}^{(m)} = \frac{\pi_k^{(m)} f_k(\mathbf{X}_i \mid \boldsymbol{\mu}_k^{(m)}, \boldsymbol{\Sigma}_k^{(m)})}{\sum_{k=1}^{K}\pi_k^{(m)} f_k(\mathbf{X}_i \mid \boldsymbol{\mu}_k^{(m)}, \boldsymbol{\Sigma}_k^{(m)})}. \tag{3.1}$$

M-step: Solve the optimization $\boldsymbol{\theta}^{(m+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)})$, which leads to the maximizers

$$\pi_k^{(m+1)} = \frac{\sum_{i=1}^{n}\eta_{ik}^{(m)}}{n}, \tag{3.2}$$

$$\boldsymbol{\mu}_k^{(m+1)} = \frac{\sum_{i=1}^{n}\eta_{ik}^{(m)}\mathbf{X}_i}{\sum_{i=1}^{n}\eta_{ik}^{(m)}}, \tag{3.3}$$

$$\boldsymbol{\Sigma}_k^{(m+1)} = \frac{\sum_{i=1}^{n}\eta_{ik}^{(m)}(\mathbf{X}_i - \boldsymbol{\mu}_k^{(m+1)})(\mathbf{X}_i - \boldsymbol{\mu}_k^{(m+1)})^T}{\sum_{i=1}^{n}\eta_{ik}^{(m)}}. \tag{3.4}$$

It is well-established in the statistical literature that the EM algorithm is guaranteed to converge monotonically to a local maximum of the log-likelihood under mild conditions [17, 43, 5].

3.2. Envelope-EM algorithm for CLEMM

In this section, we develop the envelope-EM algorithm for estimating the CLEMM parameters. The estimation problem in CLEMM is far more complicated than fitting GMM or the envelope estimation in regression or classification. First of all, we introduce the latent variable Y into the CLEMM assumptions (2.2),

$$\Pr(Y = k) = \pi_k, \quad \boldsymbol{\Gamma}^T\mathbf{X} \mid (Y = k) \sim N(\boldsymbol{\alpha}_k, \boldsymbol{\Omega}_k), \quad \boldsymbol{\Gamma}_0^T\mathbf{X} \mid (Y = k) \sim N(0, \boldsymbol{\Omega}_0), \tag{3.5}$$

with $\boldsymbol{\Gamma}_0^T\mathbf{X} \perp\!\!\!\perp \boldsymbol{\Gamma}^T\mathbf{X}$. If we know $\boldsymbol{\Gamma}$, then the EM algorithm for CLEMM is a straightforward extension of the EM algorithm for GMM. Unfortunately, $\boldsymbol{\Gamma}$ is unknown and its estimation involves solving an non-convex objective function on Grassmann manifold. Therefore, we have to efficiently integrate the optimization for $\boldsymbol{\Gamma}$ into the EM algorithm.

To distinguish the parameters in CLEMM and GMM, we define the set of unique parameters in CLEMM to be $\boldsymbol{\phi} = (\pi_1, \ldots, \pi_K, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K, \boldsymbol{\Omega}_1, \ldots, \boldsymbol{\Omega}_K, \boldsymbol{\Omega}_0, \boldsymbol{\Gamma})$. Then the GMM parameter $\boldsymbol{\theta}$ is an estimable function of $\boldsymbol{\phi}$,

$$\boldsymbol{\theta} = \boldsymbol{\theta}(\boldsymbol{\phi}), \quad \boldsymbol{\mu}_k(\phi) = \boldsymbol{\Gamma}\boldsymbol{\alpha}_k, \quad \boldsymbol{\Sigma}_k(\boldsymbol{\phi}) = \boldsymbol{\Gamma}\boldsymbol{\Omega}_k\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T, \quad k = 1, \ldots, K. \tag{3.6}$$

To further distinguish the estimators, the CLEMM estimators are denoted as $\widehat{\boldsymbol{\phi}}^{(m)}$ and $\widehat{\boldsymbol{\theta}}^{(m)} \equiv \boldsymbol{\theta}(\widehat{\boldsymbol{\phi}}^{(m)})$ at the m-th iteration.

For our envelope-EM algorithm and in all of our numerical studies, we use the same initialization as GMM. The E-step of envelope-EM is the same as the usual EM for GMM, except that we need to replace the GMM estimator $\boldsymbol{\theta}^{(m)}$ with the CLEMM estimator $\widehat{\boldsymbol{\theta}}^{(m)} = \boldsymbol{\theta}(\widehat{\boldsymbol{\phi}}^{(m)})$. The key step is the maximization of $Q(\boldsymbol{\theta}(\phi) \mid \widehat{\boldsymbol{\theta}}^{(m)})$, which is now an over-parameterized function under the CLEMM parameterization (2.3). Straightforward calculation shows that

$$\begin{aligned} Q(\boldsymbol{\theta}(\boldsymbol{\phi}) \mid \widehat{\boldsymbol{\theta}}^{(m)}) \;\simeq\; & \sum_{i=1}^{n}\sum_{k=1}^{K} \widehat{\eta}_{ik}^{(m)} \{\log\pi_k - \frac{1}{2}(\log|\boldsymbol{\Omega}_k| + \log|\boldsymbol{\Omega}_0|) \\ - \;& \frac{1}{2}(\mathbf{X}_i - \boldsymbol{\Gamma}\boldsymbol{\alpha}_k)^T(\boldsymbol{\Gamma}\boldsymbol{\Omega}_k^{-1}\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0^{-1}\boldsymbol{\Gamma}_0^T)(\mathbf{X}_i - \boldsymbol{\Gamma}\boldsymbol{\alpha}_k)\}, \end{aligned}$$

where $\widehat{\eta}_{ik}^{(m)} = \Pr(Y_i = k \mid \mathbf{X} = \mathbf{X}_i, \widehat{\boldsymbol{\theta}}^{(m)})$ is the membership weights for each point. We carefully derived the maximizer for the above Q-function under the CLEMM constraints. The results are summarized in the following proposition. We define the following quantities that are intermediate estimators for the envelope-EM algorithm,

$$\begin{aligned} \widetilde{\boldsymbol{\mu}}_k^{(m)} &= \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}\mathbf{X}_i}{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}, \quad \mathbf{S}_{\mathbf{x}} = \frac{\sum_{i=1}^n (\mathbf{X}_i - \overline{\mathbf{X}})(\mathbf{X}_i - \overline{\mathbf{X}})^T}{n}, \\ \mathbf{S}_k^{(m)} &= \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}(\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})(\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})^T}{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}, \quad \overline{\mathbf{S}}^{(m)} = \sum_{k=1}^K \widehat{\pi}_k^{(m)}\mathbf{S}_k^{(m)}. \end{aligned}$$

The mean $\widetilde{\boldsymbol{\mu}}_k^{(m)}$ and covariance $\mathbf{S}_k^{(m)}$ take the same forms as in the GMM parameter estimation (3.3) and (3.4). The only difference is that the membership weights $\widehat{\eta}_{ik}$ are now based on CLEMM estimator instead of the GMM estimator.

Proposition 1. *Given $\widehat{\eta}_{ik}^{(m)}$ (e.g. computed in the E-step of Algorithm 1), $i = 1, \ldots, n$, $k = 1, \ldots, K$, the CLEMM estimators from maximizing* (3.7) *are,*

$$\widehat{\boldsymbol{\alpha}}_k = \widehat{\boldsymbol{\Gamma}}^T\widetilde{\boldsymbol{\mu}}_k^{(m)}, \;\; \widehat{\boldsymbol{\Omega}}_k = \widehat{\boldsymbol{\Gamma}}^T\mathbf{S}_k^{(m)}\widehat{\boldsymbol{\Gamma}}, \;\; \widehat{\boldsymbol{\Omega}}_0 = \widehat{\boldsymbol{\Gamma}}_0^T\mathbf{S}_{\mathbf{x}}\widehat{\boldsymbol{\Gamma}}_0,$$

where $\widehat{\pi}_k^{(m+1)} = \sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}/n$ and $\widehat{\boldsymbol{\Gamma}} \in \mathbb{R}^{p\times u}$ is the minimizer of the following objective function under the semi-orthogonal constraint $\boldsymbol{\Gamma}^T\boldsymbol{\Gamma} = \mathbf{I}_u$,

$$G^{(m)}(\boldsymbol{\Gamma}) = \log|\boldsymbol{\Gamma}^T\mathbf{S}_{\mathbf{x}}^{-1}\boldsymbol{\Gamma}| + \sum_{k=1}^K \widehat{\pi}_k^{(m+1)} \log|\boldsymbol{\Gamma}^T\mathbf{S}_k^{(m)}\boldsymbol{\Gamma}|. \tag{3.7}$$

Algorithm 1 Envelope-EM algorithm for CLEMM

1: Initialize $\widehat{\pi}_k^{(0)}, \widehat{\boldsymbol{\mu}}_k^{(0)}, \widehat{\boldsymbol{\Sigma}}_k^{(0)}$ for $k = 1, \ldots, K$, set $m = 0$.
2: **E**-step: For $k = 1, \ldots, K$, $i = 1, \ldots, n$, calculate $\widehat{\eta}_{ik}^{(m)}$ by replacing $\boldsymbol{\theta}^{(m)}$ with the CLEMM estimators $\widehat{\boldsymbol{\theta}}^{(m)} = \boldsymbol{\theta}(\widehat{\boldsymbol{\phi}}^{(m)})$ in (3.1).
3: **M**-step:

- Calculate $\widetilde{\boldsymbol{\mu}}_k^{(m)} = \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)} \mathbf{X}_i}{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}$ and $\widehat{\pi}_k^{(m+1)} = \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}{n}$ for $k = 1, 2, ..., K$.
- Solve for $\widehat{\boldsymbol{\Gamma}} = \arg\min_{\boldsymbol{\Gamma} \in \mathcal{G}(p,u)} \{\sum_{k=1}^K \widehat{\pi}_k^{(m+1)} \log|\boldsymbol{\Gamma}^T \mathbf{S}_k^{(m)} \boldsymbol{\Gamma}| + \log|\boldsymbol{\Gamma}^T \mathbf{S}_{\mathrm{x}}^{-1} \boldsymbol{\Gamma}|\}$.
- For $k = 1, 2, ..., K$, update parameter estimates as

$$\widehat{\boldsymbol{\mu}}_k^{(m+1)} = \widehat{\boldsymbol{\Gamma}}\widehat{\boldsymbol{\Gamma}}^T \widetilde{\boldsymbol{\mu}}_k^{(m)}, \quad \widehat{\boldsymbol{\Sigma}}_k^{(m+1)} = \widehat{\boldsymbol{\Gamma}}\widehat{\boldsymbol{\Gamma}}^T \mathbf{S}_k^{(m)} \widehat{\boldsymbol{\Gamma}}\widehat{\boldsymbol{\Gamma}}^T + \widehat{\boldsymbol{\Gamma}}_0 \widehat{\boldsymbol{\Gamma}}_0^T \mathbf{S}_{\mathrm{x}} \widehat{\boldsymbol{\Gamma}}_0 \widehat{\boldsymbol{\Gamma}}_0^T. \tag{3.8}$$

4: Iterate over Steps 2 and 3 until convergence.

The objective function (3.7) for estimating the envelope is very similar to that of the envelope QDA model [47]. In discriminant analysis, where Y_i's are fully observed, the calculation can be simplified by replacing $\widehat{\pi}_k^{(m+1)}$ with the observed class size n_k/n, where n_k is the number of observations in class k, and similarly by replacing $\mathbf{S}_k^{(m)}$ with the within-class sample covariance. Then (3.7) reduces to the likelihood-based objective function for envelope QDA model. This shows the intrinsic connections between model-based clustering and discriminant analysis.

Based on the results in Proposition 1, we now can summarize the envelope-EM algorithm in Algorithm 1. In each iteration of M-step, we see that the CLEMM estimators $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are coordinate-free since they depend on $\widehat{\boldsymbol{\Gamma}}$ only through $\mathrm{span}(\widehat{\boldsymbol{\Gamma}})$ (i.e. only through the projection matrices $\widehat{\boldsymbol{\Gamma}}\widehat{\boldsymbol{\Gamma}}^T$ and $\widehat{\boldsymbol{\Gamma}}_0\widehat{\boldsymbol{\Gamma}}_0^T$). Therefore, the optimization of $\widehat{\boldsymbol{\Gamma}}$ involved in the CLEMM estimation is defined on the set of all u-dimensional linear subspaces of $\mathbb{R}^p$, which is known as the Grassmann manifold and denoted as $\mathcal{G}_{p,u}$. We discuss such optimization in the next section.

3.3. Envelope subspace estimation

The constrained minimization of the objective function $G^{(m)}(\boldsymbol{\Gamma})$ can be done through gradient descent on a Grassmann manifold. In particular, we have the following closed-form expression for the gradient of $G^{(m)}(\boldsymbol{\Gamma})$ ignoring the orthogonality constraints $\boldsymbol{\Gamma}^T\boldsymbol{\Gamma} = \mathbf{I}_u$,

$$\frac{\mathrm{d}G^{(m)}(\boldsymbol{\Gamma})}{\mathrm{d}\boldsymbol{\Gamma}} = 2\,\mathbf{S}_{\mathrm{x}}^{-1}\boldsymbol{\Gamma}\left(\boldsymbol{\Gamma}^T\mathbf{S}_{\mathrm{x}}^{-1}\boldsymbol{\Gamma}\right)^{-1} + 2\sum_{k=1}^K \widehat{\pi}_k^{(m+1)}\mathbf{S}_k^{(m)}\boldsymbol{\Gamma}\left(\boldsymbol{\Gamma}^{\mathsf{T}}\mathbf{S}_k^{(m)}\boldsymbol{\Gamma}\right)^{-1}. \tag{3.9}$$

Many manifold optimization packages only requires the above "unconstrained" matrix derivative, where the geometric constraints are taken into account by different techniques. For example, our implementation (more details can be found in the Appendix) uses the curvilinear search algorithm proposed by [41]; and

manifold gradient descent methods [1] update along the tangent space of the manifold and then projecting back onto the manifold at each iteration.

However direct optimization is computationally expensive and requires good starting values. We adopt the idea of the 1D algorithm from [15] that is fast and stable for obtaining a good initial estimator of envelopes in linear models [e.g., 14, 31]. For our problem, we propose the following modified 1D algorithm to obtain an initial estimator of minimizing $\arg\min_{\mathbf{\Gamma}\in\mathcal{G}(p,u)} G^{(m)}(\mathbf{\Gamma})$. We start with $\mathbf{g}_0 = \mathbf{G}_0 = 0$ and $\mathbf{G}_{00} = \mathbf{I}_p$, let $\mathbf{G}_l = (\mathbf{g}_1, \ldots, \mathbf{g}_l)$ denote the sequential directions obtained and let $(\mathbf{G}_l, \mathbf{G}_{0l})$ be an orthogonal matrix. Then for $l = 0, \ldots, u-1$, we obtain the sequential directions as follows.

- Calculate $\mathbf{U}_l = \mathbf{G}_{0l}^T \mathbf{S}_{\mathbf{x}} \mathbf{G}_{0l}$ and $\mathbf{V}_{lk} = \mathbf{G}_{0l}^T \mathbf{S}_k^{(m)} \mathbf{G}_{0l}$, and define the following stepwise objective function for $\mathbf{w} \in \mathbb{R}^{p-l+1}$,

$$f_l(\mathbf{w}) = \log(\mathbf{w}^T \mathbf{U}_l^{-1} \mathbf{w}) + \sum_{k=1}^{K} \widehat{\pi}_k^{(m+1)} \log(\mathbf{w}^T \mathbf{V}_{lk} \mathbf{w}). \tag{3.10}$$

- Solve $\mathbf{w}_{l+1} = \arg\min_{\mathbf{w}\in\mathbb{R}^{p-l}} f_l(\mathbf{w})$ under constraint $\mathbf{w}^T\mathbf{w} = 1$.
- Set $\mathbf{g}_{l+1} = \mathbf{G}_{0l}\mathbf{w}_{l+1}$ as the $(l+1)$-th direction of the envelope.

After the above sequential steps, we obtain an initial estimator $\mathbf{G}_u \in \mathbb{R}^{p\times u}$ for the optimization of $G^{(m)}(\mathbf{\Gamma})$ using its gradient (3.9). It is worth mentioning that in the real data applications, we are likely to have the within-class sample covariance matrix $\mathbf{S}_k^{(m)}$ to be very close to singular, if the probability π_k is very small. The singularity could lead to the unstable optimization of $G^{(m)}(\mathbf{\Gamma})$. To our experience, the estimation often improves (in stability and sometimes speed) by adding a small diagonal matrix such as $0.01\mathbf{I}_p$ to the sample covariance estimate $\mathbf{S}_k^{(m)}$ of clusters k when this cluster has relatively small size.

4. CLEMM-Shared: A special case of CLEMM

Recall that the number of free parameters in GMM is of the order $O(Kp^2)$, which can be much bigger than the number $O(p^2 + Ku^2)$ for CLEMM. When the dimension p is moderately high, GMM fitting becomes ineffective or even problematic. As we have seen in our real data analysis, even when the true clusters exhibit different covariance structures, it is often beneficial to fit a more restrictive GMM by assuming $\mathbf{\Sigma}_1 = \cdots = \mathbf{\Sigma}_K = \mathbf{\Sigma}$ to reduce the number of parameters in estimation. More specifically, the number of free parameters in GMM and CLEMM are reduced to order $O(p^2)$ under such assumptions. In this section, we consider the special case of CLEMM under the shared covariance assumption that $\mathbf{\Sigma}_1 = \cdots = \mathbf{\Sigma}_K = \mathbf{\Sigma}$, that is,

$$\mathbf{\Gamma}^T\mathbf{X} \sim \sum_{k=1}^{K} \pi_k N(\boldsymbol{\alpha}_k, \mathbf{\Omega}), \quad \mathbf{\Gamma}_0^T\mathbf{X} \sim N(\mathbf{0}, \mathbf{\Omega}_0), \quad \mathbf{\Gamma}^T\mathbf{X} \perp\!\!\!\perp \mathbf{\Gamma}_0^T\mathbf{X}, \tag{4.1}$$

where $\mathbf{\Omega} \in \mathbb{R}^{u\times u}$ is a symmetric positive definite matrix that remains the same across all clusters.

Under this shared-covariance CLEMM model (4.1), the clusters share the same shape and are only distinguishable by their centroids. We will refer to this model as CLEMM-shared (and its counterpart GMM-shared) throughout our discussion. In terms of envelope, only the subspace $\mathcal{L} = \text{span}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K - \boldsymbol{\mu}_1)$ is relevant and the envelope degenerates to $\mathcal{E}_{\boldsymbol{\Sigma}_x}(\mathcal{L})$, which is the smallest subspace $\text{span}(\boldsymbol{\Gamma})$ that satisfies (4.1). In the case of shared covariance GMM model, the total number of free parameters is reduced by $(K-1)(p-u)$ when we introduce the envelope structure and assume that the mean differences across clusters lie within a low-dimensional subspace. Similar to the general case, CLEMM-shared performs dimension reduction and efficient parameter estimation under the more restrictive shared covariance GMM.

Another more practical benefit of model (4.1) is that the envelope-EM algorithm for CLEMM-shared can be further simplified and sped up over Algorithm 1. In fact, by utilizing a special form of Grassmann manifold optimization for shared covariance, we can accelerate the convergence of envelope-EM algorithm to be even faster than the standard GMM with shared covariance. The computational cost comparisons can be found in the Appendix E.2. We see that the CLEMM is generally slower than GMM because of the Grassmann manifold optimization involved, but the CLEMM-shared estimation can be faster than the GMM-shared estimation.

The envelope-EM algorithm for CLEMM-shared is analogous to Algorithm 1. We thus omit the details and only summarize the different M-step in the following proposition. The full description of the algorithm is given in Appendix D, Algorithm 2. Define the shared covariance estimator as

$$\mathbf{S}^{(m)} = n^{-1} \sum_{i=1}^{n} \sum_{k=1}^{K} \widehat{\eta}_{ik}^{(m)} (\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})(\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})^T.$$

Proposition 2. *Given $\widehat{\eta}_{ik}^{(m)}$ (computed in the E-step in Algorithm 2), $i = 1, \ldots, n$, $k = 1, \ldots, K$, the maximum likelihood estimators of CLEMM-shared parameters are as follows,*

$$\widehat{\boldsymbol{\alpha}}_k = \widehat{\boldsymbol{\Gamma}}^T \widetilde{\boldsymbol{\mu}}_k^{(m)}, \quad k = 1, \ldots K, \quad \widehat{\boldsymbol{\Omega}} = \widehat{\boldsymbol{\Gamma}}^T \mathbf{S}^{(m)} \widehat{\boldsymbol{\Gamma}}, \quad \widehat{\boldsymbol{\Omega}}_0 = \widehat{\boldsymbol{\Gamma}}_0^T \mathbf{S}_{\mathbf{x}} \widehat{\boldsymbol{\Gamma}}_0,$$

where $\widehat{\boldsymbol{\Gamma}} \in \mathbb{R}^{p \times u}$ is the minimizer of following objective function subject to $\boldsymbol{\Gamma}^T \boldsymbol{\Gamma} = \mathbf{I}_u$,

$$F^{(m)}(\boldsymbol{\Gamma}) = \log |\boldsymbol{\Gamma}^T \mathbf{S}^{(m)} \boldsymbol{\Gamma}| + \log |\boldsymbol{\Gamma}^T \mathbf{S}_{\mathbf{x}}^{-1} \boldsymbol{\Gamma}|. \tag{4.2}$$

From Proposition 2, we see the two major differences between CLEMM and CLEMM-shared lie in the shared covariances $\{\boldsymbol{\Omega}_k\}_{k=1}^K$ versus $\boldsymbol{\Omega}$, and in the objective function for solving for envelope. Since there are only two matrices in $F^{(m)}(\boldsymbol{\Gamma})$, we can adopt the envelope coordinate descent (ECD) algorithm recently proposed by [16]. The ECD algorithm has shown to be much faster than the 1D algorithm and full Grassmann manifold updates without much loss of accuracy. Therefore, we are able to improve the computation and speed up the envelope-EM algorithm for CLEMM-shared.

5. Model Selection

Information criteria are often used to determine the number of clusters in GMM. Examples include Akaike Information Criterion (AIC, Akaike [2]), Bayesian Information Criterion (BIC, Schwarz [36]), Approximate Weight of Evidence Criterion (AWE, Banfield and Raftery [4]), among others. We adopt the AWE criterion to select the number of clusters, which suggests an approximate Bayesian solution to choose the number of clusters based on the "complete" log-likelihood $\ell_c(\boldsymbol{\theta})$,

$$\text{AWE}(K) = -2 \cdot \ell_c(\boldsymbol{\theta}) + 2 \cdot df(K) \cdot (3/2 + \log n), \tag{5.1}$$

where $df(K)$ is the number of free parameters that we have discussed earlier. Minimizing AWE(K) over $K \in \{1, 2, \dots\}$ gives us the estimated number of clusters.

After determining K, which is the same for both GMM and CLEMM, we then determine the envelope dimension $u \in \{1, \dots, p\}$ as follows. For a given K, we replace $-2\ell_c(\boldsymbol{\theta})$ in (5.1) with the objective function value $nG^{(m)}(\widehat{\boldsymbol{\Gamma}})$ in Proposition 1. This is because the objective function $G^{(m)}(\widehat{\boldsymbol{\Gamma}})$ are essentially the partially minimized negative log-likelihood (see Appendix B for the derivations). As such, the AWE criterion turns into a function of the envelope subspace dimension u,

$$\text{AWE}(u) = n \cdot G^{(m)}(\widehat{\boldsymbol{\Gamma}}) + 2 \cdot df(u) \cdot (3/2 + \log n). \tag{5.2}$$

For the special case of CLEMM-shared, we replace $G^{(m)}(\widehat{\boldsymbol{\Gamma}})$ with $F^{(m)}(\widehat{\boldsymbol{\Gamma}})$ from Proposition 2. This type of envelope objective functions, $G^{(m)}(\widehat{\boldsymbol{\Gamma}})$ and $F^{(m)}(\widehat{\boldsymbol{\Gamma}})$, can also be interpreted as the quasi-likelihood for model-free envelope estimation and consistent dimension selection [46]. This modified AWE criterion has very promising performances in our simulations and real data applications.

6. Numerical Studies

6.1. Simulations

In this section, we empirically compare the clustering results of GMM and CLEMM. We also include K-means, Hierarchical Clustering (HC), and Bayes' error of classification as benchmarks. The Bayes' error is the lowest possible classification error estimated from the true parameters and using the class labels. The built-in functions in R are used for K-means (`kmeans`) and HC (`hclust`) algorithms. We also included the mixtures of factor analyzers with common component-factor loadings method [3, MCFA], where we set the number of component-factors the same as the envelope dimension. In addition, we include the results of LDA and QDA to compare with supervised learning methods. For LDA and QDA, we generate an independent testing data set with the same sample size as the training data set and report the classification error rates on the testing data set.

To evaluate the clustering result, we assign observations to a cluster by plugging-in the estimated parameters to the Bayes' rule. Then the optimal clustering is defined by the clustering error rate [6]: $\min_{\pi\in\mathcal{P}} \mathrm{E}\{I(\phi(\mathbf{X}) \neq \pi(Y))\}$ where $\mathcal{P} = \{\pi : [1, ..., K] \to [1, ..., K]\}$ is the set of permutation function and Y is the latent label of observation $\mathbf{X}$, and $\phi(\mathbf{X})$ being the Bayes' rule of classification.

We focus on the relative improvements of CLEMM over GMM in terms of parameter estimation and also mis-clustering error rate. It is expected that the parameter estimation from the EM algorithm and the envelope-EM algorithm depends on the initialization. In order to have reasonably good initialization, we consider the following initial values of $\widehat{\pi}_k^{(0)}$, $\widehat{\boldsymbol{\mu}}_k^{(0)}$ and $\widehat{\boldsymbol{\Sigma}}_k^{(0)}$ for CLEMM and GMM: (1) true population parameters; (2) the best K-means results (the lowest within-cluster variation) among 20 random starting values; (3) HC with Wald distance. We consider both the general CLEMM and the shared-CLEMM settings.

For each of the five simulation models, we generate 100 independent data sets with sample size $n = 1000$. Models (M1)–(M3) follow the general covariance structure in (2.2) and Models (S1) and (S2) follow the shared covariance structure in (4.1). The model parameters are set in the following way. The entries in $\boldsymbol{\Gamma} \in \mathbb{R}^{p\times u}$ are first generated randomly from Uniform$(0, 1)$; and then we orthogonalize $\boldsymbol{\Gamma}$ such that $(\boldsymbol{\Gamma}, \boldsymbol{\Gamma}_0)$ is an orthogonal matrix. The mean vectors $\boldsymbol{\mu}_k = \boldsymbol{\Gamma}\boldsymbol{\eta}_k$ with each $\boldsymbol{\eta}_k \in \mathbb{R}^{u\times 1}$ randomly filled with $N(0, 1)$ random numbers. The symmetric positive definite matrices $\boldsymbol{\Omega}_k \in \mathbb{R}^{u\times u}$ and $\boldsymbol{\Omega}_0 \in \mathbb{R}^{(p-u)\times(p-u)}$ are generated as $\mathbf{A}\mathbf{A}^T/\|\mathbf{A}\mathbf{A}^T\|_F$, where $\mathbf{A}$ is a square matrix with compatible dimensions and the entries in $\mathbf{A}$ are randomly generated from Uniform$(0, 1)$. Finally, we let $\boldsymbol{\Sigma}_k^* = \boldsymbol{\Gamma}\boldsymbol{\Omega}_k\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T$ and standardize it to $\boldsymbol{\Sigma}_k = \sigma^2 \cdot \boldsymbol{\Sigma}^*/\|\boldsymbol{\Sigma}^*\|_F$, where the scalar σ^2 is chosen such that Bayes' error is around 5–10%. Other model specific parameters are summarized in the following.

- (M1). $K = 3, \pi = (0.3, 0.2, 0.5), (p, u) = (15, 1), \sigma^2 = 1.25$.We encourage more heteroscedasticity by letting $\boldsymbol{\Sigma}_k^* = \exp(-k) \cdot \boldsymbol{\Gamma}\boldsymbol{\Omega}_k\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T$.
- (M2). $K = 4, \pi = (0.25, 0.25, 0.25, 0.25), (p, u) = (15, 2), \sigma^2 = 3.5$. We encourage more heteroscedasticity by letting $\boldsymbol{\Sigma}_k^* = \exp(-0.1\times k)\cdot\boldsymbol{\Gamma}\boldsymbol{\Omega}_k\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T$.
- (M3). $K = 4, \pi = (0.25, 0.25, 0.25, 0.25), (p, u) = (15, 3), \sigma^2 = 12$.
- (S1) $K = 3, \pi = (0.3, 0.2, 0.5), (p, u) = (15, 1), \sigma^2 = 0.25$.
- (S2) $K = 4, \pi = (0.25, 0.25, 0.25, 0.25), (p, u) = (50, 2), \sigma^2 = 2$.

Table 1 summarizes the mis-clustering error rates of each method. Clearly, CLEMM improves over GMM significantly regardless of the initial values. If we use the true parameter as the initial value for the EM and the envelope-EM algorithms, then CLEMM estimator almost achieves the Bayes' error rate while the GMM estimator may still have error rate more than twice of the Bayes' error rate. In four out of the five models (except Model (S1)), the K-means and HC algorithms actually fail to provide a good initial estimator. However, the CLEMM can drastically reduce the clustering error from those poor initial estimators. This demonstrates that CLEMM is robust to initialization. In all

these models, the CLEMM demonstrates huge advantages over K-means and HC because of the model-based natural and parsimonious modeling of means as well as covariance matrices. Moreover, CLEMM either has a comparable performance or even outperforms LDA and QDA in these models. This indicates that CLEMM can utilize the latent lower-dimensional structure of data for better clustering results.

Next, we compare the parameter estimation accuracy of GMM and CLEMM. The results are summarized in Tables 2, 3 and 4 for the mean estimation of $\boldsymbol{\mu}_k$, the cluster size estimation π_k, and the covariance estimation $\boldsymbol{\Sigma}_k$, respectively. Again, not surprisingly, CLEMM has much more accurate parameter estimation comparing to GMM, especially in the more general cases (M1)–(M3) where the reduction in the number of free parameters is large. Even in the relatively simpler cases of shared covariance Models (S1) and (S2), CLEMM still has significant improvements over GMM. Moreover, as we include in Appendix E.2, the computational time costs of the envelope-EM algorithm for CLEMM is actually less than that of the EM algorithm for GMM under these two models.

Finally, we demonstrate the envelope dimension selection results by our AWE criterion in Table 5, where we report the percentage of simulated data sets from which the true dimension is selected by AWE. The dimension selection procedure seems promising, and we will further illustrate the AWE selection criterion on the following real data sets.

TABLE 1

Clustering error rates (%). The reported numbers are averaged results and their standard errors (in the parenthesis) over 100 replications.

Cluster error rate	M1	M2	M3	S1	S2
CLEMM(True)	6.4 (0.1)	7.1 (0.1)	5.2 (0.1)	5.9 (0.1)	5.8 (0.1)
GMM(True)	16.1 (0.7)	14.3 (0.5)	6.6 (0.1)	6.4 (0.1)	8.0 (0.1)
MCFA(True)	11.5 (0.1)	41.5 (0.1)	43.1 (0.1)	8.7 (0.1)	21.3 (0.1)
CLEMM(k-means)	6.9 (0.2)	12.9 (1.2)	10.1 (1.0)	5.9 (0.1)	6.7 (0.4)
GMM(k-means)	23.3 (0.6)	26.9 (0.4)	29.3 (1.2)	6.4 (0.1)	10.6 (0.6)
MCFA(k-means)	11.8 (0.1)	41.4 (0.1)	47.2 (0.1)	7.7 (0.1)	23.9 (0.1)
CLEMM(HC)	6.8 (0.2)	17.4 (1.5)	10.0 (1.0)	6.1 (0.2)	9.3 (0.9)
GMM(HC)	24.8 (0.6)	27.1 (0.3)	25.9 (1.2)	6.6 (0.2)	13.9 (1.0)
MCFA(HC)	11.5 (0.1)	42.2 (0.1)	46.6 (0.1)	9.2 (0.1)	23.8 (0.1)
k-means	36.6 (0.1)	42.9 (0.3)	67.6 (0.3)	6.4 (0.1)	39.7 (0.2)
HC	32.3 (0.3)	45.0 (0.8)	62.4 (0.5)	8.9 (0.3)	39.5 (0.5)
Bayes error	5.9 (0.1)	6.5 (0.1)	5.0 (0.1)	5.8 (0.1)	5.6 (0.1)
LDA	8.6 (1.0)	25.8 (1.8)	24.1 (1.5)	6.2 (0.8)	6.3 (0.8)
QDA	6.7 (0.9)	8.0 (0.9)	6.1 (0.9)	7.2 (0.8)	13.1 (1.3)

TABLE 2
Estimation errors in cluster locations $\sum_{k=1}^{K} \|\widehat{\boldsymbol{\mu}}_k - \boldsymbol{\mu}_k\|_F$. The reported numbers are averaged results and their standard errors (in the parenthesis) over 100 replications.

$\sum_{k=1}^{K} \|\widehat{\boldsymbol{\mu}}_k - \boldsymbol{\mu}_k\|_F$	M1	M2	M3	M4	M5
CLEMM(True)	0.21 (0.01)	0.49 (0.01)	0.84 (0.02)	0.10 (0.01)	0.39 (0.01)
GMM(True)	0.51 (0.02)	1.03 (0.04)	1.17 (0.02)	0.13 (0.01)	0.51 (0.01)
MCFA(True)	0.74 (0.08)	4.08 (0.11)	6.75 (0.22)	0.39 (0.08)	2.06 (0.11)
CLEMM(k-means)	0.24 (0.01)	1.08 (0.13)	1.73 (0.19)	0.10 (0.01)	0.50 (0.05)
GMM(k-means)	0.79 (0.02)	2.20 (0.08)	4.94 (0.22)	0.13 (0.01)	0.71 (0.06)
MCFA(k-means)	0.72 (0.07)	4.41 (0.11)	7.82 (0.22)	0.26 (0.04)	2.48 (0.10)
CLEMM(HC)	0.23 (0.01)	1.72 (0.19)	1.73 (0.20)	0.14 (0.03)	0.82 (0.10)
GMM(HC)	0.89 (0.04)	2.71 (0.11)	4.51 (0.23)	0.19 (0.03)	1.02 (0.09)
MCFA(HC)	0.68 (0.06)	4.78 (0.12)	7.59 (0.24)	0.47 (0.10)	2.47 (0.12)

TABLE 3
Estimation errors in cluster sizes $\sum_{k=1}^{K} \|\widehat{\pi}_k - \pi_k\|_F$. The reported numbers are averaged results and their standard errors (in the parenthesis) over 100 replications.

$\sum_{k=1}^{K} \|\widehat{\pi}_k - \pi_k\|_F$	M1	M2	M3	S1	S2
CLEMM(True)	0.08 (0.01)	0.07 (0.01)	0.05 (0.01)	0.04 (0.01)	0.06 (0.01)
GMM(True)	0.23 (0.02)	0.18 (0.01)	0.06 (0.01)	0.04 (0.01)	0.07 (0.01)
MCFA(True)	0.17 (0.02)	0.41 (0.02)	0.49 (0.03)	0.12 (0.10)	0.37 (0.02)
CLEMM(k-means)	0.10 (0.01)	0.12 (0.01)	0.12 (0.02)	0.04 (0.01)	0.08 (0.01)
GMM(k-means)	0.32 (0.02)	0.35 (0.01)	0.36 (0.02)	0.04 (0.01)	0.10 (0.01)
MCFA(k-means)	0.18 (0.02)	0.43 (0.01)	0.56 (0.02)	0.11 (0.01)	0.45 (0.01)
CLEMM(HC)	0.10 (0.01)	0.23 (0.02)	0.11 (0.01)	0.04 (0.01)	0.13 (0.02)
GMM(HC)	0.35 (0.02)	0.39 (0.01)	0.34 (0.02)	0.05 (0.01)	0.17 (0.02)
MCFA(HC)	0.18 (0.02)	0.48 (0.01)	0.56 (0.03)	0.14 (0.01)	0.45 (0.01)

TABLE 4
Estimation errors in cluster shapes and scales $\sum_{k=1}^{K} \|\widehat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k\|_F$. The reported numbers are averaged results and their standard errors (in the parenthesis) over 100 replications.

$\sum_{k=1}^{K} \|\widehat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k\|_F$	M1	M2	M3	S1	S2
CLEMM(True)	0.23 (0.01)	1.04 (0.03)	3.43 (0.09)	0.01 (0.01)	0.10 (0.01)
GMM (True)	0.47 (0.01)	2.24 (0.07)	5.96 (0.09)	0.02 (0.01)	0.12 (0.01)
MCFA (True)	2.85 (0.02)	7.31 (0.06)	19.52 (0.48)	0.47 (0.01)	4.35 (0.08)
CLEMM(k-means)	0.60 (0.02)	4.35 (0.17)	16.33 (0.53)	0.01 (0.01)	0.10 (0.01)
GMM(k-means)	0.79 (0.02)	5.40 (0.15)	21.47 (0.45)	0.02 (0.01)	0.13 (0.01)
MCFA(k-means)	2.85 (0.02)	7.38 (0.06)	20.50 (0.34)	0.46 (0.01)	4.38 (0.04)
CLEMM(HC)	0.60 (0.02)	5.05 (0.17)	15.94 (0.50)	0.01 (0.01)	0.12 (0.02)
GMM(HC)	0.79 (0.02)	5.98 (0.13)	20.80 (0.49)	0.02 (0.01)	0.14 (0.01)
MCFA(HC)	2.81 (0.01)	7.36 (0.07)	20.7 (0.43)	0.48 (0.01)	4.48 (0.07)

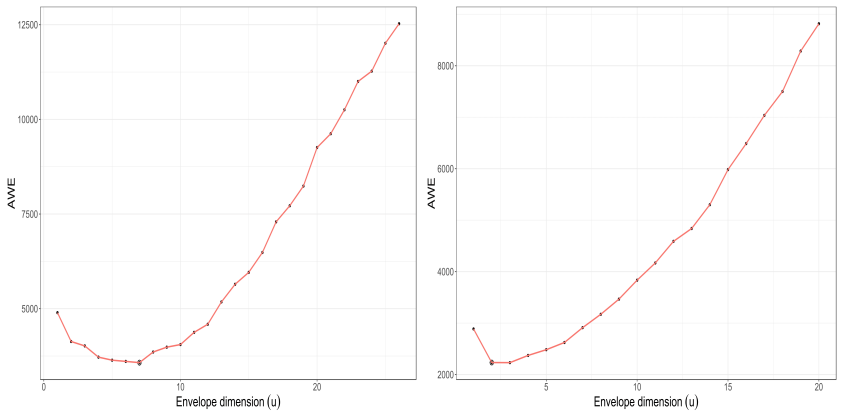
TABLE 5
Correct envelope dimension selection (%) out of 100 replications. We use the K-means estimator as the initial values in the envelope-EM algorithm for CLEMM.

Model	M1	M2	M3	S1	S2
Correct selection (%)	100	95	100	89	84

6.2. Real Data Analysis

We choose two benchmark classification data examples for comparing the relative performances of CLEMM and GMM, as well as the results from K-means, MCFA and HC algorithms. For the EM and envelope-EM algorithms, we always use the results from K-means as initial values for fair comparison.

The first example is the Forest Type data which contains four different forest types (see https://archive.ics.uci.edu/ml/datasets/Forest+type+mapping). We combine the original training and testing samples so that there are 523 observations and 27 variables. The second example is the Waveform data set (see Hastie and Tibshirani [23] for more background) which is a simulated three-class classification data set with 21 predictors and 800 observations. The three classes of waveforms are random convex combinations of two equal-lateral right triangle function plus independent Gaussian noise. It is commonly used as a benchmark data set in machine learning study to demonstrate the robustness of methods, because the distribution is actually not a mixture of Gaussian distributions.



(a) Forest Type data (b) Waveform data

Fig 2: AWE scores for dimension selection in CLEMM.

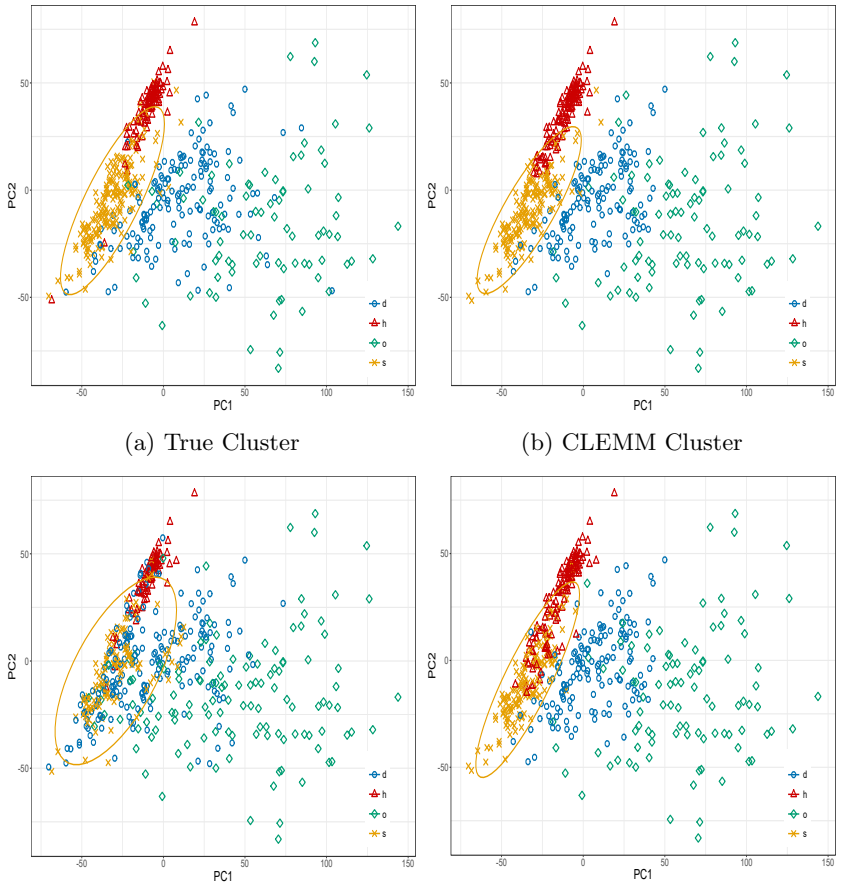
First of all, we determine the envelope dimension u using the AWE criterion proposed in Section 5. Figure 2 visualizes the AWE scores for all candidate envelope dimensions on the three data sets. It is clearly suggested from the

plots that we use $u = 7$ for the Forest Type data and $u = 2$ for the Waveform data.

In Table 6, we report the clustering error rates of each methods. In addition to the clustering methods, we also include the discriminant analysis results from LDA and QDA by using the true class labels. For LDA and QDA error rates, we use the R built-in functions `LDA` and `QDA` to obtain the prediction errors of class membership from leave-one-out cross-validation. In both data sets, we see substantial improvements by CLEMM over other clustering methods. It is very encouraging to see that CLEMM (without knowing the class label) has comparable results as the discriminant analysis methods (which makes use of the class label in estimation) in the Sound and Forest Type data. Moreover, for the Waveform data, where $\mathbf{X} \mid Y$ is no longer Gaussian, the clustering accuracy of CLEMM is even better than the classification accuracy of LDA and QDA. This indicates that our CLEMM method is more robust to non-Gaussian distributions than other clustering and discriminant analysis methods. Overall, comparing with other clustering methods, CLEMM is better at capturing the information from real data.

TABLE 6
Clustering and classification error rates (%) on the two data sets.

Cluster error	CLEMM	GMM	MCFA	K-means	HC	LDA	QDA
Forest Type	13.0	38.0	20.8	22.2	24.5	11.1	17.2
Waveform	14.8	22.6	15.8	48.3	45.6	17.8	18.3

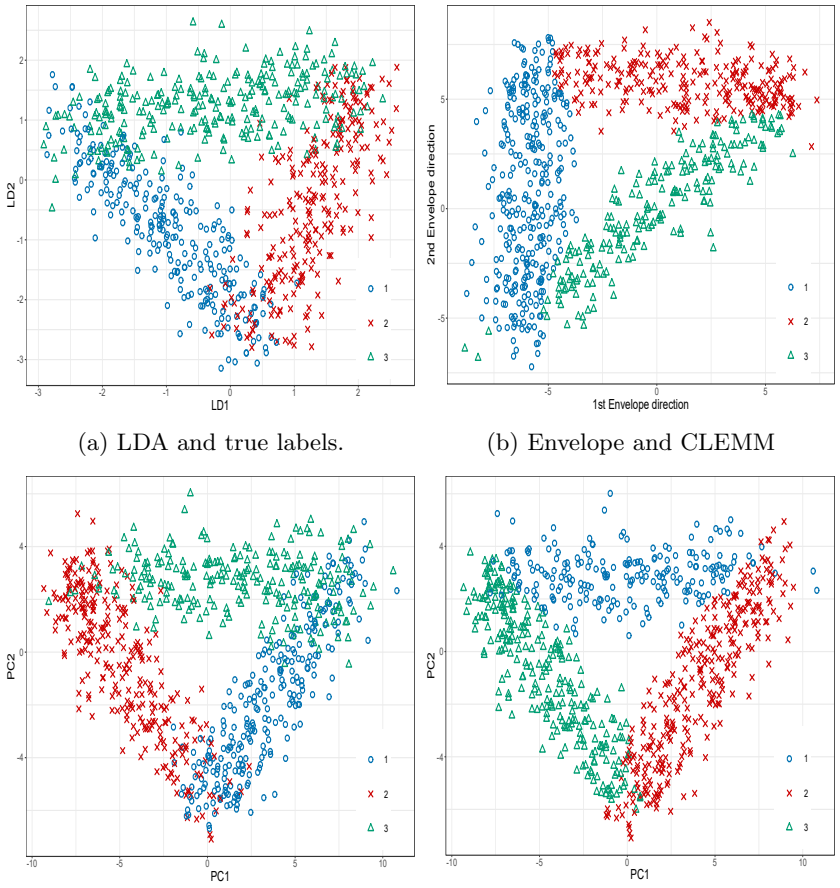


(a) True Cluster (b) CLEMM Cluster

(c) GMM Cluster (d) MCFA foresttype Cluster

Fig 3: Forest type data. Cluster Visualization, where an observation with very large PC1 score (x-axis) is removed for better visualization.

In Figure 3, we visualize the true classes and the estimated clusters by CLEMM, GMM and MCFA. We visualize the data on the first two principal components of the data. We see clearly that CLEMM can better capture the variability of clusters, especially for the cluster of "Sugi" Forest Type. For the Waveform data, since we have the envelope dimension selected as $u = 2$, we next investigate the visualization of different subspaces: envelope, LDA subspace, principal component subspace. In Figure 4, we visualize the clusters on the envelope subspace estimated by CLEMM. This produces much better separated clusters than the visualization of GMM estimated clusters on the principal component subspace, and of true classes on the LDA subspace. The improvement on visualization by CLEMM over the true classes on LDA subspace is consistent with our earlier findings on the error rates (Table 6): the CLEMM



(a) LDA and true labels.

(b) Envelope and CLEMM

(c) PCA and GMM

(d) PCA and MCFA

Fig 4: Waveform data. Cluster Visualization by different subspaces: LDA visualization (with true class labels); envelope visualization (with CLEMM estimated clusters); principal components visualization (with GMM estimated clusters).

(without knowing Y) is even more accurate than LDA classification (knowing Y). We can see that the first envelope direction can correctly characterize the cluster location and material variation for one of the cluster (the blue dots). The results in Figure 4 suggest that CLEMM can assist in data visualization when the estimated envelope dimension is small.

7. Discussion

In this paper, we extend the envelope methodology to unsupervised learning problems by considering model-based clustering. The proposed parsimonious

model, CLEMM, can simultaneously achieve dimension reduction, visualization, and improved parameter estimation and clustering. Compared to the standard GMM, CLEMM is much more effective in capturing the cluster location changes and the heterogeneous variation across different clusters. The envelope-EM algorithm developed in this paper can also be modified to handle missing data in general. As future research directions, the proposed clustering framework can be extended to mixture discriminant analysis [23, 24], to incorporate regularization techniques such as the ones from Friedman [21] and Cai et al. [6], and to various unsupervised and semi-supervised problems.

Appendix A: Equivalence between (2.2) and (2.3)

Proof. We first show that (2.2) $\Rightarrow$ (2.3). Because $(\boldsymbol{\Gamma}, \boldsymbol{\Gamma}_0) \in \mathbb{R}^{p\times p}$ is orthogonal matrix, we can write $\mathbf{X} = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^T\mathbf{X} + \boldsymbol{\Gamma}_0\boldsymbol{\Gamma}_0^T\mathbf{X} = \boldsymbol{\Gamma}\mathbf{X}_{\mathrm{M}} + \boldsymbol{\Gamma}_0\mathbf{X}_{\mathrm{IM}}$. By (2.2), we have $\mathbf{X} \sim \sum_{k=1}^K \pi_k N(\boldsymbol{\Gamma}\boldsymbol{\alpha}_k, \boldsymbol{\Gamma}\boldsymbol{\Omega}_k\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T)$, which implies (2.3).

We now show that (2.2) $\Leftarrow$ (2.3). By (2.3), we have $\mathbf{X} \sim \sum_{k=1}^K \pi_k N(\boldsymbol{\Gamma}\boldsymbol{\alpha}_k, \boldsymbol{\Gamma}\boldsymbol{\Omega}_k\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^T)$ which directly implies (2.2). □

Appendix B: Proofs for Proposition 1

For a more general case, we do not the center the data first but assume $\boldsymbol{\mu}_k - \overline{\boldsymbol{\mu}} = \boldsymbol{\Gamma}\boldsymbol{\alpha}_k$. Given an envelope subspace basis $\boldsymbol{\Gamma}$, we maximize $Q(\boldsymbol{\phi} \mid \widehat{\boldsymbol{\theta}}^{(m)})$ to update $\boldsymbol{\theta}$.

$$\begin{aligned} Q(\boldsymbol{\phi} \mid \widehat{\boldsymbol{\theta}}^{(m)}) &\simeq \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \{\log \pi_k - \frac{1}{2}\log|\boldsymbol{\Sigma}_k| - \frac{1}{2}(\mathbf{X}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\mathbf{X}_i - \boldsymbol{\mu}_k)\} \\ &\simeq \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \{\log\pi_k - \frac{1}{2}(\log|\boldsymbol{\Omega}_k| + \log|\boldsymbol{\Omega}_0|) \\ &- \frac{1}{2}(\mathbf{X}_i - \overline{\boldsymbol{\mu}} - \boldsymbol{\Gamma}\boldsymbol{\alpha}_k)^T(\boldsymbol{\Gamma}\boldsymbol{\Omega}_k^{-1}\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0^{-1}\boldsymbol{\Gamma}_0^T)(\mathbf{X}_i - \overline{\boldsymbol{\mu}} - \boldsymbol{\Gamma}\boldsymbol{\alpha}_k)\} \end{aligned}$$

We can calculate the maximizing values of all the parameters of interest. Since $\widehat{\pi}_k = \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}{n}$, we have constraints that $\sum_{k=1}^K \widehat{\pi}_k\boldsymbol{\alpha}_k = 0$, i.e. $\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)}\boldsymbol{\alpha}_k = 0$.

Maximizing value for $\overline{\boldsymbol{\mu}}$. We have $\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} [\overline{\boldsymbol{\mu}} - (\mathbf{X}_i - \boldsymbol{\Gamma}\boldsymbol{\alpha}_j)] = 0$, therefor the maximizing value for $\overline{\boldsymbol{\mu}}$ is,

$$\widehat{\overline{\boldsymbol{\mu}}} = \overline{\mathbf{X}} - \boldsymbol{\Gamma}\frac{\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)}\boldsymbol{\alpha}_k}{n} = \overline{\mathbf{X}},$$

where the last equation holds because of constraints.

Maximizing value for $\boldsymbol{\alpha}_k$. Replace $\overline{\boldsymbol{\mu}}$ by $\overline{\mathbf{X}}$, then we have $\widehat{\boldsymbol{\alpha}}_k$ is the minimizer of the function

$$\frac{1}{2}\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \{\left(\boldsymbol{\Gamma}^T(\mathbf{X}_i - \overline{\mathbf{X}}) - \boldsymbol{\alpha}_k\right)^T \boldsymbol{\Omega}_k^{-1} \left(\boldsymbol{\Gamma}^T(\mathbf{X}_i - \overline{\mathbf{X}}) - \boldsymbol{\alpha}_k\right)\}$$

under the constraint that $\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \boldsymbol{\alpha}_k = 0$. We have $\widehat{\boldsymbol{\alpha}}_k = \boldsymbol{\Gamma}^T(\frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)})\mathbf{X}_i}{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}} - \overline{\mathbf{X}}) = \boldsymbol{\Gamma}^T(\widetilde{\boldsymbol{\mu}}_k^{(m)} - \overline{\mathbf{X}})$ for $k = 1, 2, ..., K$.

Maximizing value for $\boldsymbol{\Omega}_k$, $\boldsymbol{\Omega}_0$. Replace the maximizing values for $\overline{\boldsymbol{\mu}}$ and $\boldsymbol{\alpha}_k$, we have

$$\begin{aligned} Q(\boldsymbol{\phi} \mid \widehat{\boldsymbol{\theta}}^{(m)}) \quad \simeq \quad & -\frac{1}{2}\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \log|\boldsymbol{\Omega}_j| - \frac{1}{2}\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} [\{\boldsymbol{\Gamma}^T(\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})\}^T \boldsymbol{\Omega}_k^{-1} \{\boldsymbol{\Gamma}^T(\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})\}] \\ & -\frac{n}{2}\log|\boldsymbol{\Omega}_0| - \frac{1}{2}\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} [(\mathbf{X}_i - \overline{\mathbf{X}})^T \boldsymbol{\Gamma}_0 \boldsymbol{\Omega}_0^{-1} \boldsymbol{\Gamma}_0^T (\mathbf{X}_i - \overline{\mathbf{X}})] \end{aligned}$$

Denote

$$\begin{aligned} \mathbf{S}_{\mathrm{x}} \quad &= \quad \frac{\sum_{i=1}^n (\mathbf{X}_i - \overline{\mathbf{X}})(\mathbf{X}_i - \overline{\mathbf{X}})^T}{n}, \\ \mathbf{S}_k^{(m)} \quad &= \quad \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)} (\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})(\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})}{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}, \end{aligned}$$

we have $\widehat{\boldsymbol{\Omega}}_k = \boldsymbol{\Gamma}^T \mathbf{S}_k^{(m)} \boldsymbol{\Gamma}$, $\widehat{\boldsymbol{\Omega}}_0 = \boldsymbol{\Gamma}_0^T \mathbf{S}_{\mathrm{x}} \boldsymbol{\Gamma}_0$.

Maximizing value for $\boldsymbol{\Gamma}$. We have

$$\begin{aligned} -2 \times Q(\boldsymbol{\phi} \mid \widehat{\boldsymbol{\theta}}^{(m)}) \quad \simeq \quad & \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \log|\boldsymbol{\Gamma}^T \mathbf{S}_k^{(m)} \boldsymbol{\Gamma}| + n \log|\boldsymbol{\Gamma}_0^T \mathbf{S}_{\mathrm{x}} \boldsymbol{\Gamma}_0| \\ = \quad & \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \log|\boldsymbol{\Gamma}^T \mathbf{S}_k^{(m)} \boldsymbol{\Gamma}| + n \log|\boldsymbol{\Gamma}^T \mathbf{S}_{\mathrm{x}}^{-1} \boldsymbol{\Gamma}| \end{aligned}$$

Therefore, we obtain $\widehat{\boldsymbol{\Gamma}}$ by minimizing $\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \log|\boldsymbol{\Gamma}^T \mathbf{S}_k^{(m)} \boldsymbol{\Gamma}| + n \log|\boldsymbol{\Gamma}^T \mathbf{S}_{\mathrm{x}}^{-1} \boldsymbol{\Gamma}|$ over the semi-orthogonal constrain that $\boldsymbol{\Gamma}^T \boldsymbol{\Gamma} = \mathbf{I}_u$.

Appendix C: Proofs for Proposition 2

Similar as the previous proof, we do not the center the data first but assume $\boldsymbol{\mu}_k - \overline{\boldsymbol{\mu}} = \boldsymbol{\Gamma}\boldsymbol{\alpha}_k$. Given an envelope basis $\boldsymbol{\Gamma}$, we maximize $Q(\boldsymbol{\phi} \mid \widehat{\boldsymbol{\theta}}^{(m)})$ to update $\boldsymbol{\theta}$.

$$\begin{aligned} Q(\boldsymbol{\phi} \mid \widehat{\boldsymbol{\theta}}^{(m)}) \quad \simeq \quad & \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \{\log \pi_k - \frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}(\mathbf{X}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1} (\mathbf{X}_i - \boldsymbol{\mu}_k)\} \\ \simeq \quad & \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \{\log \pi_k - \frac{1}{2}(\log|\boldsymbol{\Omega}| + \log|\boldsymbol{\Omega}_0|) \\ - \quad & \frac{1}{2}(\mathbf{X}_i - \overline{\boldsymbol{\mu}} - \boldsymbol{\Gamma}\boldsymbol{\alpha}_k)^T (\boldsymbol{\Gamma}\boldsymbol{\Omega}^{-1}\boldsymbol{\Gamma}^T + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0^{-1}\boldsymbol{\Gamma}_0^T)(\mathbf{X}_i - \overline{\boldsymbol{\mu}} - \boldsymbol{\Gamma}\boldsymbol{\alpha}_k)\} \end{aligned}$$

We can calculate the maximizing values of all the parameters of interest. Since $\widehat{\pi}_k = \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}{n}$, we have constraints that $\sum_{k=1}^K \widehat{\pi}_k \boldsymbol{\alpha}_k = 0$, i.e. $\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \boldsymbol{\alpha}_k = 0$.

Maximizing value for $\overline{\boldsymbol{\mu}}$. We have $\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \left[\overline{\boldsymbol{\mu}} - (\mathbf{X}_i - \boldsymbol{\Gamma}\boldsymbol{\alpha}_k)\right] = 0$, therefor the maximizing value for $\overline{\boldsymbol{\mu}}$ is,

$$\widehat{\overline{\boldsymbol{\mu}}} = \overline{\mathbf{X}} - \boldsymbol{\Gamma} \frac{\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \boldsymbol{\alpha}_k}{n} = \overline{\mathbf{X}},$$

where the last equation holds because of constraints.

Maximizing value for $\boldsymbol{\alpha}_k$. Replace $\bar{\boldsymbol{\mu}}$ by $\overline{\mathbf{X}}$, then we have $\widehat{\boldsymbol{\alpha}}_k$ is the minimizer of the function

$$\frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \left\{ \left(\boldsymbol{\Gamma}^T (\mathbf{X}_i - \overline{\mathbf{X}}) - \boldsymbol{\alpha}_k\right)^T \boldsymbol{\Omega}^{-1} \left(\boldsymbol{\Gamma}^T (\mathbf{X}_i - \overline{\mathbf{X}}) - \boldsymbol{\alpha}_k\right) \right\}$$

under the constraint that $\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \boldsymbol{\alpha}_k = 0$. We have $\widehat{\boldsymbol{\alpha}}_k = \boldsymbol{\Gamma}^T (\frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}) \mathbf{X}_i}{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}} - \overline{\mathbf{X}}) = \boldsymbol{\Gamma}^T (\widetilde{\boldsymbol{\mu}}_k^{(m)} - \overline{\mathbf{X}})$ for $k = 1, 2, ..., K$.

Maximizing value for $\boldsymbol{\Omega}$ and $\boldsymbol{\Omega}_0$. Replace the maximizing values for $\overline{\boldsymbol{\mu}}$ and $\boldsymbol{\alpha}_k$, we have

$$\begin{aligned} Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}) \quad \simeq \quad & -\frac{n}{2} \log |\boldsymbol{\Omega}| - \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} [\{\boldsymbol{\Gamma}^T (\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})\}^T \boldsymbol{\Omega}^{-1} \\ & \{\boldsymbol{\Gamma}^T (\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})\}] - \frac{n}{2} \log |\boldsymbol{\Omega}_0| - \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} \\ & [(\mathbf{X}_i - \overline{\mathbf{X}})^T \boldsymbol{\Gamma}_0 \boldsymbol{\Omega}_0^{-1} \boldsymbol{\Gamma}_0^T (\mathbf{X}_i - \overline{\mathbf{X}})] \end{aligned}$$

Denote

$$\begin{aligned} \mathbf{S}_{\mathrm{x}} \quad &= \quad \frac{\sum_{i=1}^n (\mathbf{X}_i - \overline{\mathbf{X}})(\mathbf{X}_i - \overline{\mathbf{X}})^T}{n}, \\ \mathbf{S}^{(m)} \quad &= \quad \frac{\sum_{i=1}^n \sum_{k=1}^K \widehat{\eta}_{ik}^{(m)} (\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})(\mathbf{X}_i - \widetilde{\boldsymbol{\mu}}_k^{(m)})^T}{n}, \end{aligned}$$

we have $\widehat{\boldsymbol{\Omega}} = \boldsymbol{\Gamma}^T \mathbf{S}^{(m)} \boldsymbol{\Gamma}$, $\widehat{\boldsymbol{\Omega}}_0 = \boldsymbol{\Gamma}_0^T \mathbf{S}_{\mathrm{x}} \boldsymbol{\Gamma}_0$.

Maximizing value for $\boldsymbol{\beta}_k$. We have $\widehat{\boldsymbol{\beta}}_k = {\boldsymbol{\Sigma}^{(m)}}^{-1} (\boldsymbol{\mu}_k^{(m)} - \boldsymbol{\mu}_1^{(m)}) = \mathbf{P}_{\widehat{\boldsymbol{\Gamma}}(\mathbf{S}^{(m)})} {\mathbf{S}^{(m)}}^{-1} \left(\widetilde{\boldsymbol{\mu}}_k^{(m)} - \widetilde{\boldsymbol{\mu}}_1^{(m)}\right)$, $k = 2, ..., K$.

Maximizing value for $\boldsymbol{\Gamma}$. We have

$$\begin{aligned} -\frac{2}{n} Q(\boldsymbol{\phi} \mid \widehat{\boldsymbol{\theta}}^{(m)}) \quad &\simeq \quad \log |\boldsymbol{\Gamma}^T \mathbf{S}^{(m)} \boldsymbol{\Gamma}| + \log |\boldsymbol{\Gamma}_0^T \mathbf{S}_{\mathrm{x}} \boldsymbol{\Gamma}_0| \\ &= \quad \log |\boldsymbol{\Gamma}^T \mathbf{S}^{(m)} \boldsymbol{\Gamma}| + \log |\boldsymbol{\Gamma}^T \mathbf{S}_{\mathrm{x}}^{-1} \boldsymbol{\Gamma}| \end{aligned}$$

Therefore, we obtain $\widehat{\boldsymbol{\Gamma}}$ by minimizing $\log |\boldsymbol{\Gamma}^T \mathbf{S}^{(m)} \boldsymbol{\Gamma}| + \log |\boldsymbol{\Gamma}^T \mathbf{S}_{\mathrm{x}}^{-1} \boldsymbol{\Gamma}|$ over the semi-orthogonal constrain that $\boldsymbol{\Gamma}^T \boldsymbol{\Gamma} = \mathbf{I}_u$

Appendix D: EM algorithm for CLEMM-Shared

Algorithm 2 EM algorithm for CLEMM-Shared

1: Data $\{\mathbf{X}_1, ..., \mathbf{X}_n\} \subset \mathbb{R}^p$ and parameters $\boldsymbol{\theta} = (\pi_1, ..., \pi_K, \boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_K, \boldsymbol{\Sigma})$
2: Initialize $\widehat{\pi}_k^{(0)}, \widehat{\boldsymbol{\mu}}_k^{(0)}, \widehat{\boldsymbol{\Sigma}}^{(0)}$ for $k = 1, 2, ...K$.
3: **E**-step: For $k = 1, ..., K$, calculate $\widehat{\eta}_{ik}^{(m)}$.
4: **M**-step:

1. Calculate $\widetilde{\boldsymbol{\mu}}_k^{(m)} = \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)} \mathbf{X}_i}{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}$ and $\hat{\pi}_k^{(m+1)} = \frac{\sum_{i=1}^n \widehat{\eta}_{ik}^{(m)}}{n}$ for $k = 1, 2, ..., K$.
2. Calculate $\widehat{\boldsymbol{\Gamma}} = \arg\min_{\boldsymbol{\Gamma} \in \mathcal{G}(p,u)} \left\{ \log |\boldsymbol{\Gamma}^T \mathbf{S}^{(m)} \boldsymbol{\Gamma}| + \log |\boldsymbol{\Gamma}^T \mathbf{S}_{\mathrm{x}}^{-1} \boldsymbol{\Gamma}| \right\}$.
3. For $k = 1, ..., K$

$$\begin{aligned} \widehat{\boldsymbol{\mu}}_k^{(m+1)} &= \overline{\mathbf{X}} + \widehat{\boldsymbol{\Gamma}}\widehat{\boldsymbol{\Gamma}}^T \left[\widetilde{\boldsymbol{\mu}}_k^{(m)} - \overline{\mathbf{X}} \right] \\ \widehat{\boldsymbol{\Sigma}}^{(m+1)} &= \widehat{\boldsymbol{\Gamma}}(\widehat{\boldsymbol{\Gamma}}^T \mathbf{S}^{(m)} \widehat{\boldsymbol{\Gamma}})\widehat{\boldsymbol{\Gamma}}^T + \widehat{\boldsymbol{\Gamma}}_0(\widehat{\boldsymbol{\Gamma}}_0^T \mathbf{S}_{\mathrm{x}} \widehat{\boldsymbol{\Gamma}}_0)\widehat{\boldsymbol{\Gamma}}_0^T \end{aligned}$$

5: Check convergence. If not converged, set $m = m + 1$.

Appendix E: Additional numerical results and implementation details

E.1. EM implementation details

For the convergence criterion of the EM algorithm, we check to see if $\ell_o(\boldsymbol{\theta}^{(m+1)}) - \ell_o(\boldsymbol{\theta}^{(m)}) < 1e-7$. We also stop running the algorithm if it reaches the maximum iteration times 800. It is worth mentioning that due to the non-convex estimation of $F^{(m)}(\boldsymbol{\Gamma})$ and $G^{(m)}(\boldsymbol{\Gamma})$, the log-likelihood sequence $\ell_o(\boldsymbol{\theta}^{(m)})$ might not be non-decreasing all the time. We might encounter the situation when the log-likelihood slightly drops, in this case, we will stop at the current iteration and use the estimation from previous step.

E.2. Computation time comparison

In CLEMM, we use 1D algorithm to find initial value for $\boldsymbol{\Gamma}$ and then do full manifold optimization to get the minimizer for $G^{(m)}(\boldsymbol{\Gamma})$. In CLEMM-Share, we use ECD alone for optimization for $F^{(m)}(\boldsymbol{\Gamma})$. From Table 7, we see that due to the estimation of the envelope subspace, CLEMM is slower than GMM. However, in the special case of CLEMM-Shared, it is significantly faster than GMM-Shared. The improvement comes from the ECD algorithm in solving $\boldsymbol{\Gamma}$ and the fast convergence of EM algorithm due to the estimation of $\boldsymbol{\Gamma}$.

TABLE 7
Computing time in seconds of (M1)-(M5). The reported numbers are averaged results and their standard errors (in the parenthesis) over 100 replications.

Computing time	M1	M2	M3	M4	M5
CLEMM(k-means)	102(4.2)	275(22)	148(5.0)	2.7(0.1)	17.1(0.5)
GMM(k-means)	23.4(1.2)	26.6(1.6)	24.1(1.2)	3.0(0.1)	28.3(1.3)

Acknowledgments

The authors would like to thank the Editor, the Associate Editor and the Referees for their insightful and constructive comments. Research for this paper was partly supported by grants CCF-1617691, CCF-1908969 and DMS-1613154 from the U.S. National Science Foundation.

References

[1] Absil, P.-A., Mahony, R., and Sepulchre, R. (2009). *Optimization algorithms on matrix manifolds*. Princeton University Press.

[2] Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike*, pages 199–213. Springer.

[3] Baek, J., McLachlan, G. J., and Flack, L. K. (2010). Mixtures of factor analyzers with common factor loadings: Applications to the clustering and visualization of high-dimensional data. *IEEE transactions on pattern analysis and machine intelligence*, 32(7):1298–1309.

[4] Banfield, J. D. and Raftery, A. E. (1993). Model-based gaussian and non-gaussian clustering. *Biometrics*, pages 803–821.

[5] Boyles, R. A. (1983). On the convergence of the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 47–50.

[6] Cai, T. T., Ma, J., and Zhang, L. (2019). Chime: Clustering of high-dimensional gaussian mixtures with em algorithm and its optimality. *The Annals of Statistics*, 47(3):1234–1267.

[7] Carreira-Perpinan, M. A. (2000). Mode-finding for mixtures of gaussian distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1318–1323.

[8] Chi, E. C. and Lange, K. (2015). Splitting methods for convex clustering. *Journal of Computational and Graphical Statistics*, 24(4):994–1013.

[9] Cook, R., Helland, I., and Su, Z. (2013). Envelopes and partial least squares regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75(5):851–877.

[10] Cook, R. D. (2018a). *An Introduction to Envelopes: Dimension Reduction for Efficient Estimation in Multivariate Statistics*. John Wiley & Sons.

[11] Cook, R. D. (2018b). Principal components, sufficient dimension reduction, and envelopes. *Annual Review of Statistics and Its Application*, 5:533–559.

[12] Cook, R. D., Li, B., and Chiaromonte, F. (2010). Envelope models for parsimonious and efficient multivariate linear regression. *Statistica Sinica*, pages 927–960.

[13] Cook, R. D. and Yin, X. (2001). Dimension reduction and visualization in discriminant analysis (with discussion). *Australian & New Zealand Journal of Statistics*, 43(2):147–199.

[14] Cook, R. D. and Zhang, X. (2015). Foundations for envelope models and methods. *Journal of the American Statistical Association*, 110(510):599–611.

[15] Cook, R. D. and Zhang, X. (2016). Algorithms for envelope estimation. *Journal of Computational and Graphical Statistics*, 25(1):284–300.

[16] Cook, R. D. and Zhang, X. (2018). Fast envelope algorithms. *Statistica Sinica*, 28(3):1179–1197.

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

[18] Ding, C. and He, X. (2004). K-means clustering via principal component analysis. In *Proceedings of the twenty-first international conference on Machine learning*, page 29. ACM.

[19] Eck, D. J., Geyer, C. J., and Cook, R. D. (2020). Combining envelope methodology and aster models for variance reduction in life history analyses. *Journal of Statistical Planning and Inference*, 205:283–292.

[20] Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, 97(458):611–631.

[21] Friedman, J. H. (1989). Regularized discriminant analysis. *Journal of the American statistical association*, 84(405):165–175.

[22] Hartigan, J. (1975). *Clustering Algorithms*. John Wiley & Sons Inc., New York.

[23] Hastie, T. and Tibshirani, R. (1996). Discriminant analysis by gaussian mixtures. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 155–176.

[24] Huang, M., Li, R., and Wang, S. (2013). Nonparametric mixture of regression models. *Journal of the American Statistical Association*, 108(503):929–941.

[25] Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323.

[26] Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254.

[27] Karlis, D. and Xekalaki, E. (2003). Choosing initial values for the em algorithm for finite mixtures. *Computational Statistics & Data Analysis*, 41(3-4):577–590.

[28] Kaufman, L. and Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons.

[29] Khare, K., Pal, S., and Su, Z. (2017). A bayesian approach for envelope models. *The Annals of Statistics*, 45(1):196–222.

[30] Li, B. (2018). *Sufficient Dimension Reduction: Methods and Applications*

with R. CRC Press.

[31] Li, L. and Zhang, X. (2017). Parsimonious tensor response regression. *Journal of the American Statistical Association*, 112(519):1131–1146.

[32] Lindsay, B. G. (1995). Mixture models: theory, geometry and applications. Institute of Mathematical Statistics.

[33] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

[34] McLachlan, G. and Peel, D. (2004). *Finite mixture models*. John Wiley & Sons.

[35] Rubin, D. B. and Thayer, D. T. (1982). Em algorithms for ml factor analysis. *Psychometrika*, 47(1):69–76.

[36] Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.

[37] Shin, S. J., Wu, Y., Zhang, H. H., and Liu, Y. (2014). Probability-enhanced sufficient dimension reduction for binary classification. *Biometrics*, 70(3):546–555.

[38] Steinbach, M., Ertöz, L., and Kumar, V. (2004). The challenges of clustering high dimensional data. In *New directions in statistical physics*, pages 273–309. Springer.

[39] Su, Z. and Cook, R. D. (2011). Partial envelopes for efficient estimation in multivariate linear regression. *Biometrika*, 98(1):133–146.

[40] Wang, J. and Wang, L. (2010). Sparse supervised dimension reduction in high dimensional classification. *Electronic Journal of Statistics*, 4:914–931.

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

[42] Witten, D. M. and Tibshirani, R. (2010). A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490):713–726.

[43] Wu, C. J. (1983). On the convergence properties of the em algorithm. *The Annals of statistics*, pages 95–103.

[44] Yao, W. and Lindsay, B. G. (2009). Bayesian mixture labeling by highest posterior density. *Journal of the American Statistical Association*, 104(486):758–767.

[45] Zhang, X. and Li, L. (2017). Tensor envelope partial least-squares regression. *Technometrics*, pages 1–11.

[46] Zhang, X. and Mai, Q. (2018). Model-free envelope dimension selection. *Electronic Journal of Statistics*, 12(2):2193–2216.

[47] Zhang, X. and Mai, Q. (2019). Efficient integration of sufficient dimension reduction and prediction in discriminant analysis. *Technometrics*, 61:259–272.

[48] Zhuang, X., Huang, Y., Palaniappan, K., and Zhao, Y. (1996). Gaussian mixture density modeling, decomposition, and applications. *IEEE Transactions on Image Processing*, 5(9):1293–1302.