

Diagonally-Weighted Generalized Method of Moments Estimation for Gaussian Mixture Modeling

Liu Zhang^{*} Oscar Mickelin[†] Sheng Xu[‡] Amit Singer[§]

Abstract

With direct computation of moments through order L in d dimensions, standard method of moments (MM) incurs $\Omega(d^L)$ computation and storage costs, and standard generalized method of moments (GMM) additionally requires estimating and inverting an often ill-conditioned covariance matrix of size $\Omega(d^L)$ -by- $\Omega(d^L)$ to obtain the optimal weighting matrix. Pereira, Kileel, and Kolda [arXiv:2202.06930 (2022)] avoid the tensor computation cost of MM through implicit moment computation. Despite this progress, incorporating statistically principled weighting while preserving computational efficiency remains an important challenge, particularly at low signal-to-noise ratio (SNR), where sampling variances can differ substantially across moment orders. We introduce diagonally-weighted GMM (DGMM), which assigns one adaptive weight to each moment order through a structured approximation of the optimal weighting matrix while avoiding the covariance inversion, thereby balancing statistical efficiency, computational complexity, and numerical stability. For general parametric models, we prove that DGMM is consistent, asymptotically normal, and near-efficient under explicit structural conditions. For heteroscedastic low-rank Gaussian mixtures, we design a computationally efficient and numerically stable algorithm that extends the implicit moment computation to the DGMM objective, gradients, and weights and incorporates Nyström approximations of the kernel sums. In low-SNR experiments with $d = 100$ and $L = 4$, where standard MM and GMM are infeasible, controlled weighting comparisons show that DGMM achieves consistently small parameter errors across tested initializations, including the challenging starts where alternative weightings struggle, with shorter median runtimes and fewer iterations. The code and data will be available upon publication at <https://github.com/liu-lzhang/dgmm>.

Keywords: Generalized Method of Moments, Gaussian Mixture Models, Numerical Tensor Algorithms, Subspace Clustering

MSC codes: 62F12, 62H30, 15A69, 65Y20

1 Introduction

Gaussian mixtures have been extensively studied for modeling high-dimensional data with cluster structures, which are ubiquitous in natural science and social science [54, 70], image processing [62],

^{*}Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544 USA (lz1619@princeton.edu).

[†]Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544 USA (hm6655@princeton.edu).

[‡]Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544 USA (sxu21@princeton.edu).

[§]Department of Mathematics and Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544 USA (amits@math.princeton.edu).

and speech processing [65]. The method of moments (MM) dates back to [59], where the first few moments were used to estimate the parameters of a mixture of two one-dimensional Gaussians. Many moment-based approaches subsequently extended the classical MM, among which the generalized method of moments (GMM)¹ has been one of the most widely studied since [36] first established the result on the optimally-weighted GMM. For textbook treatments of GMM, see, e.g., [56, 35]. In the theoretical computer science literature, [25] proposed the first provable polynomial-time algorithm for Gaussian mixture parameter estimation, under the assumption that the centers of the Gaussian components are well-separated. Since then, many provable algorithms were proposed, aiming to relax this assumption [5, 71, 16, 26]. Subsequent breakthroughs that eliminated the separation condition were independently proposed by [43] for mixtures of two Gaussians, [9] for mixtures of K Gaussians with identical spherical covariance, and [55] for mixtures of K arbitrary Gaussians. In particular, the algorithm in [55] is based on the classical MM. This prompted a renewed interest in moment-based methods and inspired many algorithms on the theme of structured decomposition of higher-order moment tensors, e.g., [41, 3, 11, 4] for isotropic Gaussian mixtures, [34] for diagonal Gaussian mixtures, and [32, 8, 50] for general Gaussian mixtures. Despite the theoretical advances, deploying moment-based methods in high-dimensional or higher-order settings is often impractical due to computational bottlenecks. The computational and storage complexity of standard MM and GMM through moment order L in d dimensions is $\Omega(d^L)$. The full GMM weighting additionally requires estimating and inverting an often ill-conditioned moment covariance matrix of size $\Omega(d^L)$ -by- $\Omega(d^L)$. [61] addressed the exponential complexity by developing an implicit moment computation for estimating Gaussian mixtures via MM, without explicitly forming the d^k -entry moment tensors. Building on the implicit moment computation introduced in [61] and the large-sample GMM theory, we address the remaining question on moment weighting: how to weight the moment conditions in a way that is statistically principled without incurring the computational bottlenecks including exponential complexity and numerical instability?

1.1 Contributions

We introduce the diagonally-weighted GMM (DGMM), which uses the optimal order-wise diagonal approximation of the full GMM weighting matrix to assign order-specific weights to moment conditions, with the following closed form:

$$\widehat{W}^{[t]} = \text{diag}(\underbrace{\widehat{w}_1^{[t]}, \dots, \widehat{w}_1^{[t]}}_{d \text{ copies}}, \dots, \underbrace{\widehat{w}_L^{[t]}, \dots, \widehat{w}_L^{[t]}}_{d^L \text{ copies}}), \quad \widehat{w}_k^{[t]} = \frac{\text{Tr} \left[\left(\widehat{S}^{[t]} \right)^{(k,k)} \right]}{\sum_{k'} \left\| \left(\widehat{S}^{[t]} \right)^{(k,k')} \right\|_F^2}, \quad (1.1)$$

where the diagonal matrix $\widehat{W}^{[t]}$ is constrained to have only L different values on its diagonal, $\widehat{S}^{[t]}$ is the estimated covariance of moment conditions at the current iterate, $(\widehat{S}^{[t]})^{(k,k')}$ is its d^k -by- $d^{k'}$ block of covariances between order- k and order- k' moment conditions. DGMM weighting is adaptive to the sampling variances across moment orders and the moment-matching residuals at the current estimation step, is near-efficient under explicit structural conditions, and can be evaluated quickly by extending the implicit moment computation to [Model 1.1](#), thereby achieving a balance among statistical efficiency, computational complexity, and numerical stability. Our contributions cover the statistical, algorithmic, and numerical aspects of DGMM:

¹In related literature, GMM is sometimes also used as an abbreviation for Gaussian mixture model. Throughout the paper, GMM will always refer to generalized method of moments and “Gaussian mixture” will always be written in full to avoid confusion.

- **Statistical aspect.** For general statistical models under standard GMM identification and regularity conditions, we establish the consistency and asymptotic normality of DGMM and derive its asymptotic covariance in [Theorem 4.1](#), and prove a conditional near-efficiency bound in [Theorem 4.2](#).
- **Algorithmic aspect.** We apply DGMM to study the parameter estimation problem for low-SNR heteroscedastic low-rank Gaussian mixture ([Model 1.1](#)) and design an algorithm to obtain the DGMM estimator with computational complexity linear in N and quadratic in the moment orders L for fixed model sizes, without explicitly computing or storing moment tensors. This is achieved by adapting the implicit moment computation machinery in [61] to the low-rank parameterization in [Model 1.1](#). This algorithm significantly reduces the exponential complexity of standard MM and GMM procedures ([Theorem 4.12](#)).
- **Numerical aspect.** We implement the algorithm and empirically validate the advantages of DGMM compared to alternative moment weighting schemes in terms of parameter estimation error, runtime, and convergence behavior in [Section 5](#).

Model 1.1 (Heteroscedastic low-rank Gaussian mixture). Fix the number of mixture components $K \geq 2$. Let $h \in [K]$ be a discrete random variable such that $\pi_j = \mathbb{P}(h = j) \in (0, 1)$ for $j = 1, \dots, K$ and $\sum_{j=1}^K \pi_j = 1$, where π_j is the mixing probability of the j -th mixture component. The j -th mixture component $\mathbf{X}_j \in \mathbb{R}^d$ is said to be a *low-rank Gaussian*² if

$$\mathbf{X}_j \sim \mathcal{N}(\boldsymbol{\mu}_j, \Sigma_j), \quad R_j := \text{rank } \Sigma_j \ll d, \quad (1.2)$$

where $\boldsymbol{\mu}_j \in \mathbb{R}^d$ and $\Sigma_j \in \mathbb{R}^{d \times d}$ are the center and the covariance of the j -th mixture component. Then the random vector $\mathbf{Y}$ drawn as $\mathbf{X}_h$ is said to be a *heteroscedastic low-rank Gaussian mixture*³ if the covariances are allowed to be non-identical, that is, $\Sigma_i \neq \Sigma_j$ for some $i \neq j$. The mixture variable $\mathbf{Y}$ has the following data generating process:

$$\mathbf{y}_n \mid (h_n = j) \sim \mathcal{N}(\boldsymbol{\mu}_j, \Sigma_j), \quad n = 1, \dots, N, \quad (1.3)$$

which can be interpreted as a two-step process of picking a component index according to mixing probabilities and sampling a low-rank Gaussian random vector from the picked component.

We use the parameterization $\Sigma_j = V_j V_j^\top$ with $V_j \in \mathbb{R}^{d \times R_j}$ to impose the low-rank constraints on the Gaussian components. The model parameters are collected in a vector concatenating the mixing probabilities, the centers, and the vectorized low-rank factors V_j , i.e., $\boldsymbol{\theta} := [\pi_1; \dots; \pi_K; \boldsymbol{\mu}_1; \dots; \boldsymbol{\mu}_K; \text{vec}(V_1); \dots; \text{vec}(V_K)] \in \Theta \subset \mathbb{R}^p$, where $p = K + dK + d \sum_{j=1}^K R_j$.

Definition 1.1 (Gaussian mixture SNR). For each $j \in [K]$, suppose that the covariance matrix of the j -th mixture component Σ_j has the following singular value decomposition

$$\Sigma_j = U_j \Lambda_j U_j^\top, \quad \Lambda_j = \text{diag}(\lambda_j^{(1)}, \dots, \lambda_j^{(R_j)}). \quad (1.4)$$

We define the signal-to-noise ratio (SNR) of a given Gaussian mixture variable $\mathbf{Y}$ to be

$$\text{SNR}(\mathbf{Y}) := \max_{j \in [K]} \frac{\|\boldsymbol{\mu}_j - \bar{\boldsymbol{\mu}}\|_2^2}{\bar{\lambda}_j}, \quad \bar{\boldsymbol{\mu}} := \sum_{j=1}^K \pi_j \boldsymbol{\mu}_j, \quad \bar{\lambda}_j := \frac{1}{R_j} \sum_{r=1}^{R_j} \lambda_j^{(r)}, \quad (1.5)$$

²For a more detailed characterization of the p.d.f. of a low-rank Gaussian, see [Section A](#).

³We use the term “low-rank Gaussian mixture” to indicate that the covariance matrix of each Gaussian component in the mixture is of low rank. We note that [52] also uses the term “low-rank Gaussian mixture” but refers to a different model where each observation of the Gaussian mixture is assumed to be matrix-valued and have a planted low-rank structure.

where $\bar{\boldsymbol{\mu}}$ is the mixture mean and $\bar{\lambda}_j$ the average variance of the j -th mixture component. A given mixture variable $\mathbf{Y}$ is said to be a *low-SNR Gaussian mixture* if $\text{SNR}(\mathbf{Y}) \ll 1$.

Remark 1.2. This definition for Gaussian mixture SNR is a generalization of [44]. When all covariances are identical, full-rank, and isotropic, that is, $\Sigma_j = \sigma^2 I_d$ for all j , we have $R_j = d$ and $\bar{\lambda}_j = \sigma^2$, and (1.5) reduces exactly to the definition of mixture SNR in [44, Definition 2.1 and Eq. 2.8], which is $\max_{j \in [K]} \frac{\|\boldsymbol{\mu}_j - \bar{\boldsymbol{\mu}}\|_2^2}{\sigma^2}$.

We study the parameter estimation problem for [Model 1.1](#) in the low-SNR regime, when the number of mixture components K and the maximum possible rank $R_{\max} \geq \max\{R_1, \dots, R_K\}$ are given and the individual ranks are unknown. A concrete example is given in [Figure 1](#).

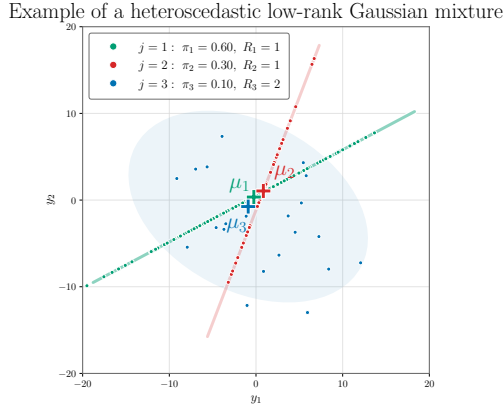


Figure 1: 200 i.i.d. observations from an example of [Model 1.1](#) in the low-SNR regime: $\mathbf{Y} \in \mathbb{R}^{100} \sim 0.6\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1) + 0.3\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2) + 0.1\mathcal{N}(\boldsymbol{\mu}_3, \Sigma_3)$, where $\text{rank } \Sigma_1 = \text{rank } \Sigma_2 = 1, \text{rank } \Sigma_3 = 2$ (plot shows the first two coordinates). This is a mixture of two Gaussians on one-dimensional support and one Gaussian on two-dimensional support. The covariance/noise scale is large relative to the center/signal scale in the sense of [Definition 1.1](#).

There are several motivations for studying this problem:

- (1) **Low-SNR:** Considering the mixture centers as the signal and the covariances as the noise, we impose the low-SNR condition in order to study the mixture estimation problem in the more challenging low-SNR regime. Related notions of low-SNR mixtures have been explored in recent works including [1, 44, 30, 29]. Note that the results on statistical properties and computational complexity of DGMM do not require the low-SNR assumption, but it serves as an important theoretical motivation.
- (2) **Heteroscedasticity:** By allowing the mixture components to have covariance matrices of different ranks and different spectral distributions, we can model data with heterogeneous noise across observations in different clusters. Such data arise when the observations are obtained under different measurement conditions, e.g., econometric data with unobserved heterogeneity [22], spectrophotometric data [20], and astronomy data [69]. Related heterogeneous noise models have been studied in [40, 39] for principal component analysis and [17] for heteroscedasticity detection.
- (3) **Low-rank:** The low-rank covariance structure in [Model 1.1](#) is particularly useful for reducing the dimension of the parameter space when the intrinsic dimension is much smaller than the ambient dimension, which is a common assumption in model-based subspace clustering

[31], structured graph learning [48], image processing [33, 63] and diffusion model training [74, 68]. More concretely, this low-rank assumption is often used in applications in finance [75], genomics [6], and spatial statistics [18].

1.2 Main results

We organize our results on DGMM in Table 1.

	MM	DGMM	GMM
Weighting	$W = I_q,$ $q = d + \dots + d^L$	$W =$ $\text{diag}(w_1 I_d, \dots, w_L I_{d^L})$	$W = S^{-1} \in \mathbb{R}^{q \times q},$ $q = d + \dots + d^L$
Statistical properties	Consistent, asymptotically normal, suboptimal efficiency	Consistent, asymptotically normal, near-efficient under structural conditions (Theorems 4.1 and 4.2)	Consistent, asymptotically normal, optimal efficiency
Weight computation	$\mathcal{O}(1)$	$\mathcal{O}(L^2 K^2 + LK^2 d R_{\max}^2$ $+ NLK + NKd R_{\max})$ (Theorem 4.10)	$\mathcal{O}(Nd^{2L} + d^{3L})$ (Theorem 4.12)
Gradient computation	$\mathcal{O}(Nd^{L+1} K R_{\max})$ (Theorem 4.12)	$\mathcal{O}(L^2 K^2 + LK^2 d R_{\max}^2$ $+ NLK + NKd R_{\max})$ (Theorem 4.11)	$\mathcal{O}(Nd^{L+1} K R_{\max}$ $+ d^{2L+1} K R_{\max})$ (Theorem 4.12)
Overall computational complexity	$\mathcal{O}(Nd^{L+1} K R_{\max})$ (Theorem 4.12)	$\mathcal{O}(L^2 K^2 + LK^2 d R_{\max}^2$ $+ NLK + NKd R_{\max}$ $+ NLdm)$ (Theorem 4.12)	$\mathcal{O}(Nd^{2L} + d^{3L}$ $+ Nd^{L+1} K R_{\max}$ $+ d^{2L+1} K R_{\max})$ (Theorem 4.12)
Numerical stability	Stable	Stable	Unstable, due to $(\hat{S}^{[t]})^{-1}$ in (2.18)

Table 1: Comparison of moment-based estimators. N : the sample size, d : the number of dimensions, K : the number of mixture components, $R_{\max} \geq \max(R_1, \dots, R_K)$: the given maximum possible rank, L : the highest moment order, m : the number of landmarks for Nyström approximation in Theorem 4.6.

1.3 Outline

We give the preliminaries in Section 2 and present the proposed DGMM estimator in Section 3. In Section 4, we detail our results on the statistical properties of the DGMM estimator and the computational complexity of our algorithm for obtaining the DGMM estimator for Model 1.1. In Section 5, we report the numerical results.

2 Preliminaries

In this section, we introduce the relevant preliminary background on tensors, moments and cumulants, the model, moment-based methods including MM and GMM.

2.1 Tensors

The definitions are based on standard references, e.g., [21, 47].

Definition 2.1 (Tensors, [47]). Let $\mathbb{V}_1, \dots, \mathbb{V}_k$ be finite-dimensional vector spaces, each with its own coordinate system. An element of the tensor product of $\mathbb{V}_1, \dots, \mathbb{V}_k$ is called a *tensor* and we say that the tensor is of order k and dimensions $(\dim \mathbb{V}_1, \dots, \dim \mathbb{V}_k)$.

For the purpose of our paper, we consider $\mathcal{X} \in \mathbb{T}^k(\mathbb{R}^d) := \underbrace{\mathbb{R}^d \otimes \dots \otimes \mathbb{R}^d}_{k \text{ copies}}$, which is the set of all real-valued tensors of order k and dimensions $(d, \dots, d)$.

Definition 2.2 (Inner product and norm on $\mathbb{T}^k(\mathbb{R}^d)$, [47]). The *inner product* of two tensors $\mathcal{X}, \mathcal{Y} \in \mathbb{T}^k(\mathbb{R}^d)$ is

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^d \dots \sum_{i_k=1}^d \mathcal{X}_{i_1 \dots i_k} \mathcal{Y}_{i_1 \dots i_k}. \quad (2.1)$$

The *norm* of a tensor $\mathcal{X} \in \mathbb{T}^k(\mathbb{R}^d)$ is

$$\|\mathcal{X}\| = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle} = \sqrt{\sum_{i_1=1}^d \dots \sum_{i_k=1}^d \mathcal{X}_{i_1 \dots i_k}^2}. \quad (2.2)$$

Definition 2.3 (Symmetric tensors, [21]). An order- k tensor $\mathcal{X} \in \mathbb{T}^k(\mathbb{R}^d)$ is *symmetric* if, for any permutation of the k indices, its entries remain unchanged, i.e.,

$$\mathcal{X}_{i_{\sigma(1)} \dots i_{\sigma(k)}} = \mathcal{X}_{i_1 \dots i_k}, \quad i_1, \dots, i_k \in [d], \quad (2.3)$$

for all permutations $\sigma \in \mathfrak{S}_k$, where $\mathfrak{S}_k$ is the symmetric group of permutations on the indices $\{1, \dots, k\}$. $\mathbb{S}^k(\mathbb{R}^d)$ denotes the set of symmetric tensors in $\mathbb{T}^k(\mathbb{R}^d)$.

Definition 2.4 (Symmetrization operator, [21]). The symmetrization operator $\text{sym} : \mathbb{T}^k(\mathbb{R}^d) \rightarrow \mathbb{S}^k(\mathbb{R}^d)$ is the linear operator that acts on $\mathcal{X} \in \mathbb{T}^k(\mathbb{R}^d)$ in the following way

$$(\text{sym}(\mathcal{X}))_{i_1 \dots i_k} = \frac{1}{k!} \sum_{\sigma \in \mathfrak{S}_k} \mathcal{X}_{i_{\sigma(1)} \dots i_{\sigma(k)}}. \quad (2.4)$$

By [21, Proposition 3.3], an equivalent reformulation of symmetric tensors is via the symmetrization operator: an order- k tensor $\mathcal{X} \in \mathbb{T}^k(\mathbb{R}^d)$ is symmetric if and only if $\text{sym}(\mathcal{X}) = \mathcal{X}$.

2.2 Moments, cumulants, Bell polynomials

We present the definitions of moments, cumulants, and Bell polynomials, as well as how they are related. Throughout the paper, we assume that every random variable is defined on a suitable probability space $(\Omega, \mathcal{F}, \mathbb{P})$.

Definition 2.5 (Moments). Suppose X is a real-valued random variable. Then the k -th *uncentered moment* of X is defined as

$$\mathcal{M}^{(k)}(X) = \mathbb{E} [X^k]. \quad (2.5)$$

Definition 2.6 (Cumulants). The k -th *cumulant* $\kappa_k(X)$ of a random variable X is defined recursively by the following relation between moments and cumulants:

$$\mathbb{E}[X^k] = \sum_{\pi \in \Upsilon_k} \prod_{S \in \pi} \kappa_{|S|}(X), \quad (2.6)$$

where Υ_k is the set of all partitions of the set $[k]$. A partition $\pi \in \Upsilon_k$ is a set of disjoint non-empty subsets S such that $\bigsqcup_{S \in \pi} S = [k]$.

Definition 2.7 (Complete Bell polynomials, [10] (cf. [19, Def. 11.1])). The *complete Bell polynomials* in the variables $x_1, \dots, x_k$ are defined by the sum

$$B_k(x_1, \dots, x_k) = \sum_{(j_1, \dots, j_k) \in \mathcal{P}_k} \frac{k!}{j_1! j_2! \cdots j_k!} \left(\frac{x_1}{1!}\right)^{j_1} \left(\frac{x_2}{2!}\right)^{j_2} \cdots \left(\frac{x_k}{k!}\right)^{j_k}, \quad (2.7)$$

where $\mathcal{P}_k = \{(j_1, \dots, j_k) : j_1 + 2j_2 + \cdots + kj_k = k, j_1, \dots, j_k \in \mathbb{Z}_{\geq 0}\}$.

Definition 2.8 (Partial Bell polynomials, [10] (cf. [19, Def. 11.2])). The *partial Bell polynomials* in the variables $x_1, \dots, x_k$ of degree l are defined by the sum

$$B_{k,l}(x_1, \dots, x_k) = \sum_{(j_1, \dots, j_k) \in \mathcal{P}_{k,l}} \frac{k!}{j_1! j_2! \cdots j_k!} \left(\frac{x_1}{1!}\right)^{j_1} \left(\frac{x_2}{2!}\right)^{j_2} \cdots \left(\frac{x_k}{k!}\right)^{j_k}, \quad (2.8)$$

where $\mathcal{P}_{k,l} = \{(j_1, \dots, j_k) : j_1 + 2j_2 + \cdots + kj_k = k, j_1 + j_2 + \cdots + j_k = l, j_1, \dots, j_k \in \mathbb{Z}_{\geq 0}\}$, i.e., the summation is taken over the partitions of k into l parts.

The following proposition relates moments, cumulants, and Bell polynomials.

Proposition 2.9 ([23, p. 160], [60, Proposition 3.3.1]). *Let X be a random variable such that $\mathbb{E}[|X|^k] < \infty$. For $l = 1, \dots, k$, let $\mathcal{M}^{(l)}(X) = \mathbb{E}[X^l]$ be the l -th moment and $\kappa^{(l)}(X)$ be the l -th cumulant. Then*

$$\mathcal{M}^{(k)}(X) = B_k(\kappa^{(1)}(X), \dots, \kappa^{(k)}(X)) \quad (2.9)$$

$$\kappa^{(k)}(X) = \sum_{l=1}^k (-1)^{l-1} (l-1)! B_{k,l}(\mathcal{M}^{(1)}(X), \dots, \mathcal{M}^{(k-l+1)}(X)). \quad (2.10)$$

2.3 Method of moments (MM) and generalized method of moments (GMM)

To apply moment-based methods, including MM and GMM, to estimate the unknown parameter $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ of [Model 1.1](#), we start from defining the *moment function* $g : \Theta \times \mathbb{R}^d \rightarrow \mathbb{R}^q$ by

$$g(\boldsymbol{\theta}; \mathbf{y}) := [\text{vec}(\mathcal{M}^{(1)}(\boldsymbol{\theta}) - \mathbf{y}); \dots; \text{vec}(\mathcal{M}^{(L)}(\boldsymbol{\theta}) - \mathbf{y}^{\otimes L})] \in \mathbb{R}^q, \quad (2.11)$$

where $q = d + d^2 + \cdots + d^L$, L is the highest moment order, and $\mathcal{M}^{(k)}(\boldsymbol{\theta})$ denotes the k -th population moment that can be computed by [61, Theorem 4.1]:

$$\mathcal{M}^{(k)}(\boldsymbol{\theta}) = \sum_{j=1}^K \sum_{l=0}^{\lfloor k/2 \rfloor} \pi_j C_{k,l} \text{sym}(\boldsymbol{\mu}_j^{\otimes(k-2l)} \otimes (V_j V_j^\top)^{\otimes l}) \in \mathcal{S}^k(\mathbb{R}^d), \quad C_{k,l} = \frac{k!}{(k-2l)! l! 2^l}. \quad (2.12)$$

Following the standard practice (see [36, 56]), we restrict the parameter space Θ to be a compact set containing the true parameters. For the purpose of this paper, we assume the standard global and local GMM identification conditions (see, e.g., [36, 56]), that is, there exists a suitable L such that [Model 1.1](#) is

- (1) *globally identifiable*, i.e., $\mathbb{E}[g(\boldsymbol{\theta}; \mathbf{y}_n)] = \mathbf{0}$ if and only if $\boldsymbol{\theta} = \boldsymbol{\theta}^*$, and
- (2) *locally identifiable*, i.e., $\text{rank}(G(\boldsymbol{\theta}^*)) = p$, where $G(\boldsymbol{\theta}^*) = \mathbb{E}[\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}^*; \mathbf{y}_n)]$.

Given N independent observations $y_1, \dots, y_N \in \mathbb{R}^d$, the corresponding vector of *sample moment conditions* is

$$\bar{g}_N(\boldsymbol{\theta}) := \frac{1}{N} \sum_{n=1}^N g(\boldsymbol{\theta}; \mathbf{y}_n) \in \mathbb{R}^q. \quad (2.13)$$

If $q = p$, then one may set up a system of polynomial equations and solve for $\boldsymbol{\theta}$ exactly:

$$\bar{g}_N(\boldsymbol{\theta}) = \mathbf{0} \in \mathbb{R}^q. \quad (2.14)$$

When $q > p$, the model is overidentified, that is, there are more sample moment conditions than the unknown parameters. In this case, one instead obtains an estimator that makes the sample moment conditions as close as possible to zero according to some metric or measure of distance. More specifically, the MM estimator, $\hat{\boldsymbol{\theta}}^{(\text{MM})}$, is obtained by solving a moment-matching optimization problem in $\boldsymbol{\theta}$:

$$\underset{\boldsymbol{\theta} \in \Theta}{\text{minimize}} \quad \|\bar{g}_N(\boldsymbol{\theta})\|_2^2 = \sum_{k=1}^L \left\| \mathcal{M}^{(k)}(\boldsymbol{\theta}) - \frac{1}{N} \sum_{n=1}^N \mathbf{y}_n^{\otimes k} \right\|^2, \quad (2.15)$$

$$\text{subject to } 0 < \pi_1, \dots, \pi_K < 1, \quad \sum_{j=1}^K \pi_j = 1. \quad (2.16)$$

The GMM estimator, $\hat{\boldsymbol{\theta}}^{(\text{GMM})}$ replaces the Euclidean metric by a weighted version:

$$\underset{\boldsymbol{\theta} \in \Theta}{\text{minimize}} \quad \bar{g}_N(\boldsymbol{\theta})^\top W \bar{g}_N(\boldsymbol{\theta}), \quad (2.17)$$

$$\text{subject to } 0 < \pi_1, \dots, \pi_K < 1, \quad \sum_{j=1}^K \pi_j = 1,$$

for symmetric positive semi-definite weighting matrix $W \in \mathbb{R}^{q \times q}$. [36] shows that the GMM estimator achieves optimal asymptotic efficiency when $W = cS^{-1}$, for any $c > 0$ and $S = \mathbb{E}[g(\boldsymbol{\theta}^*; \mathbf{y}_n)g(\boldsymbol{\theta}^*; \mathbf{y}_n)^\top]$. When $W = I$, $\hat{\boldsymbol{\theta}}^{(\text{GMM})}$ is equivalent to the MM estimator, $\hat{\boldsymbol{\theta}}^{(\text{MM})}$. In practice, $\hat{\boldsymbol{\theta}}^{(\text{GMM})}$ is obtained via a multi-step estimation procedure, where at each estimation step, a consistent estimator of S and a consistent estimator of $\boldsymbol{\theta}$ are obtained alternately:

$$\hat{S}^{[t]} = \frac{1}{N} \sum_{n=1}^N g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n)^\top, \quad (2.18)$$

$$\hat{\boldsymbol{\theta}}^{[t]} = \underset{\boldsymbol{\theta} \in \Theta}{\text{arg min}} \quad \bar{g}_N(\boldsymbol{\theta})^\top \hat{W}^{[t]} \bar{g}_N(\boldsymbol{\theta}), \quad \hat{W}^{[t]} = \left(\hat{S}^{[t]} \right)^{-1},$$

which terminates when $t = T$ or $\|\hat{\boldsymbol{\theta}}^{[t]} - \hat{\boldsymbol{\theta}}^{[t-1]}\| < \varepsilon_{\boldsymbol{\theta}}$ with stopping criteria constants $T, \varepsilon_{\boldsymbol{\theta}}$.

3 Proposed DGMM estimator

A key computational bottleneck in deploying MM and GMM in practice is that directly computing the sample moment conditions $\bar{g}_N(\boldsymbol{\theta})$ in (2.13) and the optimization objective $\bar{g}_N(\boldsymbol{\theta})^\top W \bar{g}_N(\boldsymbol{\theta})$ in (2.17) through moment order L in d dimensions incurs computational and storage cost of $\Omega(d^L)$. This quickly becomes prohibitive for high-dimensional data or when higher-order moments are required. In addition, the full GMM weighting in (2.18) requires estimating and inverting a $q \times q$ covariance matrix. When d and L are large, the number of moment conditions $q \geq d^L$, or at least $d^L/L!$ after removing redundant moment conditions, is large relative to the sample size N , making this inversion step numerically unstable. More specifically, when $q > N$, $\hat{S}^{[t]}$ is not even invertible. When $q < N$ but q/N is not negligible, $\hat{S}^{[t]}$ is invertible but has a large condition number. Prior works including [49, 66] have discussed this issue more extensively and pointed out that it is usually unrealistic in practice to find large enough sample size N to make the ratio q/N negligible. As a result, the full GMM weighting may end up degrading the parameter estimation performance, despite achieving the optimal asymptotic efficiency.

To address these computational bottlenecks, we propose the diagonally-weighted GMM (DGMM), where instead of the full GMM weighting, we restrict to the class of diagonal matrices of the form $\mathcal{W} = \{\text{diag}(\underbrace{w_1, \dots, w_1}_{d \text{ copies}}, \dots, \underbrace{w_L, \dots, w_L}_{d^L \text{ copies}}) \in \mathbb{R}^{q \times q}\}$ and find the optimal approximation of the theoretically optimal weighting matrix within $\mathcal{W}$:

$$W = \underset{W \in \mathcal{W}}{\text{arg min}} \|WS - I\|_F^2, \quad (3.1)$$

where $\|\cdot\|_F$ denotes the Frobenius norm. This particular form of weighting is motivated by the following considerations:

(1) **Structured weighting without covariance inversion.**

The S matrix describes the second-order interactions in the sample moment conditions. By using the structured approximate-inverse, DGMM circumvents the numerically unstable inversion step in (2.18) while preserving the most important order-wise interactions, at the cost of some loss in asymptotic efficiency that we will precisely quantify and prove a conditional near-efficiency bound in Theorem 4.2. This efficiency gap between DGMM and the optimally-weighted GMM is reduced when the cross-order correlations across sample moment conditions of different orders are weak and when the sample moment conditions of the same order share comparable sampling variances.

(2) **Weighting across moment orders is especially relevant in the low-SNR regime.**

For a fixed sample size, the sampling variances of the sample moment conditions can grow rapidly with the moment order (see, e.g., [73, Lemma 5]). The differences in sampling variances become especially prominent in the low-SNR regime. With identity weighting, the noisiest moment order may dominate the objective, resulting in suboptimal statistical efficiency and numerical conditioning issues. DGMM weighting mitigates this issue through the order-specific weights in (3.1). This is conceptually aligned with [44, Theorem 2.2], which shows that, in the low-SNR identical-covariance setting, the population likelihood expansion is organized as a hierarchy of moment-matching discrepancies indexed by moment order.

(3) **Estimating fewer weights reduces finite-sample bias.**

Estimating the GMM weights can produce substantial finite-sample bias. This phenomenon has been documented in many previous studies in econometrics, e.g., [37, 2, 57]. Therefore, estimating only L weights empirically reduces the finite-sample bias.

- (4) **Computational complexity considerations.** An unrestricted diagonal version of (3.1) estimates $q = \Omega(d^L)$ coordinate-specific weights, whereas DGMM uses one weight for each moment order. This reduces the number of weights from q to L and allows for implicit moment computation speed-up, as shown in Theorem 4.4-Theorem 4.12.

Intuitively, DGMM weighting is a structured regularization of the full GMM weighting. It prioritizes the order-wise weighting to account for potentially large differences in sampling variances across moment orders. In this way, DGMM addresses the statistical-computational trade-offs in moment-based estimation. More specifically, it trades some potential efficiency gain from additional weighting for substantially lower computational and storage costs, improved numerical stability, and reduced finite-sample bias from weight-estimation error.

To obtain a closed-form solution for w_k in (3.1), we use the first-order optimality condition:

$$\begin{aligned} 0 &= \nabla_{w_k} \sum_{i \in \mathcal{I}_k} \sum_{j=1}^q (w_k S_{ij} - I_{ij})^2 = 2 \sum_{i \in \mathcal{I}_k} \sum_{j=1}^q (w_k S_{ij} - I_{ij}) \nabla_{w_k} (w_k S_{ij}) \\ &= 2 \sum_{i \in \mathcal{I}_k} \sum_{j=1}^q (w_k S_{ij} - I_{ij}) S_{ij} = 2 \sum_{i \in \mathcal{I}_k} \left(\sum_{j=1}^q w_k S_{ij}^2 - S_{ii} \right), \end{aligned} \quad (3.2)$$

where $\mathcal{I}_k$ denotes the set of indices in S for the k -th moment, which gives the desired weights

$$w_k = \frac{\sum_{i \in \mathcal{I}_k} S_{ii}}{\sum_{i \in \mathcal{I}_k} \sum_{j=1}^q S_{ij}^2} = \frac{\text{Tr} [S^{(k,k)}]}{\sum_{k'} \|S^{(k,k')}\|_F^2}, \quad k = 1, \dots, L, \quad (3.3)$$

where $S^{(k,k')}$ is the d^k -by- $d^{k'}$ submatrix of S , corresponding to the cross-order covariance between the k -th order and k' -th order moment conditions

$$S^{(k,k')} := (S_{ij})_{i \in \mathcal{I}_k, j \in \mathcal{I}_{k'}} = \mathbb{E} \left[g^{(k)}(\boldsymbol{\theta}^*; \mathbf{y}_n) g^{(k')}(\boldsymbol{\theta}^*; \mathbf{y}_n)^\top \right]. \quad (3.4)$$

We obtain the DGMM estimator $\hat{\boldsymbol{\theta}}^{(\text{DGMM})}$ via a multi-step procedure similar to (2.18):

$$\begin{aligned} \hat{S}^{[t]} &= \frac{1}{N} \sum_{n=1}^N g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n)^\top, \\ \hat{\boldsymbol{\theta}}^{[t]} &= \underset{\boldsymbol{\theta} \in \Theta}{\text{arg min}} \quad Q_N^{[t]}(\boldsymbol{\theta}) := \bar{g}_N(\boldsymbol{\theta})^\top \widehat{W}^{[t]} \bar{g}_N(\boldsymbol{\theta}), \\ \widehat{W}^{[t]} &= \text{diag}(\underbrace{\hat{w}_1^{[t]}, \dots, \hat{w}_1^{[t]}}_{d \text{ copies}}, \dots, \underbrace{\hat{w}_L^{[t]}, \dots, \hat{w}_L^{[t]}}_{d^L \text{ copies}}), \quad \hat{w}_k^{[t]} = \frac{\text{Tr} \left[\left(\hat{S}^{[t]} \right)^{(k,k)} \right]}{\sum_{k'} \left\| \left(\hat{S}^{[t]} \right)^{(k,k')} \right\|_F^2}, \\ \text{subject to } & 0 < \pi_1, \dots, \pi_K < 1, \quad \sum_{j=1}^K \pi_j = 1. \end{aligned} \quad (3.5)$$

4 Statistical properties and computational complexity

Since DGMM weighting applies broadly to other parametric models beyond Model 1.1, we state our results on the asymptotic properties of the DGMM estimator in the general form in Theorem 4.1 and Theorem 4.2. Subsequent computational complexity results are specialized to the context of Model 1.1.

Theorem 4.1 (Statistical properties of $\hat{\boldsymbol{\theta}}^{(DGMM)}$). *Under the standard GMM global and local identification and regularity conditions, the DGMM estimator $\hat{\boldsymbol{\theta}}^{(DGMM)}$ has the following statistical properties:*

- (1) (Consistency) $\hat{\boldsymbol{\theta}}^{(DGMM)} \xrightarrow{p} \boldsymbol{\theta}^*$ as $N \rightarrow \infty$, where $\xrightarrow{p}$ denotes convergence in probability.
- (2) (Asymptotic normality) $\sqrt{N}(\hat{\boldsymbol{\theta}}^{(DGMM)} - \boldsymbol{\theta}^*) \xrightarrow{d} \mathcal{N}(0, V^{(DGMM)})$, where $\xrightarrow{d}$ denotes convergence in distribution.
- (3) (Asymptotic covariance) With the weighting given in (3.3), the asymptotic covariance matrix of the DGMM estimator is

$$V^{(DGMM)} = \left[\sum_{k=1}^L w_k G^{(k)\top} G^{(k)} \right]^{-1} \left[\sum_{k=1}^L \sum_{k'=1}^L w_k w_{k'} G^{(k)\top} S^{(k,k')} G^{(k')} \right] \left[\sum_{k=1}^L w_k G^{(k)\top} G^{(k)} \right]^{-1}, \quad (4.1)$$

where $G^{(k)}$ is the d^k -by- p submatrix of $G := G(\boldsymbol{\theta}^*) = \mathbb{E}[\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}^*; \mathbf{y}_n)]$ corresponding to the k -th order moment conditions.

Proof. See the corresponding proof in [Appendix B](#) on page 24.

Proof sketch. The proof uses standard multi-step GMM asymptotic theory. We first check the necessary conditions and then apply [56, Theorem 2.6] and [56, Theorem 3.4] to get consistency and asymptotic normality, respectively. Multiplication in moment blocks then gives the asymptotic efficiency in (4.1).

Theorem 4.2 (Conditional near-efficiency of the DGMM estimator). *Under the assumptions of Theorem 4.1, suppose that there exists $0 \leq \eta < 1$ such that, for all $k = 1, \dots, L$,*

$$\left\| w_k S^{(k,k)} - I_{d^k} \right\|_{\text{op}} + \sum_{k' \neq k} \sqrt{w_k w_{k'}} \left\| S^{(k,k')} \right\|_{\text{op}} \leq \eta, \quad (4.2)$$

where $\|\cdot\|_{\text{op}}$ denotes the operator norm. Then

$$V^{(GMM)} \preceq V^{(DGMM)} \preceq \left(1 + \frac{\eta^2}{1 - \eta^2} \right) V^{(GMM)}, \quad V^{(GMM)} = (G^\top S^{-1} G)^{-1}. \quad (4.3)$$

Proof. See the corresponding proof in [Appendix B](#) on page 25.

Proof sketch. The lower bound is a known result in large-sample GMM theory. After applying a change of coordinates $\tilde{S} = W^{\frac{1}{2}} S W^{\frac{1}{2}}$ and $\tilde{G} = W^{\frac{1}{2}} G$, we can then bound the spectrum of $\tilde{S}$ in $[1 - \eta, 1 + \eta]$ and relate the sandwich forms of $V^{(DGMM)}$ and $V^{(GMM)}$ using a complete-the-square argument inspired by the proof in [36, Theorem 3.2].

Remark 4.3. As the cross-order correlations across sample moment conditions become weaker and as sample moment conditions of the same order share comparable sampling variances, $\eta \rightarrow 0$ and the efficiency gap $V^{(DGMM)} - V^{(GMM)}$ is reduced. When S is exactly block diagonal and each within-order block is a scalar multiple of the identity, $S^{(k,k')} = 0$ for $k \neq k'$ and $S^{(k,k)} = \sigma_k^2 I_{d^k}$, the DGMM weights in (3.3) evaluate to $w_k = \sigma_k^{-2}$ and the condition (4.2) holds for all $k = 1, \dots, L$ with $\eta = 0$. This is when the DGMM estimator is exactly efficient.

Having established the statistical properties of the DGMM estimator in general, we provide a computationally efficient algorithm to obtain $\hat{\theta}^{(\text{DGMM})}$ specializing to [Model 1.1](#). The main idea is to compute the DGMM weights $\hat{w}_k^{[t]}$ and the analytic gradients of the DGMM objective $Q_N^{[t]}(\theta)$ in [\(3.5\)](#) from these quantities that can be obtained via implicit moment computation:

$$\phi^{(k)}(\theta) := \left\| \mathcal{M}^{(k)}(\theta) \right\|^2, \psi^{(k)}(\theta; \mathbf{y}_n) := \left\langle \mathcal{M}^{(k)}(\theta), \mathbf{y}_n^{\otimes k} \right\rangle, C_{k,n,n'} := \left\langle \mathbf{y}_n^{\otimes k}, \mathbf{y}_{n'}^{\otimes k} \right\rangle, \quad (4.4)$$

whose computational complexities are shown in [Theorems 4.4](#) to [4.6](#). [Theorems 4.4](#) and [4.5](#) are specializations of [\[61, Theorem 4.2 and Theorem 4.3\]](#) to the low-rank parameterization $\Sigma_j = V_j V_j^\top$ of [Model 1.1](#). Theorems related to computing the inner product sum in [Theorem 4.6](#) and [Lemmas 4.7](#) and [4.8](#) and the subsequent use of these primitives to derive the complexity of DGMM estimator in [Theorems 4.10](#) to [4.12](#) are new results of this work.

Theorem 4.4 (Computational complexity of $\phi^{(k)}(\theta)$ and its gradients).

$$\phi^{(k)}(\theta) = \sum_{i=1}^K \sum_{j=1}^K \pi_i \pi_j \left\langle \mathcal{M}^{(k)}(\theta_i), \mathcal{M}^{(k)}(\theta_j) \right\rangle = \sum_{i=1}^K \sum_{j=1}^K \pi_i \pi_j B_k \left(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k)} \right) \quad (4.5)$$

and can be computed in $\mathcal{O}(k^2 K^2 + k K^2 d R_{\max}^2)$ operations. Its gradients with respect to the parameters $\nabla_{\pi_j} \phi^{(k)}(\theta), \nabla_{\mu_j} \phi^{(k)}(\theta), \nabla_{V_j} \phi^{(k)}(\theta)$ can each be computed in $\mathcal{O}(k^2 K + k K d R_{\max}^2)$.

Proof. See the corresponding proof in [Appendix B](#) on page 26.

Theorem 4.5 (Computational complexity of $\psi^{(k)}(\theta; \mathbf{y}_n)$ and its gradients).

$$\psi^{(k)}(\theta; \mathbf{y}_n) = \sum_{j=1}^K \pi_j \left\langle \mathcal{M}^{(k)}(\theta_j), \mathbf{y}_n^{\otimes k} \right\rangle = \sum_{j=1}^K \pi_j B_k \left(\mathbf{y}_n^\top \mu_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0 \right) \quad (4.6)$$

and can be computed in $\mathcal{O}(kK + K d R_{\max})$ operations. Its gradients $\nabla_{\pi_j} \psi^{(k)}(\theta; \mathbf{y}_n), \nabla_{\mu_j} \psi^{(k)}(\theta; \mathbf{y}_n)$, and $\nabla_{V_j} \psi^{(k)}(\theta; \mathbf{y}_n)$ can each be computed in $\mathcal{O}(kK + K d R_{\max})$ operations.

Proof. See the corresponding proof in [Appendix B](#) on page 28.

The next three results concern the computation of the polynomial kernel row sums $\sum_{n'} C_{k,n,n'}$ that are used to compute the DGMM weights. In [Theorem 4.6](#), we derive the complexity of approximately computing the row sums via randomized Nyström approximation, with the detailed computational steps given in the proof. In [Lemma 4.7](#), we apply the kernel k -means++ guarantee of [\[58\]](#) to derive an expected error bound for the kernel approximation. Finally, [Lemma 4.8](#) is an algebraic bound on the kernel rank under [Model 1.1](#).

Theorem 4.6 (Computational complexity of $\sum_{n'} C_{k,n,n'}$). Define the polynomial kernel $h^{(k)}(\mathbf{y}_n, \mathbf{y}_{n'}) = \langle \mathbf{y}_n, \mathbf{y}_{n'} \rangle^k$ and its associated Gram matrix $H^{(k)} \in \mathbb{R}^{N \times N}$, where $H_{n,n'}^{(k)} = \langle \mathbf{y}_n, \mathbf{y}_{n'} \rangle^k$ for $\mathbf{y}_n, \mathbf{y}_{n'} \stackrel{\text{i.i.d.}}{\sim}$ [Model 1.1](#). Using Nyström approximation⁴ with m landmarks from kernel k -means++ sampling as in [\[58\]](#), $\sum_{n'} C_{k,n,n'}$ can be approximated in $\mathcal{O}(Ndm)$ operations, with the expected error bound for the Nyström approximation given in [Lemma 4.7](#).

Proof. See the corresponding proof in [Appendix B](#) on page 28.

Proof sketch. Since we only require an approximation of the row sums, neither the $N \times N$ kernel matrix nor the kernel approximation is formed. The dominant computation is landmark selection and evaluations between N observations and m landmarks, which cost $\mathcal{O}(Ndm)$.

⁴Standard treatments of Nyström-based kernel approximation can be found in [\[72, 28\]](#).

Lemma 4.7 (Nyström approximation expected error bound). *Let $H_m^{(k)}$ be the optimal rank- m approximation of $H^{(k)}$ and $\widetilde{H}_m^{(k)}$ be the Nyström approximation of $H^{(k)}$ using m landmarks from kernel k -means++ sampling as in [58]. Denote by $\lambda_i^{(k)}$ the i -th eigenvalue of $H^{(k)}$. Suppose that $H^{(k)}$ exhibits polynomial spectral decay,⁵ i.e., $\lambda_i^{(k)} \in \mathcal{O}(i^{-a})$ with $a > 1$ and that the clustering potential condition of [58] holds, i.e.,*

$$\phi(C^\star | U_m) \leq \sqrt{N - m} \|H^{(k)} - H_m^{(k)}\|_F, \quad (4.7)$$

where $C^\star$ is an optimal $(m + 1)$ -clustering, U_m is the span of the top m eigenvectors of $H^{(k)}$, and $\phi(C^\star | U_m)$ is the clustering potential ([27, 15]) given by the projections of $C^\star$ onto the subspace U_m . Then we have the following expected error bound:

$$\mathbb{E} \left[\left\| H^{(k)} - \widetilde{H}_m^{(k)} \right\|_F \right] \in \mathcal{O} \left(\ln(m) \sqrt{N - m} \left(m^{-a+1/2} \right) \right), \quad (4.8)$$

where the expectation is taken over random choice of landmarks. In particular, $\mathbb{E} \left[\left\| H^{(k)} - \widetilde{H}_m^{(k)} \right\|_F \right] \rightarrow 0$ as $m \rightarrow \text{rank } H^{(k)}$, which has the algebraic bound in Lemma 4.8.

Proof. See the corresponding proof in Appendix B on page 29.

Proof sketch. By applying [58, Theorem 5] and bounding $\|H^{(k)} - H_m^{(k)}\|_F$ by the eigenvalue decay via an integral test, we get (4.8).

Lemma 4.8 (Algebraic bound on $\text{rank } H^{(k)}$). *For observations of Model 1.1, $\text{rank } H^{(k)} \leq \sum_{j=1}^K \binom{R_j+k}{k} \leq K \binom{R_{\max}+k}{k}$. If all mixture component supports $\mu_j + \text{range}(\Sigma_j)$ are contained in a common linear subspace U with $\dim U \leq R_{\max}$, then $\text{rank } H^{(k)} \leq \binom{R_{\max}+k-1}{k}$.*

Proof. See the corresponding proof in Appendix B on page 29.

Proof sketch. On the j -th component support, $\mathbf{y} \in U_j \subseteq \text{span}\{\mu_j\} + \text{range}(\Sigma_j)$, whose dimension is at most $R_j + 1$. Therefore, $\mathbf{y}^{\otimes k}$ span at most $\binom{R_j+k}{k}$ directions, the number of distinct monomials of degree k in $R_j + 1$ variables. If all observations lie in a common linear subspace U of dimension at most $R_{\max}$, the bound becomes $\text{rank } H^{(k)} \leq \binom{R_{\max}+k-1}{k}$.

Remark 4.9. To interpret the bounds in Lemmas 4.7 and 4.8 in practice, the spectral decay of $H^{(k)}$ and the cluster structure in the data should be considered.

- A faster spectral decay leads to a faster expected error decay, which implies that a smaller number of landmarks m is required to achieve the same approximation error. For i.i.d. data from Model 1.1, we observe that $H^{(k)}$ exhibits fast spectral decay. Consequently, the actual number of landmarks required is considerably smaller than the algebraic bound in Lemma 4.8. However, for data whose corresponding $H^{(k)}$ does not exhibit spectral decay, the number of landmarks required will be closer to the algebraic bound in Lemma 4.8.
- The clustering potential $\phi(C^\star | U_m)$ in (4.7) is the sum of within-cluster variances under the optimal cluster assignment after projecting onto U_m , which measures how “clusterable” the data is. For i.i.d. data from Model 1.1, the inherent cluster structure means that the points are still concentrated around their respective centroids after projecting onto U_m , leading to small clustering potential. However, for data without a cluster structure, the clustering potential condition (4.7) might not hold.

⁵This is a typical assumption for Nyström approximation. See, e.g., [58, Section 3] and [7, Section 4.3].

Combining the preceding results, we obtain [Theorems 4.10 to 4.12](#). These are original results of the present work, where we derive the complexity of DGMM weights and gradients, as well as the overall complexity of the DGMM estimator in comparison to standard MM and GMM.

Theorem 4.10 (Computational complexity of $\hat{w}_k^{[t]}$). *The proposed weights $\hat{w}_k^{[t]}$ in (3.5) can be expressed in terms of the quantities in (4.4):*

$$\hat{w}_k^{[t]} = \frac{N \sum_n \left(\phi_k^{[t-1]} - 2\psi_{k,n}^{[t-1]} + C_{k,n,n} \right)}{\sum_{n,n'} \sum_{k'} \left(\phi_k^{[t-1]} - \psi_{k,n}^{[t-1]} - \psi_{k',n'}^{[t-1]} + C_{k,n,n'} \right) \left(\phi_{k'}^{[t-1]} - \psi_{k',n}^{[t-1]} - \psi_{k',n'}^{[t-1]} + C_{k',n,n'} \right)}, \quad (4.9)$$

where $\phi_k^{[t-1]} := \phi^{(k)}(\hat{\theta}^{[t-1]})$, $\psi_{k,n}^{[t-1]} := \psi^{(k)}(\hat{\theta}^{[t-1]}; \mathbf{y}_n)$. Further, $\hat{w}_k^{[t]}$ can be computed in $\mathcal{O}(L^2 K^2 + LK^2 dR_{\max}^2 + NLK + NKdR_{\max})$ operations.

Proof. See the corresponding proof in [Appendix B](#) on page 30.

Theorem 4.11 (Computational complexity of $\nabla_{\theta} Q_N^{[t]}(\theta)$). *At each iteration of the first-order solver, the gradients $\nabla_{\theta} Q_N^{[t]}(\theta)$ can be expressed in terms of the quantities in (4.4):*

$$\nabla_{\theta} Q_N^{[t]}(\theta) = \sum_{k=1}^L \hat{w}_k^{[t]} \left[\nabla_{\theta} \phi^{(k)}(\theta) - \frac{2}{N} \sum_{n=1}^N \nabla_{\theta} \psi^{(k)}(\theta; \mathbf{y}_n) \right]. \quad (4.10)$$

Further, $\nabla_{\theta} Q_N^{[t]}(\theta)$ can be computed in $\mathcal{O}(L^2 K^2 + LK^2 dR_{\max}^2 + NLK + NKdR_{\max})$.

Proof. See the corresponding proof in [Appendix B](#) on page 30.

Finally, we compare the computational complexity of DGMM with MM and GMM:

Theorem 4.12 (Overall computational complexity of DGMM). *The DGMM estimator in (3.5) is obtained in $\mathcal{O}(L^2 K^2 + LK^2 dR_{\max}^2 + NLK + NKdR_{\max} + NLdm)$ operations. In contrast, the MM estimator is obtained in $\mathcal{O}(Nd^{L+1}KR_{\max})$ operations, and the GMM estimator is obtained in $\mathcal{O}(Nd^{2L} + d^{3L} + Nd^{L+1}KR_{\max} + d^{2L+1}KR_{\max})$ operations.*

Proof. See the corresponding proof in [Appendix B](#) on page 31.

5 Numerical studies

We test the DGMM procedure in (3.5) on parameter estimation for low-SNR heteroscedastic low-rank mixtures in [Model 1.1](#), given the number of components K , the maximum rank $R_{\max}$, and $\{\mathbf{y}_n\}_{n=1}^N \in \mathbb{R}^d$ i.i.d. [Model 1.1](#). We consider two cases in [Table 2](#): (1) when the covariances of the mixture components share identical rank, i.e., $R_i = R_j$ for all $i, j \in [K]$, with $R_1 = R_2 = R_3 = R_{\max} = 3$, and (2) when the covariances of the mixture components have non-identical ranks, i.e., $R_i \neq R_j$ for some i, j , with $(R_1, R_2, R_3) = (1, 3, 4)$, $R_{\max} = 5$. We generate $N = 10^5$ observations in dimension $d = 100$ from a $K = 3$ mixture with mixing probabilities $\pi^* = (0.2, 0.3, 0.5)$. The highest moment order is $L = 4$. Ground-truth mixtures are generated in the following steps:

- $\mu_j^* \sim \text{Unif}(\{x \in \mathbb{R}^d : \|x\|_2 = 1\})$,
- $\Sigma_j^* = U_j^* \Lambda_j^* U_j^{*\top}$, where U_j^* is a $d \times R_j$ random orthonormal matrix and $\Lambda_j^* = \text{diag}\left(\left(\lambda_j^{(1)*}\right)^2, \dots, \left(\lambda_j^{(R_j)*}\right)^2\right)$, $\lambda_j^{(1)*}, \dots, \lambda_j^{(R_j)*} \sim \text{Unif}(\lambda_{\min}, \lambda_{\max})$ and $\lambda_{\min}, \lambda_{\max} \gg 1$. We use $\lambda_{\min} = 5, \lambda_{\max} = 10$ to enforce the low-SNR condition in [Definition 1.1](#). [Table 2](#) reports the empirical SNR of the ground-truth mixtures.

Ten different random initializations are generated in the following steps:

- $\pi_j^{[0]} = \frac{1}{K}$ for all j , that is, the initial guess is a uniform mixture,
- $\mu_j^{[0]} \sim \text{Unif}(\{x \in \mathbb{R}^d : \|x\|_2 = 1\})$,
- $\Sigma_j^{[0]} = U_j^{[0]} U_j^{[0]\top}$, where $U_j^{[0]}$ is a $d \times R_{\max}$ random orthonormal matrix.

In both sets of experiments, we test standard MM and GMM on the same parameter estimation task. To isolate the effect of weighting, we compare DGMM adaptive weights against identity weights and augmented-system weights in [61], using the same data, initializations, and optimization framework in (3.5) except for the weights:

- (1) **Standard MM:** solving (2.15) with explicit moment computation.
- (2) **Standard GMM:** solving (2.18) with explicit moment computation.
- (3) **DGMM framework with identity weights:** solving multi-step procedure in (3.5) with implicit moment computation and fixed weights $\hat{w}_1^{[t]} = \dots = \hat{w}_L^{[t]} = 1$ for all t .
- (4) **DGMM framework with [61]’s augmented-system weights:** solving multi-step procedure in (3.5) with implicit moment computation and fixed weights $\hat{w}_k^{[t]} = \binom{L}{k} \omega^{2(L-k)}$ for all t and $k = 1, \dots, L$, where $\omega = 0.5$ following the choice in [61, Section 7]. In particular, the set of weights from [61] is $(0.0625, 0.375, 1, 1)$ for $L = 4$.
- (5) **DGMM framework with the proposed adaptive weights:** solving multi-step procedure

in (3.5) with implicit moment computation and adaptive weights $\hat{w}_k^{[t]} = \frac{\text{Tr}[(\hat{S}^{[t]})^{(k,k)}]}{\sum_{k'} \left\| (\hat{S}^{[t]})^{(k,k')} \right\|_F^2}$.

DGMM weights are re-estimated from $\hat{S}^{[t]}$ at each step and in these two low-SNR experiments, $\hat{w}_1^{[t]} > \hat{w}_3^{[t]} > \hat{w}_2^{[t]} > \hat{w}_4^{[t]}$, decreasing within each parity class. Odd-order weights are larger than the even-order weights since $\hat{w}_k^{[t]}$ is approximately the inverse sampling variance of the noisiest fourth-order conditions, divided by their squared correlation with the order- k conditions, which is weaker for odd k .

Across all methods, we set the stopping criteria for the multi-step estimation: the maximum number of estimation steps $T = 10$ and the convergence tolerance $\varepsilon_\theta = 10^{-4}$, whichever is met first. Within each step, we re-parameterize the constrained optimization problem in (3.5) to obtain an equivalent unconstrained problem for computational simplicity, using the softmax function⁶ $\pi_j = \frac{\exp\{(\tilde{\pi}_j - \tilde{\pi}_{\max})/\tau\}}{\sum_{j'=1}^K \exp\{(\tilde{\pi}_{j'} - \tilde{\pi}_{\max})/\tau\}}$, where τ is a regularization parameter that improves numerical stability and convergence behavior by controlling how “peaked” the distribution of the mixing probabilities becomes after the softmax re-parameterization. We then solve the unconstrained problem using L-BFGS [51] with the analytic gradients in (4.10) and the maximum number of L-BFGS iterations $I = 300$. All experiments are run on a 2020 MacBook Pro (Apple M1 processor, 16GB RAM).

⁶The softmax re-parameterization and regularization parameter are standard machine learning practices, for which the theoretical justifications can be found in prior works including [12, 13, 14, 38, 42, 53].

Case	d	N	K	L	True ranks	Fitted $R_{\max}$	π^*	SNR
Identical covariance ranks	100	100000	3	4	(3, 3, 3)	3	(0.2, 0.3, 0.5)	0.023728
Non-identical covariance ranks	100	100000	3	4	(1, 3, 4)	5	(0.2, 0.3, 0.5)	0.019681

Table 2: Summary of the experimental set-ups.

The metrics for comparison are (1) computational performance measured by the average runtime and total L-BFGS iterations, and (2) statistical performance measured by the average relative estimation error with the permutation $\hat{\sigma} \in \mathbf{arg\,min}_{\sigma \in S_K} \frac{1}{K} \sum_{j=1}^K \frac{\|\hat{\Sigma}_j - \Sigma_{\sigma(j)}^*\|_F}{\|\Sigma_{\sigma(j)}^*\|_F}$:

$$\text{Err}_\pi = \frac{1}{K} \sum_{j=1}^K \frac{|\hat{\pi}_j - \pi_{\hat{\sigma}(j)}^*|}{|\pi_{\hat{\sigma}(j)}^*|}, \text{Err}_\mu = \frac{1}{K} \sum_{j=1}^K \frac{\|\hat{\mu}_j - \mu_{\hat{\sigma}(j)}^*\|_2}{\|\mu_{\hat{\sigma}(j)}^*\|_2}, \text{Err}_\Sigma = \frac{1}{K} \sum_{j=1}^K \frac{\|\hat{\Sigma}_j - \Sigma_{\hat{\sigma}(j)}^*\|_F}{\|\Sigma_{\hat{\sigma}(j)}^*\|_F}. \quad (5.1)$$

5.1 Results

In the case of same covariance ranks, we run each method for ten different random initializations starting from fixed initialization across methods and given the number of components $K = 3$ and $R_1 = R_2 = R_3 = R_{\max} = 3$. The flexibility of [Model 1.1](#) allows us to study the case when different mixture components have covariance matrices of different ranks with the maximum possible rank $R_{\max} = 5$ (individual ranks are not known a priori). Standard MM and GMM with explicit moment computation, methods (1) and (2), are also attempted, but are computationally infeasible to run in both cases at this scale. We therefore focus on examining the effects of different moment weightings in methods (3), (4), and (5).

In terms of estimation error ([Figure 2](#)), in both cases, DGMM adaptive weights lead to narrower error distributions and uniformly small parameter errors from all ten different random initializations, including the more difficult initializations from which the identity weights and the augmented-system weights of [\[61\]](#) lead to poor solutions. The parameter errors show that while favorable starts may produce comparable errors for all weights, DGMM adaptive weights make the estimation less sensitive to bad initializations than the identity weighting and the augmented-system weighting. It is worth noting that the experiments in [\[61, Section 7.3\]](#) use unit-norm means and diagonal covariances with entries of size at most 0.4, which gives a much larger SNR than the mixtures considered here. The numerical results suggest that difficult initializations may be more common at low SNR, making statistically principled moment weighting particularly crucial in this regime.

In terms of runtime ([Figure 3](#)), in both cases, over the ten trials with different random initializations, the DGMM adaptive weights require fewer total L-BFGS iterations for [\(3.5\)](#), thus leading to a faster median runtime than the identity weights and augmented-system weights. This observed improvement in the convergence behavior suggests that the DGMM adaptive weights not only improve the statistical efficiency in theory but also improve the numerical conditioning during the multi-step GMM optimization by providing the appropriate scaling for each moment order.

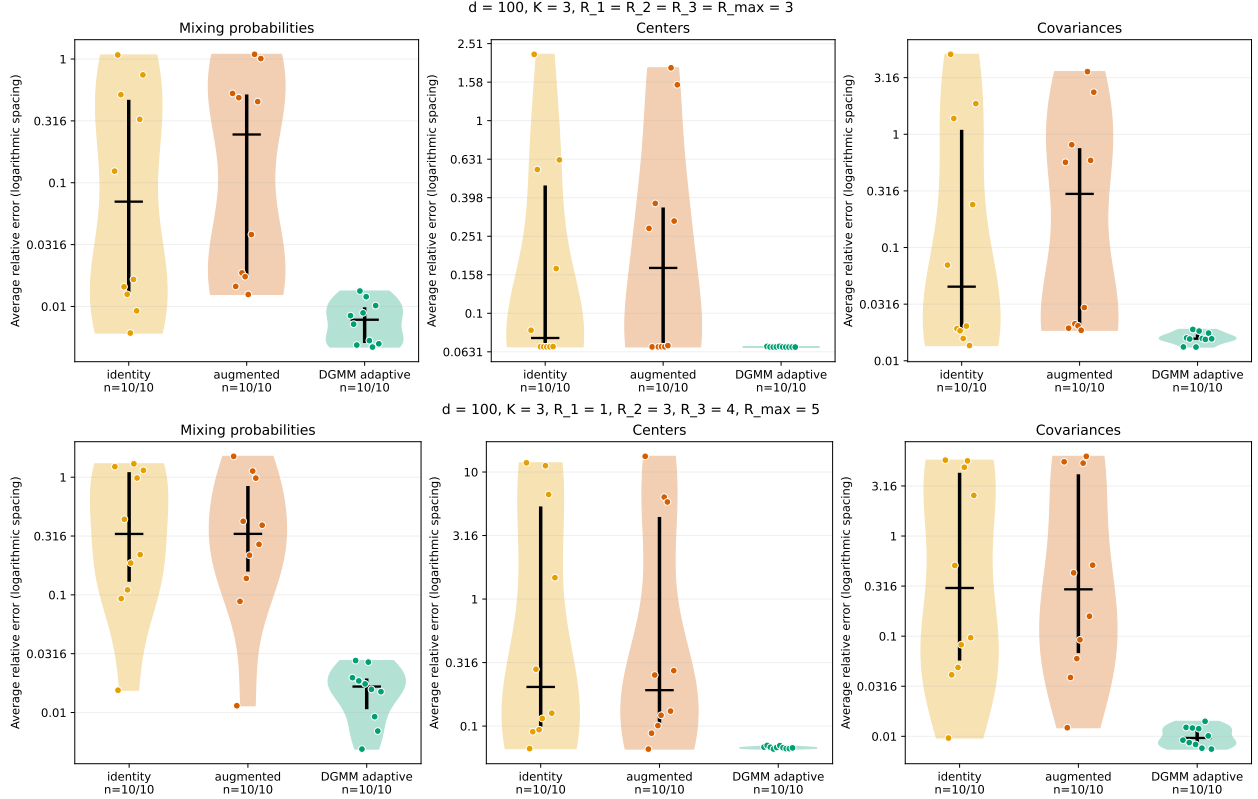


Figure 2: Distribution of average relative parameter errors Err_π , Err_μ , and Err_Σ in (5.1) in logarithmic scale for DGMM with identity, augmented-system, and adaptive weights (methods (3), (4), and (5)), from the ten different random initializations on one fixed data set, under Case 1 identical covariance ranks (above) and Case 2 non-identical covariance ranks (below). Dots represent individual trials; shaded regions are kernel density estimates of the logarithms of the errors; black vertical bars and horizontal ticks indicate interquartile ranges and medians.

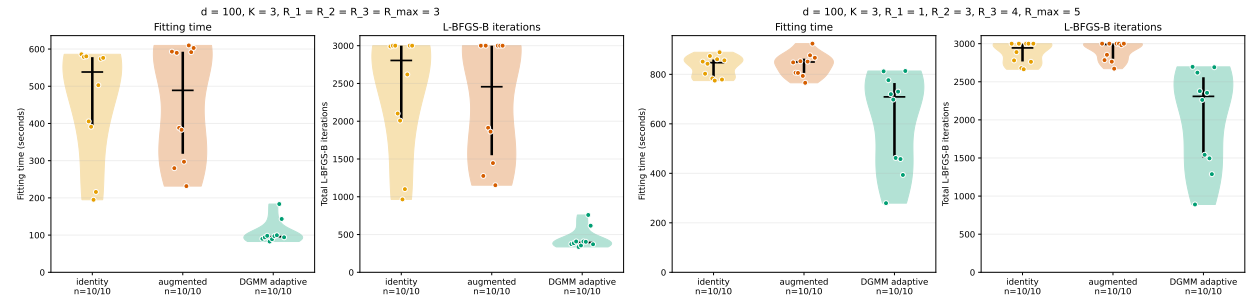


Figure 3: Distribution of fitting runtimes in seconds and total number of L-BFGS iterations summed over all estimation steps capped at $TI = 3000$, under case 1 identical covariance ranks (left) and case 2 non-identical covariance ranks (right). Methods, initializations, and plot conventions as in Figure 2.

6 Conclusion

We proposed the diagonally-weighted GMM (DGMM) to address the challenge of balancing statistical efficiency, computational complexity, and numerical stability in moment-based estimation. The key insight was to use a diagonal weighting matrix that assigns less weight to noisier sample moment

conditions, thereby improving the estimation accuracy relative to the unweighted counterpart. We developed this insight further by deriving the statistical properties of the DGMM estimator in [Theorem 4.1](#), showing that it is consistent and asymptotically normal and near-efficient relative to optimally-weighted GMM under explicit structural conditions in [Theorem 4.2](#). Notably, this weighting strategy is conceptually aligned with recent results on likelihood and moment expansion in related low-SNR problems, e.g., [1, 44, 30, 29], reinforcing the theoretical soundness of our approach.

To a more practical end, we provided a computationally efficient and numerically stable algorithm for obtaining the DGMM estimator in the context of low-SNR heteroscedastic low-rank Gaussian mixture ([Model 1.1](#)). Our algorithm leveraged two main techniques to improve the computational complexity. First, by exploiting the relation between moments, cumulants, and Bell polynomials, we avoided explicitly computing or storing the moment tensors. This significantly lowered the cost of computing weights and gradients, effectively extending the approach of [61] to the settings of [Model 1.1](#) under a weighted moment-matching objective function. Second, we applied Nyström approximation with k -means++ landmarks [58] to efficiently approximate the inner-product kernel used in weight computation, with a provably bounded expected approximation error for kernel k -means++ landmarks ([Lemma 4.7](#)). In the numerical studies, DGMM with adaptive weights attained accurate estimates from all ten different random initializations and required fewer iterations and runtime, whereas the standard MM and GMM procedures were infeasible to run and alternative weighting schemes under the same solver settings were sensitive to the initialization.

While we have chosen to focus on applying DGMM for Gaussian mixture modeling due to its ubiquitous applications, the utility of DGMM extends to more general settings. In fact, the advantages of DGMM discussed in [Section 3](#) hold for general parameter estimation problems and the statistical properties of DGMM hold as long as the model of interest satisfies the global and local identification assumptions and the standard regularity conditions⁷. More broadly, our results support DGMM weighting as a tractable approach to the statistical and computational tradeoffs of moment-based low-SNR mixture estimation. The algorithm for obtaining DGMM, while tailored to [Model 1.1](#), is designed and implemented with adaptability in mind and can be modified for other models. Given its theoretical properties and empirical performance, we expect that DGMM will be useful for a broad class of statistical estimation problems, especially in challenging low-SNR regimes.

Acknowledgments

We would like to thank Afonso S. Bandeira, Tamir Bendory, Charles Fefferman, Marc A. Gilles, Anya Katsevich, Joe Kileel, João M. Pereira, and Liza Rebrova for helpful discussions. In particular, we thank Marc A. Gilles for suggesting the randomly pivoted Cholesky algorithm to speed up the code and João M. Pereira for helpful suggestions on numerical optimization. We would also like to acknowledge that the anonymous reviews have greatly improved the manuscript. Claude Opus 5.5 was used to assist in developing a wrapper for repeated trials of the existing implementation of the algorithms and in writing code to plot the numerical results. The authors assume responsibility for all content.

Funding

The work was supported in part by AFOSR FA9550-20-1-0266 and FA9550-23-1-0249, the Simons Foundation Math+X Investigator Award, NSF DMS 2009753, and NIH/NIGMS R01GM136780-01.

⁷See standard references, e.g., [56] for a comprehensive discussion on this.

References

- [1] Asaf Abas, Tamir Bendory, and Nir Sharon. The generalized method of moments for multi-reference alignment. *IEEE Transactions on Signal Processing*, 70:1377–1388, 2022. [4](#), [18](#)
- [2] Joseph G Altonji and Lewis M Segal. Small-sample bias in GMM estimation of covariance structures. *Journal of Business & Economic Statistics*, 14(3):353–366, 1996. [9](#)
- [3] Animashree Anandkumar, Rong Ge, Daniel J Hsu, Sham M Kakade, Matus Telgarsky, et al. Tensor decompositions for learning latent variable models. *J. Mach. Learn. Res.*, 15(1):2773–2832, 2014. [2](#)
- [4] Joseph Anderson, Mikhail Belkin, Navin Goyal, Luis Rademacher, and James Voss. The more, the merrier: The blessing of dimensionality for learning large Gaussian mixtures. In *Proc. COLT*, pages 1135–1164, 2014. [2](#)
- [5] Sanjeev Arora and Ravi Kannan. Learning mixtures of arbitrary Gaussians. In *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, pages 247–257, 2001. [2](#)
- [6] Jimmy Ba, Murat A Erdogdu, Taiji Suzuki, Zhichao Wang, and Denny Wu. Learning in the presence of low-dimensional structure: A spiked random matrix perspective. *Adv. Neural Inf. Process. Syst.*, 36:17420–17449, 2023. [5](#)
- [7] Francis Bach. Sharp analysis of low-rank kernel matrix approximations. In *Proc. COLT*, pages 185–209, 2013. [13](#)
- [8] Ainesh Bakshi, Ilias Diakonikolas, He Jia, Daniel M Kane, Pravesh K Kothari, and Santosh S Vempala. Robustly learning mixtures of k arbitrary Gaussians. In *Proc. 54th ACM STOC*, pages 1234–1247, 2022. [2](#)
- [9] Mikhail Belkin and Kaushik Sinha. Polynomial learning of distribution families. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 103–112. IEEE, 2010. [2](#)
- [10] Eric Temple Bell. Exponential polynomials. *Annals of Mathematics*, 35(2):258–277, 1934. [7](#), [26](#), [27](#)
- [11] Aditya Bhaskara, Moses Charikar, Ankur Moitra, and Aravindan Vijayaraghavan. Smoothed analysis of tensor decompositions. In *Proc. 46th ACM STOC*, pages 594–603, 2014. [2](#)
- [12] Christopher M Bishop. Mixture density networks, 1994. [15](#)
- [13] Christopher M Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995. [15](#)
- [14] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. [15](#)
- [15] Christos Boutsidis, Petros Drineas, and Michael W Mahoney. Unsupervised feature selection for the k -means clustering problem. *Advances in Neural Information Processing Systems*, 22, 2009. [13](#)
- [16] Spencer Charles Brubaker and Santosh Vempala. Isotropic PCA and affine-invariant clustering. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 551–560. IEEE, 2008. [2](#)

- [17] T Tony Cai, X Jessie Jeng, and Jiashun Jin. Optimal detection of heterogeneous and heteroscedastic mixtures. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73(5):629–662, 2011. [4](#)
- [18] Jian Cao, Marc G Genton, David E Keyes, and George M Turkiyyah. Exploiting low-rank covariance structures for computing high-dimensional normal and student-t probabilities. *Statistics and Computing*, 31:1–16, 2021. [5](#)
- [19] Charalambos A Charalambides. *Enumerative Combinatorics*. Chapman and Hall/CRC, 2018. [7](#), [26](#)
- [20] Robert N Cochran and Frederick H Horne. Statistically weighted principal component analysis of rapid scanning wavelength kinetics experiments. *Analytical Chemistry*, 49(6):846–853, 1977. [4](#)
- [21] Pierre Comon, Gene Golub, Lek-Heng Lim, and Bernard Mourrain. Symmetric tensors and symmetric tensor rank. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1254–1279, 2008. [6](#)
- [22] Giovanni Compiani and Yuichi Kitamura. Using mixtures in econometric models: A brief review and some new results. *The Econometrics Journal*, 19(3):C95–C127, 2016. [4](#)
- [23] Louis Comtet. *Advanced Combinatorics: The Art of Finite and Infinite Expansions*. Springer, 1974. [7](#)
- [24] Harald Cramér. *Mathematical Methods of Statistics*, volume 26. Princeton University Press, 1946. [23](#)
- [25] Sanjoy Dasgupta. Learning mixtures of Gaussians. In *Proc. 40th IEEE FOCS*, pages 634–644, 1999. [2](#)
- [26] Sanjoy Dasgupta and Leonard J. Schulman. A two-round variant of EM for Gaussian mixtures. In *Proc. 16th UAI*, pages 152–159, 2000. [2](#)
- [27] Chris Ding and Xiaofeng He. K -means clustering via principal component analysis. In *Proc. 21st ICML*, page 29, 2004. [13](#)
- [28] Petros Drineas, Michael W Mahoney, and Nello Cristianini. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. *J. Mach. Learn. Res.*, 6(72):2153–2175, 2005. [12](#)
- [29] Zhou Fan, Roy R Lederman, Yi Sun, Tianhao Wang, and Sheng Xu. Maximum likelihood for high-noise group orbit estimation and single-particle cryo-EM. *The Annals of Statistics*, 52(1):52–77, 2024. [4](#), [18](#)
- [30] Zhou Fan, Yi Sun, Tianhao Wang, and Yihong Wu. Likelihood landscape and maximum likelihood estimation for the discrete orbit recovery model. *Comm. Pure Appl. Math.*, 76(6):1208–1302, 2023. [4](#), [18](#)
- [31] Chris Fraley and Adrian E Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, 2002. [5](#)
- [32] Rong Ge, Qingqing Huang, and Sham M Kakade. Learning mixtures of Gaussians in high dimensions. In *Proc. 47th ACM STOC*, pages 761–770, 2015. [2](#)

- [33] Sixue Gong, Vishnu Naresh Boddeti, and Anil K Jain. On the intrinsic dimensionality of image representations. In *Proc. IEEE/CVF CVPR*, pages 3987–3996, 2019. [5](#)
- [34] Bingni Guo, Jiawang Nie, and Zi Yang. Learning diagonal Gaussian mixture models and incomplete tensor decompositions. *Vietnam Journal of Mathematics*, pages 1–26, 2022. [2](#)
- [35] Alastair R Hall. Generalized method of moments. *Handbook of Research Methods and Applications in Empirical Macroeconomics*, pages 313–333, 2013. [2](#)
- [36] Lars Peter Hansen. Large sample properties of generalized method of moments estimators. *Econometrica*, 50(4):1029–1054, 1982. [2](#), [7](#), [8](#), [11](#), [25](#), [26](#)
- [37] Lars Peter Hansen, John Heaton, and Amir Yaron. Finite-sample properties of some alternative GMM estimators. *Journal of Business & Economic Statistics*, 14(3):262–280, 1996. [9](#)
- [38] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. [15](#)
- [39] David Hong and Laura Balzano. Optimal sample acquisition for optimally weighted PCA from heterogeneous quality sources. *IEEE Signal Processing Letters*, 2025. [4](#)
- [40] David Hong, Fan Yang, Jeffrey A Fessler, and Laura Balzano. Optimally weighted PCA for high-dimensional heteroscedastic data. *SIAM Journal on Mathematics of Data Science*, 5(1):222–250, 2023. [4](#)
- [41] Daniel Hsu and Sham M Kakade. Learning mixtures of spherical Gaussians: Moment methods and spectral decompositions. In *Proc. 4th ITCS*, pages 11–20, 2013. [2](#)
- [42] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. [15](#)
- [43] Adam Tauman Kalai, Ankur Moitra, and Gregory Valiant. Efficiently learning mixtures of two Gaussians. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, pages 553–562, 2010. [2](#)
- [44] Anya Katsevich and Afonso S Bandeira. Likelihood maximization and moment matching in low SNR Gaussian mixture models. *Communications on Pure and Applied Mathematics*, 76(4):788–842, 2023. [4](#), [9](#), [18](#)
- [45] C. G. Khatri. Some results for the singular normal multivariate regression models. *Sankhyā: The Indian Journal of Statistics, Series A*, 30(3):267–280, 1968. [24](#)
- [46] Oliver Knill. Cauchy–Binet for pseudo-determinants. *Linear Algebra and Its Applications*, 459:522–547, 2014. [24](#)
- [47] Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009. [6](#)
- [48] Sandeep Kumar, Jiaxi Ying, José Vinícius de M Cardoso, and Daniel P Palomar. A unified framework for structured graph learning via spectral constraints. *Journal of Machine Learning Research*, 21(22):1–60, 2020. [5](#)
- [49] Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004. [9](#)

- [50] Allen Liu and Ankur Moitra. Robustly learning general mixtures of Gaussians. *J. ACM*, 70(2):1–53, 2023. [2](#)
- [51] Dong C Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1):503–528, 1989. [15](#)
- [52] Zhongyuan Lyu and Dong Xia. Optimal estimation and computational limit of low-rank Gaussian mixtures. *The Annals of Statistics*, 51(2):646–667, 2023. [3](#)
- [53] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016. [15](#)
- [54] Geoffrey McLachlan and David Peel. *Finite Mixture Models*. Wiley, 2004. [1](#)
- [55] Ankur Moitra and Gregory Valiant. Settling the polynomial learnability of mixtures of Gaussians. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 93–102. IEEE, 2010. [2](#)
- [56] Whitney K Newey and Daniel McFadden. Large sample estimation and hypothesis testing. In *Handbook of Econometrics*, volume 4, pages 2111–2245. Elsevier, 1994. [2](#), [7](#), [11](#), [18](#), [24](#), [25](#)
- [57] Whitney K. Newey and Richard J. Smith. Higher order properties of GMM and generalized empirical likelihood estimators. *Econometrica*, 72(1):219–255, 2004. [9](#)
- [58] Dino Oglic and Thomas Gärtner. Nyström method with kernel k -means++ samples as landmarks. In *International Conference on Machine Learning*, pages 2652–2660. PMLR, 2017. [12](#), [13](#), [18](#), [28](#), [29](#)
- [59] Karl Pearson. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A*, 185:71–110, 1894. [2](#)
- [60] Giovanni Peccati and Murad S Taqqu. *Wiener Chaos: Moments, Cumulants and Diagrams*. Springer, 2011. [7](#)
- [61] Jose M Pereira, Joe Kileel, and Tamara Kolda. Tensor moments of Gaussian mixture models: Theory and applications. *arXiv preprint arXiv:2202.06930*, 2022. [2](#), [3](#), [7](#), [12](#), [15](#), [16](#), [18](#), [26](#)
- [62] Haim Permuter, Joseph Francos, and Ian Jermyn. Gaussian mixture models of texture and colour for image database retrieval. In *Proc. IEEE ICASSP*, volume 3, pages III–569, 2003. [1](#)
- [63] Phillip Pope, Chen Zhu, Ahmed Abdelkader, Micah Goldblum, and Tom Goldstein. The intrinsic dimension of images and its impact on learning. *arXiv preprint arXiv:2104.08894*, 2021. [5](#)
- [64] C. R. Rao. Multivariate analysis. In *Linear Statistical Inference and its Applications*, chapter 8, pages 516–604. Wiley, 2nd edition, 1973. [23](#)
- [65] Douglas A Reynolds and Richard C Rose. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3(1):72–83, 1995. [2](#)
- [66] David Roodman. A note on the theme of too many instruments. *Oxford Bulletin of Economics and Statistics*, 71(1):135–158, 2009. [9](#)

- [67] M.S. Srivastava and C.G. Khatri. *An Introduction to Multivariate Statistics*. North-Holland, 1979. [23](#)
- [68] Jan Pawel Stanczuk, Georgios Batzolis, Teo Deveney, and Carola-Bibiane Schönlieb. Diffusion models encode the intrinsic dimension of data manifolds. In *Proc. 41st ICML*, 2024. [5](#)
- [69] O Tamuz, T Mazeh, and S Zucker. Correcting systematic effects in a large set of photometric light curves. *Monthly Notices of the Royal Astronomical Society*, 356(4):1466–1470, 2005. [4](#)
- [70] D.M. Titterton, A.F.M. Smith, and U.E. Makov. *Statistical Analysis of Finite Mixture Distributions*. Applied Section. Wiley, 1985. [1](#)
- [71] Santosh Vempala and Grant Wang. A spectral algorithm for learning mixture models. *Journal of Computer and System Sciences*, 68(4):841–860, 2004. [2](#)
- [72] Christopher Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. *Adv. Neural Inf. Process. Syst.*, 13, 2000. [12](#)
- [73] Yihong Wu and Pengkun Yang. Optimal estimation of Gaussian mixtures via denoised method of moments. *Ann. Statist.*, 48(4):1981–2007, 2020. [9](#)
- [74] Huangjie Zhang, Jiaming Zhou, Yufei Lu, Mingyuan Guo, Peiyi Wang, Li Shen, and Qiang Qu. The emergence of reproducibility and generalizability in diffusion models. *arXiv preprint arXiv:2310.05264*, 2024. [5](#)
- [75] Rui Zhou, Jiaxi Ying, and Daniel P Palomar. Covariance matrix estimation under low-rank factor model with nonnegative correlations. *IEEE Transactions on Signal Processing*, 70:4020–4030, 2022. [5](#)

A Supporting proofs for model specifications

Suppose the covariance of the j -th Gaussian component in [Model 1.1](#), Σ_j , has the compact singular value decomposition:

$$\Sigma_j = U_j \Lambda_j U_j^\top, \quad U_j \in \mathbb{R}^{d \times R_j}, \quad \Lambda_j = \text{diag} \left(\lambda_j^{(1)}, \dots, \lambda_j^{(R_j)} \right) \in \mathbb{R}^{R_j \times R_j}. \quad (\text{A.1})$$

Due to the rank deficiency of Σ_j , the p.d.f. of $\mathbf{X}_j$ does not exist w.r.t. the d -dimensional Lebesgue measure ν_d , as shown in the following proposition:

Proposition A.1. *The p.d.f. of $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma) \in \mathbb{R}^d$ where $\text{rank } \Sigma = R < d$ does not exist w.r.t. the d -dimensional Lebesgue measure ν_d .*

Proof. For $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ with singular Σ , $\nu_d(\boldsymbol{\mu} + \text{range}(\Sigma)) = 0$. For any measurable subset A of the affine space $\boldsymbol{\mu} + \text{range}(\Sigma)$, we have $\nu_d(A) \leq \nu_d(\boldsymbol{\mu} + \text{range}(\Sigma))$. Thus, $\nu_d(A) = 0$. As shown by [24, (21.12.5) on p. 290], [64, (8a.4.11) and (8a.4.12) on p. 527-528], and [67, p.43], if Σ is singular, then $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ lies on a rank Σ -dimensional subspace of $\mathbb{R}^d$, almost surely. This implies that there exists some measurable subset $A \subseteq \boldsymbol{\mu} + \text{range}(\Sigma)$ such that $\mathbb{P}[\mathbf{X} \in A] \neq 0$. By the Radon-Nikodým theorem, the p.d.f. of $\mathbf{X}$ does not exist w.r.t. ν_d . $\square$

However, the p.d.f. of $\mathbf{X}_j$ does exist w.r.t. the following R_j -dimensional measure supported on the affine space $\boldsymbol{\mu}_j + \text{range}(\Sigma_j)$, as shown in the following proposition:

Proposition A.2 ([45]). Let $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ with $\text{rank } \Sigma = R$. Suppose Σ has the compact SVD: $\Sigma = U_R \Lambda_R U_R^\top$, with $U_R \in \mathbb{R}^{d \times R}$ and $\Lambda_R = \text{diag}(\lambda^{(1)}, \dots, \lambda^{(R)}) \in \mathbb{R}^{R \times R}$. Let $N \in \mathbb{R}^{d \times (d-R)}$ be a matrix of rank $(d-R)$ such that $N^\top \Sigma = 0$. Define the affine map $\phi : \mathbb{R}^R \rightarrow \mathbb{R}^d$, $\phi(\cdot) := \boldsymbol{\mu} + U_R(\cdot)$. Let $\nu_R : \mathcal{B}(\mathbb{R}^R) \rightarrow \mathbb{R}_{\geq 0}$ denote the R -dimensional Lebesgue measure, where $\mathcal{B}(\mathbb{R}^d)$ denotes the standard Borel algebra of $\mathbb{R}^d$. Then the p.d.f. of $\mathbf{X}$ exists w.r.t. the following R -dimensional measure supported on the affine space $\boldsymbol{\mu} + \text{range}(\Sigma)$:

$$\rho : \mathcal{B}(\mathbb{R}^d) \rightarrow \mathbb{R}_{\geq 0}, \quad \rho := \nu_R \circ \phi^{-1}, \quad (\text{A.2})$$

and the p.d.f. is given by $p_{\mathbf{X}}(\mathbf{x}) = \frac{\exp\{-\frac{1}{2}(x-\boldsymbol{\mu})^\top \Sigma^\dagger (x-\boldsymbol{\mu})\}}{(2\pi)^{\frac{R}{2}} (\text{Det}(\Sigma))^{\frac{1}{2}}}$ for $x \in \boldsymbol{\mu} + \text{range}(\Sigma)$, where $\Sigma_j^\dagger$ is Moore-Penrose generalized inverse, $\text{Det}(\Sigma_j) = \prod_{r=1}^{R_j} \lambda_j^{(r)}$ is the pseudo-determinant (see [46]).

B Supporting proofs for computational complexity and statistical properties

Proof of Theorem 4.1. Define the population objective function:

$$Q(\boldsymbol{\theta}) := \mathbb{E}[g(\boldsymbol{\theta}; \mathbf{y}_n)]^\top W \mathbb{E}[g(\boldsymbol{\theta}; \mathbf{y}_n)], \quad W = \text{diag}(\underbrace{w_1, \dots, w_1}_{d \text{ copies}}, \dots, \underbrace{w_L, \dots, w_L}_{d^L \text{ copies}}). \quad (\text{B.1})$$

We check the standard regularity conditions for consistency, in e.g. [56, Theorem 2.6]:

- (a) The sample data is i.i.d., assumed in the specifications of Model 1.1.
- (b) The parameter space Θ is compact and contains $\boldsymbol{\theta}^*$ as an interior point, which is a standard assumption.
- (c) The moment function $g(\boldsymbol{\theta}; \mathbf{y}_n)$ in (2.11) depends polynomially on the parameter $\boldsymbol{\theta}$ and is therefore continuous at each $\boldsymbol{\theta} \in \Theta$.
- (d) Since Θ is compact and all moments of any finite Gaussian mixture are finite, we have $\mathbb{E}[\sup_{\boldsymbol{\theta} \in \Theta} \|g(\boldsymbol{\theta}; \mathbf{y}_n)\|] < \infty$.
- (e) The model is globally identifiable, which is a standard assumption (see Section 2.3).
- (f) The DGMM weighting matrices in (3.5) satisfy $\widehat{W}^{[t]} \xrightarrow{p} W$. W is positive-definite due to the following: The numerator in (3.5) $\text{Tr}[S^{(k,k)}] = \mathbb{E}[\|g^{(k)}(\boldsymbol{\theta}^*; \mathbf{y}_n)\|^2] > 0$. The denominator in (3.5) is positive: every term under the double sum is a square, with at least one strictly positive entry.

Therefore, we get that $\widehat{\boldsymbol{\theta}}^{[t]} \xrightarrow{p} \boldsymbol{\theta}^*$, which proves the consistency of $\widehat{\boldsymbol{\theta}}^{(\text{DGMM})}$.

To show that $\widehat{\boldsymbol{\theta}}^{(\text{DGMM})}$ is asymptotically normal, in addition to the above regularity conditions for consistency, we need to check the following conditions:

- (g) The moment function $g(\boldsymbol{\theta}; \mathbf{y}_n)$ in (2.11) is continuously differentiable in a neighborhood of $\boldsymbol{\theta}^*$ with probability approaching one as $N \rightarrow \infty$.
- (h) $\mathbb{E}[\|g(\boldsymbol{\theta}^*; \mathbf{y}_n)\|^2] < \infty$ and $\mathbb{E}[\sup_{\boldsymbol{\theta} \in \mathcal{N}(\boldsymbol{\theta}^*)} \|\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}; \mathbf{y}_n)\|] < \infty$, which follows from the definitions of Model 1.1 and (2.11).

- (i) $G(\boldsymbol{\theta}^*)^\top W G(\boldsymbol{\theta}^*)$ is non-singular, which follows from the fact that W is positive-definite and the standard assumption discussed in [Section 2.3](#) that the model is locally identifiable, that is, $\text{rank}(G(\boldsymbol{\theta}^*)) = p$.

By the standard asymptotic normality result, in e.g. [\[56, Theorem 3.4\]](#), if conditions (a)-(i) are satisfied, then for any GMM estimator $\hat{\boldsymbol{\theta}}$,

$$\sqrt{N} \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^* \right) \xrightarrow{d} \mathcal{N} \left(\mathbf{0}, \left(G^\top W G \right)^{-1} G^\top W S W G \left(G^\top W G \right)^{-1} \right), \quad (\text{B.2})$$

where we write $G = G(\boldsymbol{\theta}^*)$ for notational convenience. For the DGMM estimator, since the weighting matrix W is block-diagonal, we can express the asymptotic variance S in block form to highlight how each moment group contributes to the overall asymptotic variance. To do this, we partition the Jacobian matrix G into L blocks, $G = \left[G^{(1)\top}, \dots, G^{(L)\top} \right]^\top$, so that $G^{(k)}$ is the d^k -by- p submatrix of G corresponding to the k -th moment. We also partition S into blocks such that $S^{(k,k')}$ is the d^k -by- $d^{k'}$ submatrix of S , consisting of the cross-order covariance between the k -th and k' -th sample moment conditions, for $k, k' = 1, \dots, L$, i.e.,

$$S^{(k,k')} := (S_{ij})_{i \in \mathcal{I}_k, j \in \mathcal{I}_{k'}} = \mathbb{E} \left[g^{(k)}(\boldsymbol{\theta}^*; \mathbf{y}_n) g^{(k')}(\boldsymbol{\theta}^*; \mathbf{y}_n)^\top \right]. \quad (\text{B.3})$$

Then we get the form of the asymptotic covariance matrix of $\hat{\boldsymbol{\theta}}^{(\text{DGMM})}$ in [\(4.1\)](#). $\square$

Proof of Theorem 4.2. As shown in [\[36, Theorem 3.2\]](#), in the class of GMM estimators, $V^{(\text{GMM})} = \left(G^\top S^{-1} G \right)^{-1}$ is the minimum asymptotic covariance within the class of GMM estimators, which directly implies that $V^{(\text{DGMM})} - V^{(\text{GMM})} \succeq 0$. We now obtain the upper bound on $V^{(\text{DGMM})}$. Let W be the DGMM weighting matrix in [\(3.3\)](#). We first apply a change of coordinate to get $\tilde{S} = W^{\frac{1}{2}} S W^{\frac{1}{2}}$. Let $E = \tilde{S} - I_q$. Then

$$E^{(k,k)} = w_k S^{(k,k)} - I_{d^k}, \quad E^{(k,k')} = \sqrt{w_k w_{k'}} S^{(k,k')} (k \neq k'). \quad (\text{B.4})$$

Since E is a symmetric matrix, $E^{(k,k')} = E^{(k',k)\top}$ and so $\|E^{(k,k')}\|_{\text{op}} = \|E^{(k',k)}\|_{\text{op}}$. For an arbitrary vector $\mathbf{x} = [\mathbf{x}_1; \dots; \mathbf{x}_L] \in \mathbb{R}^q$ where $\mathbf{x}_k \in \mathbb{R}^{d^k}$, by applying Cauchy-Schwarz, AM-GM inequality, and the symmetry of E , we get

$$\begin{aligned} |\mathbf{x}^\top E \mathbf{x}| &\leq \sum_{k,k'} \|E^{(k,k')}\|_{\text{op}} \|\mathbf{x}_k\| \|\mathbf{x}_{k'}\| \leq \frac{1}{2} \sum_{k,k'} \|E^{(k,k')}\|_{\text{op}} \left(\|\mathbf{x}_k\|^2 + \|\mathbf{x}_{k'}\|^2 \right) \\ &= \sum_k \left(\sum_{k'} \|E^{(k,k')}\|_{\text{op}} \right) \|\mathbf{x}_k\|^2 \leq \eta \|\mathbf{x}\|^2, \end{aligned}$$

where the last step is due to [\(4.2\)](#). So we get $(1 - \eta)I_q \preceq \tilde{S} \preceq (1 + \eta)I_q$, which implies that for any eigenvalue λ of $\tilde{S}$, $\lambda \in [1 - \eta, 1 + \eta]$ and is strictly positive. Therefore,

$$(\lambda - (1 - \eta))((1 + \eta) - \lambda) \geq 0 \implies \lambda + \frac{1 - \eta^2}{\lambda} \leq 2 \implies \tilde{S} + (1 - \eta^2)\tilde{S}^{-1} \preceq 2I_q. \quad (\text{B.5})$$

Define $\tilde{G} = W^{\frac{1}{2}} G$. Substituting $G = W^{-\frac{1}{2}} \tilde{G}$ and $S^{-1} = W^{\frac{1}{2}} \tilde{S}^{-1} W^{\frac{1}{2}}$ into the asymptotic covariances $V^{(\text{DGMM})} = \left(G^\top W G \right)^{-1} G^\top W S W G \left(G^\top W G \right)^{-1}$, $V^{(\text{GMM})} = \left(G^\top S^{-1} G \right)^{-1}$:

$$V^{(\text{DGMM})} = \left(\tilde{G}^\top \tilde{G} \right)^{-1} \tilde{G}^\top \tilde{S} \tilde{G} \left(\tilde{G}^\top \tilde{G} \right)^{-1}, \quad V^{(\text{GMM})} = \left(\tilde{G}^\top \tilde{S}^{-1} \tilde{G} \right)^{-1}. \quad (\text{B.6})$$

By combining (B.5) and (B.6) and using the shorthand $H := (\tilde{G}^\top \tilde{G})^{-1}$, we get

$$\begin{aligned} H\tilde{G}^\top (\tilde{S} + (1 - \eta^2)\tilde{S}^{-1})\tilde{G}H &\preceq H\tilde{G}^\top (2I_q)\tilde{G}H \\ V^{(\text{DGMM})} + (1 - \eta^2)H(V^{(\text{GMM})})^{-1}H &\preceq 2H \\ V^{(\text{DGMM})} &\preceq 2H - (1 - \eta^2)H(V^{(\text{GMM})})^{-1}H. \end{aligned} \quad (\text{B.7})$$

By a completion-of-the-square argument inspired by the proof of [36, Theorem 3.2], we get

$$\begin{aligned} &\frac{1}{1 - \eta^2}V^{(\text{GMM})} - \left(2H - (1 - \eta^2)H(V^{(\text{GMM})})^{-1}H\right) \\ &= (1 - \eta^2) \left(H - \frac{1}{1 - \eta^2}V^{(\text{GMM})}\right) (V^{(\text{GMM})})^{-1} \left(H - \frac{1}{1 - \eta^2}V^{(\text{GMM})}\right) \succeq 0, \end{aligned}$$

where the last step is due to $(V^{(\text{GMM})})^{-1} \succ 0, 1 - \eta^2 > 0$. It then follows that

$$2H - (1 - \eta^2)H(V^{(\text{GMM})})^{-1}H \preceq \frac{1}{1 - \eta^2}V^{(\text{GMM})}. \quad (\text{B.8})$$

Combining (B.7) and (B.8), we get $V^{(\text{DGMM})} \preceq \frac{1}{1 - \eta^2}V^{(\text{GMM})} = \left(1 + \frac{\eta^2}{1 - \eta^2}\right)V^{(\text{GMM})}$. $\square$

Proof of Theorem 4.4. We compute $\phi^{(k)}(\boldsymbol{\theta})$ by applying Proposition 2.9:

$$\phi^{(k)}(\boldsymbol{\theta}) = \sum_{i=1}^K \sum_{j=1}^K \pi_i \pi_j \langle \mathcal{M}^{(k)}(\boldsymbol{\theta}_i), \mathcal{M}^{(k)}(\boldsymbol{\theta}_j) \rangle = \sum_{i=1}^K \sum_{j=1}^K \pi_i \pi_j B_k(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k)}), \quad (\text{B.9})$$

where $\mathcal{M}^{(k)}(\boldsymbol{\theta}_j)$ denotes the k -th moment of the j -th Gaussian component, $\boldsymbol{\theta}_j = [\pi_j; \boldsymbol{\mu}_j; V_j]^\top$ are the parameters of the j -th Gaussian component, and $\kappa_{ij}^{(l)}$ denotes the l -th cumulant of the inner product between the i -th Gaussian component $\mathbf{X}_i$ and an independent draw of the j -th Gaussian component $\mathbf{X}'_j$, i.e., $\kappa_{ij}^{(l)} := \kappa^{(l)}(\langle \mathbf{X}_i, \mathbf{X}'_j \rangle)$, $l = 1, \dots, k$. The Bell polynomials are evaluated by applying the recurrence relation (see, e.g., [19, Theorem 11.2] or [10, Eq. 7.12]):

$$\begin{cases} B_0 = 1, & (\text{base case}) \\ B_k(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k)}) = \sum_{l=0}^{k-1} \binom{k-1}{l} B_{k-l-1}(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k-l-1)}) \kappa_{ij}^{(l+1)}. & (\text{induction step}) \end{cases} \quad (\text{B.10})$$

By applying [61, Proposition 3.3],

$$\kappa_{ij}^{(l)} = \begin{cases} \langle \boldsymbol{\mu}_j, \boldsymbol{\mu}_i \rangle, & (l = 1) \\ (l-1)! \text{Tr}((V_i V_i^\top V_j V_j^\top)^{\frac{l}{2}}) + \frac{l}{2} \boldsymbol{\mu}_i^\top V_j V_j^\top (V_i V_i^\top V_j V_j^\top)^{\frac{l-2}{2}} \boldsymbol{\mu}_i \\ \quad + \frac{l}{2} \boldsymbol{\mu}_j^\top (V_i V_i^\top V_j V_j^\top)^{\frac{l-2}{2}} V_i V_i^\top \boldsymbol{\mu}_j, & (l \text{ is even}) \\ l! \boldsymbol{\mu}_j^\top (V_i V_i^\top V_j V_j^\top)^{\frac{l-1}{2}} \boldsymbol{\mu}_i. & (l \text{ is odd}) \end{cases} \quad (\text{B.11})$$

Note that the computational complexity of the matrix multiplication $V_j^\top V_j$ is $\mathcal{O}(dR_j^2)$, better than that of $V_j V_j^\top$, which is $\mathcal{O}(d^2 R_j)$. By carefully choosing the order of the consecutive matrix

multiplication, we compute $\kappa_{ij}^{(l)}$ in (B.11) using $\mathcal{O}(ldR_{\max}^2)$ operations⁸. $\phi^{(k)}(\boldsymbol{\theta})$ in (B.9) requires (1) pre-computing $\kappa_{ij}^{(l)}$ for all $i, j \in [K]$ and $l \in [k]$ in $\mathcal{O}(kK^2dR_{\max}^2)$ operations, (2) using the recurrence relation in (B.10) to compute $B_k(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k)})$ for all $i, j \in [K]$ in $\mathcal{O}(k^2K^2)$ operations. In total, $\phi^{(k)}(\boldsymbol{\theta})$ takes $\mathcal{O}(k^2K^2 + kK^2dR_{\max}^2)$ operations.

The derivative of the Bell polynomials is given by [10, Eq. 5.1]:

$$\frac{\partial B_k}{\partial x_i}(x_1, \dots, x_k) = \binom{k}{i} B_{k-i}(x_1, \dots, x_{k-i}). \quad (\text{B.12})$$

Using (B.9), the above identity, and the chain rule, we compute:

$$\nabla_{\pi_j} \phi^{(k)}(\boldsymbol{\theta}) = 2 \sum_{i=1}^K \pi_i B_k(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k)}), \quad (\text{B.13})$$

$$\nabla_{\boldsymbol{\mu}_j} \phi^{(k)}(\boldsymbol{\theta}) = 2 \sum_{i=1}^K \pi_i \pi_j \sum_{l=1}^k \binom{k}{l} B_{k-l}(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k-l)}) \nabla_{\boldsymbol{\mu}_j} \kappa_{ij}^{(l)}, \quad (\text{B.14})$$

$$\nabla_{V_j} \phi^{(k)}(\boldsymbol{\theta}) = 2 \sum_{i=1}^K \pi_i \pi_j \sum_{l=1}^k \binom{k}{l} B_{k-l}(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k-l)}) \nabla_{V_j} \kappa_{ij}^{(l)}. \quad (\text{B.15})$$

From the cumulants in (B.11), we get the gradients of the cumulants w.r.t. $\boldsymbol{\mu}_j$ and V_j :

$$\nabla_{\boldsymbol{\mu}_j} \kappa_{ij}^{(l)} = \begin{cases} \boldsymbol{\mu}_i, & (l = 1) \\ l! \left(V_i V_i^\top V_j V_j^\top \right)^{\frac{l-2}{2}} V_i V_i^\top \boldsymbol{\mu}_j, & (l \text{ is even}) \\ l! \left(V_i V_i^\top V_j V_j^\top \right)^{\frac{l-1}{2}} \boldsymbol{\mu}_i, & (l \text{ is odd}) \end{cases} \quad (\text{B.16})$$

$$\nabla_{V_j} \kappa_{ij}^{(l)} = \begin{cases} 0, & (l = 1) \\ l! \left(V_i V_i^\top V_j V_j^\top \right)^{\frac{l-2}{2}} V_i V_i^\top V_j \\ \quad + l! \sum_{p=0}^{\frac{l-2}{2}} \left(V_i V_i^\top V_j V_j^\top \right)^p \boldsymbol{\mu}_i \boldsymbol{\mu}_i^\top \left(V_j V_j^\top V_i V_i^\top \right)^{\frac{l-2}{2}-p} V_j \\ \quad + l! \sum_{p=0}^{\frac{l-2}{2}-1} V_i V_i^\top \left(V_j V_j^\top V_i V_i^\top \right)^p \boldsymbol{\mu}_j \boldsymbol{\mu}_j^\top \left(V_i V_i^\top V_j V_j^\top \right)^{\frac{l-2}{2}-1-p} V_i V_i^\top V_j, & (l \text{ is even}) \\ l! \sum_{p=0}^{\frac{l-1}{2}-1} V_i V_i^\top \left(V_j V_j^\top V_i V_i^\top \right)^p \boldsymbol{\mu}_j \boldsymbol{\mu}_i^\top \left(V_j V_j^\top V_i V_i^\top \right)^{\frac{l-1}{2}-1-p} V_j \\ \quad + l! \sum_{p=0}^{\frac{l-1}{2}-1} \left(V_i V_i^\top V_j V_j^\top \right)^p \boldsymbol{\mu}_i \boldsymbol{\mu}_j^\top \left(V_i V_i^\top V_j V_j^\top \right)^{\frac{l-1}{2}-1-p} V_i V_i^\top V_j. & (l \text{ is odd}) \end{cases} \quad (\text{B.17})$$

$\nabla_{\pi_j} \phi^{(k)}(\boldsymbol{\theta})$ requires (1) pre-computing $\kappa_{ij}^{(l)}$ for all $i \in [K]$, $l \in [k]$ in $\mathcal{O}(kKdR_{\max}^2)$ operations, (2) using the recurrence relation in (B.10) to compute $B_k(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k)})$ for all $i \in [K]$ in $\mathcal{O}(k^2K)$ operations. In total, $\nabla_{\pi_j} \phi^{(k)}(\boldsymbol{\theta})$ takes $\mathcal{O}(k^2K + kKdR_{\max}^2)$ operations. $\nabla_{\boldsymbol{\mu}_j} \phi^{(k)}(\boldsymbol{\theta})$ requires (1) pre-computing $\kappa_{ij}^{(l)}$ for all $i \in [K]$, $l \in [k]$ in $\mathcal{O}(kKdR_{\max}^2)$ operations, (2) pre-computing $\nabla_{\boldsymbol{\mu}_j} \kappa_{ij}^{(l)}$ for all $i \in [K]$, $l \in [k]$ in $\mathcal{O}(kKdR_{\max})$ operations, (3) using the recurrence relation in (B.10) to compute

⁸Our implementation also follows this faster order.

$B_{k-l}(\kappa_{ij}^{(1)}, \dots, \kappa_{ij}^{(k-l)})$, $l = 1, \dots, k$ for all $i \in [K]$ in $\mathcal{O}(k^2 K)$ operations. In total, $\nabla_{\mu_j} \phi^{(k)}(\theta)$ takes $\mathcal{O}(k^2 K + k K d R_{\max}^2)$ operations. With a similar calculation, $\nabla_{V_j} \phi^{(k)}(\theta)$ takes $\mathcal{O}(k^2 K + k K d R_{\max}^2)$ operations. $\square$

Proof of Theorem 4.5. We compute $\psi^{(k)}(\theta; \mathbf{y}_n)$ by applying Proposition 2.9:

$$\psi^{(k)}(\theta; \mathbf{y}_n) = \sum_{j=1}^K \pi_j \langle \mathcal{M}^{(k)}(\theta_j), \mathbf{y}_n^{\otimes k} \rangle = \sum_{j=1}^K \pi_j B_k(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0). \quad (\text{B.18})$$

The Bell polynomials are evaluated by applying the recurrence relation:

$$\begin{cases} B_0 = 1, & (\text{base case}) \\ B_1(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0) = \mathbf{y}_n^\top \boldsymbol{\mu}_j, & (\text{base case}) \\ B_k(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0) = \\ \quad B_{k-1}(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0) \mathbf{y}_n^\top \boldsymbol{\mu}_j \\ \quad + (k-1) B_{k-2}(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0) \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n. & (\text{induction step}) \end{cases} \quad (\text{B.19})$$

Using (B.18), the chain rule, and the derivative of the Bell polynomials, we compute $\nabla_{\pi_j} \psi^{(k)}(\theta; \mathbf{y}_n)$, $\nabla_{\mu_j} \psi^{(k)}(\theta; \mathbf{y}_n)$, $\nabla_{V_j} \psi^{(k)}(\theta; \mathbf{y}_n)$ for each j -th component:

$$\nabla_{\pi_j} \psi^{(k)}(\theta; \mathbf{y}_n) = B_k(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0), \quad (\text{B.20})$$

$$\nabla_{\mu_j} \psi^{(k)}(\theta; \mathbf{y}_n) = k \pi_j B_{k-1}(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0) \mathbf{y}_n, \quad (\text{B.21})$$

$$\nabla_{V_j} \psi^{(k)}(\theta; \mathbf{y}_n) = \begin{cases} 0, & (k = 1) \\ 2 \binom{k}{2} \pi_j B_{k-2}(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0) \mathbf{y}_n \mathbf{y}_n^\top V_j, & (k \geq 2). \end{cases} \quad (\text{B.22})$$

For all $j \in [K]$, $\mathbf{y}_n^\top \boldsymbol{\mu}_j$, and $\mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n$; $\mathbf{y}_n \mathbf{y}_n^\top V_j$ can be pre-computed in $\mathcal{O}(Kd)$, $\mathcal{O}(Kd R_{\max})$, and $\mathcal{O}(Kd R_{\max})$, respectively, by carefully choosing the order of matrix multiplication. For all $j \in [K]$, using the recurrence relation in (B.19), we can compute $B_k(\mathbf{y}_n^\top \boldsymbol{\mu}_j, \mathbf{y}_n^\top V_j V_j^\top \mathbf{y}_n, 0, \dots, 0)$ in $\mathcal{O}(kK)$ operations. In total, $\psi^{(k)}(\theta; \mathbf{y}_n)$, $\nabla_{\pi_j} \psi^{(k)}(\theta; \mathbf{y}_n)$, $\nabla_{\mu_j} \psi^{(k)}(\theta; \mathbf{y}_n)$, and $\nabla_{V_j} \psi^{(k)}(\theta; \mathbf{y}_n)$ each requires $\mathcal{O}(kK + Kd R_{\max})$ operations. $\square$

Proof of Theorem 4.6. The sum $\sum_{n'} C_{k,n,n'}$ can be approximated in the following steps:

- (1) Select m landmarks from N points using k -means++ sampling proposed in [58], which takes $\mathcal{O}(Ndm)$ operations (d -dimensional distances for N points for m centroids).
- (2) Compute $C^{(k)}$, an N -by- m matrix containing kernel evaluations between all N data points and the m landmarks, which takes $\mathcal{O}(Ndm)$ operations.
- (3) Compute $W^{(k)}$, an m -by- m Gram matrix on the m landmarks for the k -th order moment, which takes $\mathcal{O}(m^2 d)$ operations.
- (4) Apply pivoted Cholesky decomposition to factorize $W^{(k)} \approx L^{(k)} (L^{(k)})^\top$, where $L^{(k)}$ is the Cholesky factor of rank $R_{\text{Chol}} < m$. This takes $\mathcal{O}(m R_{\text{Chol}}^2)$ operations.

- (5) To compute the inverse $(W^{(k)})^{-1} \mathbf{v}$ (which is required for the Nyström approximation), we perform a forward and back substitution through the triangular factor: $L^{(k)} \mathbf{z} = \mathbf{v}$ (forward substitution) and then $(L^{(k)})^\top \mathbf{y} = \mathbf{z}$ (back substitution). This step takes $\mathcal{O}(m^2)$ operations.
- (6) After obtaining the solution vector $\mathbf{y}$ from the triangular solve step, we compute the approximate $\sum_{n'} C_{k,n,n'} \approx C^{(k)} \mathbf{y}$, which takes $\mathcal{O}(Nm)$ operations.

Thus, the overall computational complexity is $\mathcal{O}(Ndm + m^2d + mR_{\text{Chol}}^2 + m^2 + Nm)$. Taking the dominant terms, we get $\mathcal{O}(Ndm)$.

□

Proof of Lemma 4.7. Applying [58, Theorem 5] to our problem, we get

$$\mathbb{E} \left[\frac{\|H^{(k)} - \widetilde{H}_m^{(k)}\|_F}{\|H^{(k)} - H_m^{(k)}\|_F} \right] \leq 8(\ln(m+1) + 2) \left(\sqrt{N-m} + \frac{\phi(C^\star | U_m)}{\|H^{(k)} - H_m^{(k)}\|_F} \right). \quad (\text{B.23})$$

Under polynomial spectral decay of $H^{(k)}$, $\lambda_i^{(k)} \in \mathcal{O}(i^{-a})$ with $a > 1$, i.e., there exists some constant $C > 0$ such that for large i , $\lambda_i^{(k)} \leq Ci^{-a}$ for $a > 1$. By the integral test, we have

$$\begin{aligned} \|H^{(k)} - H_m^{(k)}\|_F^2 &= \sum_{i=m+1}^N (\lambda_i^{(k)})^2 \leq \sum_{i=m+1}^N (Ci^{-a})^2 \leq C^2 \int_m^N x^{-2a} dx \\ &= \frac{C^2}{2a-1} (m^{-2a+1} - N^{-2a+1}) \leq \frac{C^2}{2a-1} (m^{-2a+1}). \end{aligned}$$

After taking the square root on both sides, we get $\|H^{(k)} - H_m^{(k)}\|_F \leq \frac{C}{\sqrt{2a-1}} (m^{-a+1/2})$. Combining this, (B.23), and the cluster potential condition in (4.7), we get

$$\begin{aligned} \mathbb{E} \left[\left\| H^{(k)} - \widetilde{H}_m^{(k)} \right\|_F \right] &\leq 8(\ln(m+1) + 2) \left(\sqrt{N-m} \|H^{(k)} - H_m^{(k)}\|_F + \phi(C^\star | U_m) \right) \\ &\leq 8(\ln(m+1) + 2) \left(2\sqrt{N-m} \|H^{(k)} - H_m^{(k)}\|_F \right) \\ &\leq 16(\ln(m+1) + 2) \sqrt{N-m} \frac{C}{\sqrt{2a-1}} (m^{-a+1/2}), \end{aligned}$$

which implies that $\mathbb{E} \left[\left\| H^{(k)} - \widetilde{H}_m^{(k)} \right\|_F \right] \in \mathcal{O}(\ln(m) \sqrt{N-m} (m^{-a+1/2}))$. In particular,

as $m \rightarrow \text{rank } H^{(k)}$, the expected error goes to zero.

□

Proof of Lemma 4.8. Since $\mathbf{y}_1, \dots, \mathbf{y}_N \stackrel{\text{i.i.d.}}{\sim}$ Model 1.1, by the characterization of its distribution given in Proposition A.2, $\mathbf{y}_1, \dots, \mathbf{y}_N$ lie in a union of K affine subspaces, $\bigcup_{j=1}^K U_j$, $\dim U_j = R_j \leq R_{\max}$. Every point $\mathbf{y} \in U_j = \boldsymbol{\mu}_j + \text{range}(\Sigma_j)$ lies in the linear subspace $\text{span}\{\boldsymbol{\mu}_j\} + \text{range}(\Sigma_j)$, whose dimension is at most $R_j + 1$. Writing $\mathbf{y}$ in a basis of this subspace, every coordinate of $\mathbf{y}^{\otimes k}$ is then a homogeneous polynomial of degree k in at most $R_j + 1$ variables. Since the number of distinct monomials of degree k in $R_j + 1$ variables is $\binom{R_j+k}{k}$, $\mathbf{y}^{\otimes k}$ with $\mathbf{y} \in U_j$ span at most $\binom{R_j+k}{k}$ directions.

The global span is therefore contained in the sum of these K subspaces, with dimension bounded by $\sum_{j=1}^K \binom{R_j+k}{k}$. Then, $\text{rank } H^{(k)} \leq \sum_{j=1}^K \binom{R_j+k}{k} \leq K \binom{R_{\max}+k}{k}$. If we further assume that U_j is contained in a single linear subspace U with $\dim U \leq R_{\max}$ for all j , then all observations lie in the linear subspace U itself. In this case, we would not need the additional dimension for $\boldsymbol{\mu}_j$, and $\mathbf{y}^{\otimes k}$ span at most $\binom{\dim U+k-1}{k} \leq \binom{R_{\max}+k-1}{k}$ directions. $\square$

Proof of Theorem 4.10. The numerator of $\hat{w}_k^{[t]}$ in (3.5) can be written as

$$\begin{aligned} \sum_{i \in \mathcal{I}_k} \hat{S}_{ii}^{[t]} &= \sum_{i \in \mathcal{I}_k} \left(\frac{1}{N} \sum_{n=1}^N g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n)^\top \right)_{ii} = \sum_{i \in \mathcal{I}_k} \frac{1}{N} \sum_{n=1}^N g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n)_i^2 \\ &= \frac{1}{N} \sum_{n=1}^N \left\| \text{vec} \left(\mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) - \mathbf{y}_n^{\otimes k} \right) \right\|^2 = \frac{1}{N} \sum_{n=1}^N \left\| \mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) - \mathbf{y}_n^{\otimes k} \right\|^2 \\ &= \frac{1}{N} \sum_{n=1}^N \left\| \mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) \right\|^2 - 2 \left\langle \mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right), \mathbf{y}_n^{\otimes k} \right\rangle + \langle \mathbf{y}_n, \mathbf{y}_n \rangle^k. \end{aligned}$$

The denominator of $\hat{w}_k^{[t]}$ in (3.5) can be written as

$$\begin{aligned} \sum_{i \in \mathcal{I}_k} \sum_{j=1}^q \left(\hat{S}_{ij}^{[t]} \right)^2 &= \sum_{i \in \mathcal{I}_k} \sum_{j=1}^q \left(\frac{1}{N} \sum_{n=1}^N g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n)^\top \right)_{ij}^2 \\ &= \frac{1}{N^2} \sum_{n, n'=1}^N \sum_{i \in \mathcal{I}_k} \left(\sum_{j=1}^q \left(g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n)^\top \right)_{ij} \left(g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'}) g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'})^\top \right)_{ij} \right) \\ &= \frac{1}{N^2} \sum_{n, n'=1}^N \left\langle g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) [i_k : i_k'], g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'}) [i_k : i_k'] \right\rangle \text{Tr} \left(g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'})^\top \right) \\ &= \frac{1}{N^2} \sum_{n, n'=1}^N \left\langle g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) [i_k : i_k'], g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'}) [i_k : i_k'] \right\rangle \text{Tr} \left(g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'})^\top g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) \right) \\ &= \frac{1}{N^2} \sum_{n, n'=1}^N \left\langle g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n) [i_k : i_k'], g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'}) [i_k : i_k'] \right\rangle \left\langle g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_n), g(\hat{\boldsymbol{\theta}}^{[t-1]}; \mathbf{y}_{n'}) \right\rangle \\ &= \frac{1}{N^2} \sum_{n, n'=1}^N \left\langle \mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) - \mathbf{y}_n^{\otimes k}, \mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) - \mathbf{y}_{n'}^{\otimes k} \right\rangle \sum_{k'=1}^L \left\langle \mathcal{M}^{(k')} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) - \mathbf{y}_n^{\otimes k'}, \mathcal{M}^{(k')} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) - \mathbf{y}_{n'}^{\otimes k'} \right\rangle \end{aligned}$$

Combining the numerator and the denominator, substituting with the quantities in (4.4), and simplifying, we get the expression for DGMM weights in (4.9). Observe that $\hat{w}_k^{[t]}$ requires (1) $\phi_k^{[t-1]} = \left\| \mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right) \right\|^2$, $k = 1, \dots, L$, which, by Theorem 4.4, takes $\mathcal{O}(L^2 K^2 + L K^2 d R_{\max}^2)$ operations, and (2) $\psi_{k,n}^{[t-1]} = \left\langle \mathcal{M}^{(k)} \left(\hat{\boldsymbol{\theta}}^{[t-1]} \right), \mathbf{y}_n^{\otimes k} \right\rangle$, $k = 1, \dots, L, n = 1, \dots, N$, which, by Theorem 4.5, takes $\mathcal{O}(NLK + NKdR_{\max})$ operations. In total, $\hat{w}_k^{[t]}$ in (3.5) can be computed in $\mathcal{O}(L^2 K^2 + L K^2 d R_{\max}^2 + NLK + NKdR_{\max})$. $\square$

Proof of Theorem 4.11. We rewrite the DGMM objective function in (3.5):

$$Q_N^{[t]}(\boldsymbol{\theta}) = \bar{g}_N(\boldsymbol{\theta})^\top \text{diag} \left(\underbrace{\hat{w}_1^{[t]}, \dots, \hat{w}_1^{[t]}}_{d \text{ copies}}, \dots, \underbrace{\hat{w}_L^{[t]}, \dots, \hat{w}_L^{[t]}}_{d^L \text{ copies}} \right) \bar{g}_N(\boldsymbol{\theta})$$

$$\begin{aligned}
&= \sum_{k=1}^L \hat{w}_k^{[t]} \sum_{i \in \mathcal{I}_k} \left(\text{vec} \left(\mathcal{M}^{(k)}(\boldsymbol{\theta}) - \frac{1}{N} \sum_{n=1}^N \mathbf{y}_n^{\otimes k} \right)_i \right)^2 = \sum_{k=1}^L \hat{w}_k^{[t]} \left\| \mathcal{M}^{(k)}(\boldsymbol{\theta}) - \frac{1}{N} \sum_{n=1}^N \mathbf{y}_n^{\otimes k} \right\|^2 \\
&= \sum_{k=1}^L \hat{w}_k^{[t]} \left(\left\| \mathcal{M}^{(k)}(\boldsymbol{\theta}) \right\|^2 - \frac{2}{N} \sum_{n=1}^N \langle \mathcal{M}^{(k)}(\boldsymbol{\theta}), \mathbf{y}_n^{\otimes k} \rangle + \frac{1}{N^2} \sum_{n=1}^N \sum_{n'=1}^N \langle \mathbf{y}_n^{\otimes k}, \mathbf{y}_{n'}^{\otimes k} \rangle \right), \\
&= \sum_{k=1}^L \hat{w}_k^{[t]} \left(\phi^{(k)}(\boldsymbol{\theta}) - \frac{2}{N} \sum_{n=1}^N \psi^{(k)}(\boldsymbol{\theta}; \mathbf{y}_n) + \frac{1}{N^2} \sum_{n=1}^N \sum_{n'=1}^N C_{k,n,n'} \right). \tag{B.24}
\end{aligned}$$

Observe that $\nabla_{\boldsymbol{\theta}} Q_N^{[t]}(\boldsymbol{\theta})$ requires the following gradients: $\nabla_{\pi_j} \phi^{(k)}(\boldsymbol{\theta})$, $\nabla_{\mu_j} \phi^{(k)}(\boldsymbol{\theta})$, $\nabla_{V_j} \phi^{(k)}(\boldsymbol{\theta})$ and $\nabla_{\pi_j} \psi^{(k)}(\boldsymbol{\theta}; \mathbf{y}_n)$, $\nabla_{\mu_j} \psi^{(k)}(\boldsymbol{\theta}; \mathbf{y}_n)$, $\nabla_{V_j} \psi^{(k)}(\boldsymbol{\theta}; \mathbf{y}_n)$, for $k = 1, \dots, L, n = 1, \dots, N$, which, by [Theorem 4.4](#) and [Theorem 4.5](#), takes $\mathcal{O}(L^2 K^2 + LK^2 dR_{\max}^2 + NLK + NKdR_{\max})$ operations in total. $\nabla_{\boldsymbol{\theta}} Q_N^{[t]}(\boldsymbol{\theta})$ is then obtained by vectorizing and concatenating these gradients. $\square$

Proof of Theorem 4.12. Let I be the number of L-BFGS iterations, T the number of GMM steps. The DGMM estimator, $\hat{\boldsymbol{\theta}}^{(\text{DGMM})}$, requires the following computations:

- (1) For one time, pre-compute $\sum_{n'=1}^N C_{k,n,n'}$, for $k = 1, \dots, 2L, n = 1, \dots, N$, and store in a $(2L+1) \times N$ matrix. By [Theorem 4.6](#), this requires $\mathcal{O}(NLdm)$ operations.
- (2) For at most T times, compute $\hat{w}_k^{[t-1]}$ for $k = 1, \dots, L$, which, by [Theorem 4.10](#), takes $\mathcal{O}(TL^2 K^2 + TLK^2 dR_{\max}^2 + TNLK + TNKdR_{\max})$ operations.
- (3) For at most I times, compute the necessary gradients, which, by [Theorem 4.11](#), takes $\mathcal{O}(IL^2 K^2 + ILK^2 dR_{\max}^2 + INLK + INKdR_{\max})$ operations.

Taking the dominant terms and absorbing the stopping criteria constants, we get that the overall computational complexity of the DGMM estimator $\hat{\boldsymbol{\theta}}^{(\text{DGMM})}$ in (3.5) is $\mathcal{O}(L^2 K^2 + LK^2 dR_{\max}^2 + NLK + NKdR_{\max} + NLdm)$.

In contrast, the MM estimator, $\hat{\boldsymbol{\theta}}^{(\text{MM})}$, requires the following computations:

- (1) For one time, pre-compute the sample moments, which takes $\mathcal{O}(Nq)$, where $q = d + d^2 + \dots + d^L$. Taking the dominant terms, we get $\mathcal{O}(Nd^L)$.
- (2) The weighting matrix is the identity matrix and therefore takes $\mathcal{O}(1)$ time to compute.
- (3) For at most I times, compute the moment condition vector $\bar{g}_N(\boldsymbol{\theta})$ in $\mathcal{O}(Nq)$ operations, the objective function $\bar{g}_N(\boldsymbol{\theta})^\top \bar{g}_N(\boldsymbol{\theta})$ in $\mathcal{O}(q^2)$ operations, the Jacobian matrix $G(\boldsymbol{\theta}) = \frac{\partial \bar{g}_N(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{q \times p}$ in $\mathcal{O}(Nqp)$ operations, and the gradients $\nabla_{\boldsymbol{\theta}} \bar{g}_N(\boldsymbol{\theta})^\top \bar{g}_N(\boldsymbol{\theta}) = 2G(\boldsymbol{\theta})^\top \bar{g}_N(\boldsymbol{\theta})$ in $\mathcal{O}(qp)$ operations. Taking the dominant term and absorbing the stopping criteria constants, we get $\mathcal{O}(INd^{L+1}KR_{\max} + Id^{L+1}KR_{\max}) = \mathcal{O}(Nd^{L+1}KR_{\max})$.

Similarly, the GMM estimator, $\hat{\boldsymbol{\theta}}^{(\text{GMM})}$, requires the following computations:

- (1) For one time, pre-compute the sample moments, which takes $\mathcal{O}(Nq)$.
- (2) For at most T times, compute $\hat{S}^{[t]}$ in $\mathcal{O}(Nq^2)$ and invert $\hat{S}^{[t]}$ in $\mathcal{O}(q^3)$.
- (3) For at most I times, compute the moment condition vector $\bar{g}_N(\boldsymbol{\theta})$ in $\mathcal{O}(Nq)$ operations, the objective function $\bar{g}_N(\boldsymbol{\theta})^\top \hat{W}^{[t]} \bar{g}_N(\boldsymbol{\theta})$ in $\mathcal{O}(q^2)$ operations, the Jacobian matrix $G(\boldsymbol{\theta}) = \frac{\partial \bar{g}_N(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$ in $\mathcal{O}(Nqp)$ operations, and the gradients $\nabla_{\boldsymbol{\theta}} \bar{g}_N(\boldsymbol{\theta})^\top \hat{W}^{[t]} \bar{g}_N(\boldsymbol{\theta}) = 2G(\boldsymbol{\theta})^\top \hat{W}^{[t]} \bar{g}_N(\boldsymbol{\theta})$ in $\mathcal{O}(q^2 p)$ operations. Taking the dominant term, we get $\mathcal{O}(INd^{L+1}KR_{\max} + Id^{2L+1}KR_{\max})$.

- (4) The overall computational complexity of obtaining $\hat{\boldsymbol{\theta}}^{(\text{GMM})}$ is $\mathcal{O}(Nd^{2L} + d^{3L} + Nd^{L+1}KR_{\max} + d^{2L+1}KR_{\max})$ after absorbing the stopping criteria constants.

□