Multiway Spherical Clustering via Degree-Corrected Tensor Block Models

Jiaxin Hu
Department of Statistics
University of Wisconsin-Madison
jhu267@wisc.edu

Miaoyan Wang
Department of Statistics
University of Wisconsin-Madison
miaoyan.wang@wisc.edu

Abstract

We consider the problem of multiway clustering in the presence of unknown degree heterogeneity. Such data problems arise commonly in applications such as recommendation system, neuroimaging, community detection, and hypergraph partitions in social networks. The allowance of degree heterogeneity provides great flexibility in clustering models, but the extra complexity poses significant challenges in both statistics and computation. Here, we develop a degree-corrected tensor block model with estimation accuracy guarantees. We present the phase transition of clustering performance based on the notion of angle separability, and we characterize three signal-to-noise regimes corresponding to different statistical-computational behaviors. In particular, we demonstrate that an intrinsic statistical-to-computational gap emerges only for tensors of order three or greater. Further, we develop an efficient polynomial-time algorithm that provably achieves exact clustering under mild signal conditions. The efficacy of our procedure is demonstrated through both simulations and analyses of Peru Legislation dataset.

1 Introduction

Multiway arrays have been widely collected in various fields including social networks [4], neuroscience [22], and computer science [18]. Tensors effectively represent the multiway data and serve as the foundation in higher-order data analysis. One data example is from multi-tissue multi-individual gene expression study [23, 14], where the data tensor consists of expression measurements indexed by (gene, individual, tissue) triplets. Another example is *hypergraph* network [12, 11, 3, 15] in social science. A K-uniform hypergraph can be naturally represented as an order-K tensor, where each entry indicates the presence of K-way hyperedge among nodes (a.k.a. entities). In both examples, identifying the similarity among tensor entities is important for scientific discovery.

This paper studies the problem of multiway clustering based on a data tensor. The goal of multiway clustering is to identify a checkerboard structure from a noisy data tensor. Fig. 1 illustrates the noisy tensor and the underlying structures discovered by multiway clustering methods. The checkerboard structure serves as a meta tool to many popular structures including the low-rankness [26], latent space models [24], and isotonic models [21]. In the hypergraph example, the multiway clustering aims to identify the underlying block partition of nodes based on their higher-order connectivities; therefore, we also refer to the clustering as *higher-order clustering*. The most common model for higher-order clustering is called *tensor block model* (TBM) [7, 25], which extends the usual matrix stochastic block model [1] to tensors. The matrix analysis tools, however, are sub-optimal for higher-order clustering. Tensor tools for solving block models have received increased interest recently [13, 25].

TBM suffers from drawbacks to model real world data in spite of the popularity. The key underlying assumption of block model is that all nodes in the same community are exchangeable; i.e., the nodes have no individual effects apart from the block effects. However, the exchangeability assumption is often non-realistic. Each node may contribute to the data variation by its own multiplicative effect.

Second Workshop on Quantum Tensor Networks in Machine Learning. In conjunction with 35th NeurIPS, 2021

Such degree heterogeneity appears commonly in social networks. For example, regular block model fails to model the Karate Club network [5] without addressing degree heterogeneity.

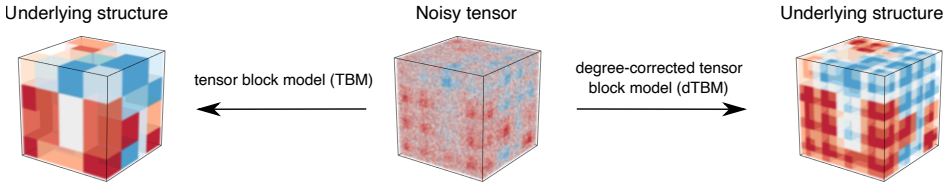

Figure 1: Examples for order-3 tensor block model (TBM) with and without degree correction. Both TBM and dTBM have four communities on each mode, while dTBM allows a richer structure with degree heterogeneity.

We develop the *degree-corrected tensor block model* (dTBM) to account for the degree heterogeneity. The dTBM combines a higher-order checkerboard structure with degree parameter $\boldsymbol{\theta} = (\theta(1), \ldots, \theta(p))^T$ to allow heterogeneity among p nodes. Fig. 1 compares the underlying structures of TBM and dTBM with the same number of communities. The dTBM allows varying values within the same community, thereby allowing a richer structure.

Our contributions. The primary goal of this paper is to provide both statistical and computational guarantees for dTBM. Our main contributions are summarized below.

- We develop a general dTBM and establish the identifiability for the uniqueness of clustering using the notion of angle seperability.
- We present the phase transition of clustering performance with respect to three different statistical and computational behaviors. We characterize, for the first time, the critical signal-to-noise (SNR) thresholds in dTBMs, revealing the intrinsic distinctions between (matrix) biclustering and (tensor) higher-order clustering. Specific SNR thresholds and algorithm behaviours are depicted in Fig. 2.
- We provide an angle-based algorithm that achieves exact clustering *in polynomial time* under mild conditions. Simulation and data studies demonstrate the outperformance of our algorithm compared with existing higher-order clustering algorithms.

The last two contributions, to our best knowledge, are new to the literature of dTBMs.

	Gao et al [10]	Han et al [13]	Ke et al [15]	Ghoshdastidar et al [11]	**Ours**
Applicable to tensors	×	√	√	√	√
Allow degree heterogeneity	√	×	√	√	√
Allow various data types	×	√	×	√	√
Misclustering rate (for order 2)	$\exp(-p)$	$\exp(-p)$	p^{-1}	p^{-1}	$\exp(-p)$

Table 1: Comparison between previous methods with our method.

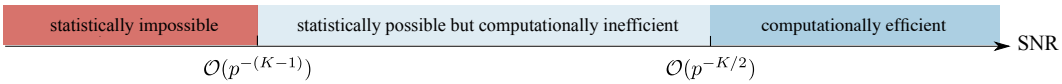

Figure 2: SNR thresholds for statistical and computational limits in order-K dTBM with dimension $(p, ..., p)$. The SNR gap between statistical possibility and computational efficiency exists only for tensors with $K \geq 3$.

Related work. We emphasize the comparisons that set our work apart from earlier literature from two perspectives—models and algorithms. From the model perspective, our work extends the previous degree-corrected model from matrices to tensors. There is a huge literature on degree-corrected matrix models; see the review paper [1] and the references therein. The tensor counterparts, however, are relatively less understood. Tab. 1 summarizes the most relevant models for higher-order clustering. Earlier tensor methods either fail to allow degree heterogeneity [9, 13, 29, 12, 16], or suffer from sub-optimal misclustering rates [15, 11, 7]. In contrast, our method addresses the degree heterogeneity, allows discrete and continuous entries, and achieves exponentially fast rate in clustering tasks.

From the algorithm perspective, our methods generalize the recent global-to-local strategy for matrix learning [10, 8, 27] to tensors [13, 2, 16]. Despite the conceptual similarity, we address several fundamental challenges associated with this non-convex, non-continuous problem. We show the insufficiency of the conventional tensor HOSVD [17], and we develop a weighted higher-order initialization that relaxes the eigen-gap separation condition. Furthermore, our local iteration leverages the angle-based clustering in order to avoid explicit estimation of degree heterogeneity. Our

bounds reveal the interesting interplay between the computational and statistical errors. We show that our final estimate provably achieves the exact clustering within only polynomial-time complexity.

Notation. We use lower-case letters (e.g., a, θ) for scalars, lower-case boldface letters (e.g., $\boldsymbol{a}, \boldsymbol{\theta}$) for vectors, upper-case boldface letters (e.g., $\boldsymbol{X}, \boldsymbol{Y}$) for matrices, and calligraphy letters (e.g., $\mathcal{X}, \mathcal{Y}$) for tensors of order three or greater. We use $|\cdot|$ for the cardinality of a set and $\mathbb{1}\{\cdot\}$ for the indicator function. For an integer $p \in \mathbb{N}_+$, we use the shorthand $[p] = \{1, 2, ..., p\}$. For a p-length vector $\boldsymbol{a} = (a_1, \ldots, a_p)$, we use $\boldsymbol{a}_I$ to denote the sub-vector by restricting the indices in the set $I \subset [p]$. We use $\|\boldsymbol{a}\| = (\sum_i a_i^2)^{1/2}$ to denote the ℓ_2-norm, $\|\boldsymbol{a}\|_1 = \sum_i |a_i|$ to denote the ℓ_1 norm of $\boldsymbol{a}$. For two vector $\boldsymbol{a}, \boldsymbol{b}$ of the same dimension, we denote the angle between $\boldsymbol{a}, \boldsymbol{b}$ by

$$\cos(\boldsymbol{a}, \boldsymbol{b}) = \langle \boldsymbol{a}, \boldsymbol{b} \rangle / \|\boldsymbol{a}\| \|\boldsymbol{b}\|,$$

where $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ is the inner product of two vectors and $\cos(\boldsymbol{a}, \boldsymbol{b}) \in [-1, 1]$. We make the convention that $\cos(\boldsymbol{a}, \boldsymbol{b}) = \cos(\boldsymbol{a}^T, \boldsymbol{b}^T)$. For a matrix $\boldsymbol{Y}$, we use $\boldsymbol{Y}_{i:}$ to denote the i-th row of the matrix. We let $\mathcal{Y} = [\![\mathcal{Y}(i_1, \ldots, i_K)]\!] \in \mathbb{R}^{p_1 \times \cdots \times p_K}$ denote an order-K $(p_1, ..., p_K)$-dimensional tensor. The multilinear multiplication of a tensor $\mathcal{S} \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ by matrices $\boldsymbol{M}_k \in \mathbb{R}^{p_k \times r_k}$ results in an order-d $(p_1, \ldots, p_K)$-dimensional tensor $\mathcal{X}$, denoted

$$\mathcal{X} = \mathcal{S} \times_1 \boldsymbol{M}_1 \times \cdots \times_K \boldsymbol{M}_K,$$

where the entries of $\mathcal{X}$ are defined by

$$\mathcal{X}(i_1, \ldots, i_K) = \sum_{(j_1, \ldots, j_K)} \mathcal{S}(j_1, \ldots, j_K) \boldsymbol{M}_1(i_1, j_1) \cdots \boldsymbol{M}_K(i_K, j_K).$$

We use $\mathrm{Ave}(\cdot)$ to denote the operation of taking averages across elements and $\mathrm{Mat}_k(\cdot)$ to denote the unfolding operation that reshapes the tensor along mode k into a matrix. For a symmetric tensor $\mathcal{Y} \in \mathbb{R}^{p \times \cdots \times p}$, we omit the subscript and use $\mathrm{Mat}(\mathcal{Y}) \in \mathbb{R}^{p \times p^{K-1}}$ to denote the unfolding. For two sequences $\{a_p\}, \{b_p\}$, we denote $a_p \lesssim b_p$ if $\lim_{p \to \infty} a_p / b_p \leq c$ and $a_p = \Omega(b_p)$ if $c b_p \leq a_p \leq C b_p$, for some constants $c, C \geq 0$. We use the terms "community" and "clusters" exchangeably.

2 Model formulation

2.1 Degree-corrected tensor block model

Suppose we have an order-K data tensor $\mathcal{Y} \in \mathbb{R}^{p \times \cdots \times p}$. For ease of notation, we focus on symmetric tensors in this section; our framework easily extends to general asymmetric tensors. Assume there exist $r \geq 2$ disjoint communities among the p nodes. We represent the community assignment by a function $z \colon [p] \mapsto [r]$, where $z(i) = a$ for i-th node that belongs to the a-th community. Then, $z^{-1}(a) = \{i \in [p] \colon z(i) = a\}$ denotes the set of nodes that belong to the a-th community, and $|z^{-1}(a)|$ denotes the number of nodes in the a-th community. Let $\boldsymbol{\theta} = (\theta(1), \ldots, \theta(p))^T$ denote the degree heterogeneity for p nodes. We consider the order-K dTBM [11, 15],

$$\mathcal{Y}(i_1, \ldots, i_K) = \mathcal{S}(z(i_1), \ldots, z(i_K)) \prod_{k=1}^{K} \theta_{i_k} + \mathcal{E}(i_1, \ldots, i_K),$$

where $\mathcal{S} \in \mathbb{R}^{r \times \cdots \times r}$ is an order-K tensor collecting the block means among communities, and $\mathcal{E} \in \mathbb{R}^{p \times \cdots \times p}$ is a noise tensor consisting of independent mean-zero sub-Gaussian entries with variance bounded by σ^2. The unknown parameters are z, S, and $\boldsymbol{\theta}$. The dTBM can be equivalently written in a compact form of tensor-matrix product:

$$\mathbb{E}\mathcal{Y} = \mathcal{S} \times_1 \boldsymbol{\Theta}\boldsymbol{M} \times_2 \cdots \times_K \boldsymbol{\Theta}\boldsymbol{M}, \tag{1}$$

where $\boldsymbol{\Theta} = \mathrm{diag}(\theta(1), ..., \theta(p)) \in \mathbb{R}^{p \times p}$ is a diagonal matrix, $\boldsymbol{M} \in \{0, 1\}^{p \times r}$ is the membership matrix associated with community assignment z such that $\boldsymbol{M}(i, j) = \mathbb{1}\{z(i) = j\}$. By definition, each row of $\boldsymbol{M}$ has one copy of 1's and 0's elsewhere. Note that the discrete nature of $\boldsymbol{M}$ renders our model (1) more challenging than Tucker decomposition. We call a tensor $\mathcal{X}$ an r-block tensor with degree $\boldsymbol{\theta}$ if $\mathcal{X}$ admits dTBM (1). Here, we give two special cases of dTBM.

Example 1 (Gaussian TBM)**.** Let $\theta(i) = 1$ for all $i \in [p]$ and $\mathcal{E}$ be a noise tensor with i.i.d. $N(0, \sigma^2)$ entries. Our dTBM reduces to a non-degree Gaussian TBM [25, 13], which is widely used in previous clustering algorithms [25, 7]. The theoretical results in TBM serve as benchmarks for dTBM.

Example 2 (Binary dTBM)**.** Consider a K-uniform hypergraph $H = (V, E)$, where $V = [p]$ collects the nodes with r disjoint communities and E collects all the K-way hyperedges. Let $\mathcal{Y} \in \{0,1\}^{p\times\cdots\times p}$ denote the adjacency tensor, where the entries encode the presence or absence of hyperdeges among p nodes. Specifically, let $\mathcal{Y}(i_1, ..., i_K) = 1$ if $(i_1, ..., i_K) \in E$, otherwise, $\mathcal{Y}(i_1, ..., i_K) = 0$, for all $(i_1, \ldots, i_K) \in [p]^K$. The equation (1) models $\mathbb{E}\mathcal{Y}$ with unknown degree heterogeneity and subgaussianity parameter $\sigma^2 = 1/4$.

Remark 1 (Comparison with non-degree models)**.** Our dTBM uses fewer block parameters than TBM. Let the subscripts "deg" and "non" denote quantities in the models with and without degrees, respectively. Then, every r_{non}-block tensor can be represented by a degree-corrected r_{deg}-block tensor with $r_{\text{deg}} \le r_{\text{non}}$. In particular, there exist tensors with $r_{\text{non}} = p$ but $r_{\text{deg}} = 1$, so the reduction in r can be dramatic from p to 1. This fact highlights the benefits of introducing degree heterogeneity in higher-order clustering tasks.

2.2 Identifiability under angle gap condition

The goal of clustering is to estimate the partition function z from model (1). We use $\mathcal{P}$ to denote the following parameter space for $(z, \mathcal{S}, \boldsymbol{\theta})$,

$$\left\{(z,\mathcal{S},\boldsymbol{\theta})\colon\ \boldsymbol{\theta}\in\mathbb{R}_+^p,\ \frac{c_1 p}{r}\le|z^{-1}(a)|\le\frac{c_2 p}{r}, c_3\le\|\text{Mat}(\mathcal{S})_{a:}\|\le c_4, \left\|\boldsymbol{\theta}_{z^{-1}(a)}\right\|_1=|z^{-1}(a)|, a\in[r]\right\}, \quad (2)$$

where $c_i > 0$'s are universal constants. First, the entrywise positivity constraint on $\boldsymbol{\theta} \in \mathbb{R}_+^p$ is imposed to avoid sign ambiguity between entries in $\boldsymbol{\theta}_{z^{-1}(a)}$ and $\mathcal{S}$. This constraint can be achieved without sacrificing model flexibility, by using a slightly larger dimension of $\mathcal{S}$ in the factorization (1) (see Supplement). Second, the constants c_1, c_2 in the $|z^{-1}(a)|$ bound assume the roughly balanced size across r communities. Third, the constants c_3, c_4 in the magnitude of $\text{Mat}(\mathcal{S})_{a:}$ requires no purely zero slide in $\mathcal{S}$, so the core tensor $\mathcal{S}$ is not trivially reduced to lower rank. Lastly, the ℓ_1 normalization $\|\boldsymbol{\theta}_{z^{-1}(a)}\|_1 = |z^{-1}(a)|$ is imposed to avoid the scalar ambiguity between $\boldsymbol{\theta}_{z^{-1}(a)}$ and $\mathcal{S}$. This constraint, again, incurs no restriction to model flexibility but makes our presentation cleaner.

We first provide the identifiablity conditions for our model before estimation procedures. When $r = 1$, the decomposition (1) is always unique (up to cluster label permutation) in $\mathcal{P}$, because dTBM is equivalent to the rank-1 tensor family under this case. When $r \ge 2$, the Tucker rank of signal tensor $\mathbb{E}\mathcal{Y}$ in (1) is bounded by, but not necessarily equal to, the number of blocks r [25]. Therefore, one can not apply the classical identifiability conditions for low-rank tensors to dTBM. Here, we introduce a key separation condition on the core tensor.

Assumption 1 (Angle gap)**.** Let $\boldsymbol{S} = \text{Mat}(\mathcal{S})$. Assume the minimal gap between normalized rows of $\boldsymbol{S}$ is bounded away from zero, i.e.,

$$\Delta_{\min} := \min_{a\neq b\in[r]}\left\|\frac{\boldsymbol{S}_{a:}}{\|\boldsymbol{S}_{a:}\|}-\frac{\boldsymbol{S}_{b:}}{\|\boldsymbol{S}_{b:}\|}\right\| > 0. \quad (3)$$

Equivalently, none of the two rows in $\boldsymbol{S}$ are parallel, i.e., $\max_{a\neq b\in[r]}\cos(\boldsymbol{S}_{a:}, \boldsymbol{S}_{b:}) = 1 - \Delta_{\min}^2/2$.

The quantity $\Delta_{\min}$ characterizes the non-redundancy among clusters measured by angle separation. The definition (3) is well posed because of the lower bound on $\|\boldsymbol{S}_{a:}\|$ in (2). The following theorem shows that the angle gap separation is sufficient and necessary for dTBM parameter identifiability.

Theorem 1 (Identifiability)**.** Consider the dTBM with $r \ge 2$. The parameterization (1) is unique in $\mathcal{P}$ up to cluster label permutations, if and only if Assumption 1 holds.

The identifiability guarantee for the dTBM is more appealing than classical Tucker model. In the Tucker model, the factor matrix $\boldsymbol{M}$ is identifiable only up to orthogonal rotations. In contrast, our model does not suffer from rotational invariance. As we will show in Section 4, each column of the membership matrix $\boldsymbol{M}$ can be precisely recovered under our algorithm. This property benefits the interpretation of dTBM in practice.

3 Statistical-computational gaps for tensors of order $K \ge 3$

In this section, we study the statistical and computational limits of dTBM. We propose signal-to-noise ratio (SNR) by

$$\text{SNR} := \Delta_{\min}^2/\sigma^2 = p^{\gamma}, \quad (4)$$

with varying $\gamma > 0$ that quantifies different regimes of interest. We call γ the *signal exponent*. Intuitively, a larger SNR, or equivalently a larger γ, benefits the clustering in the presence of noise. With quantification (4), we consider the following parameter space,

$$\mathcal{P}(\gamma) = \mathcal{P} \cap \{\mathcal{S} \text{ satisfies SNR condition (4) with } \gamma\}. \tag{5}$$

Our goal is to characterize the clustering accuracy with respect to γ. Let $\hat{z}$ and z be estimated and true clustering functions in the family (2). Define the misclustering error by

$$\ell(\hat{z}, z) = \frac{1}{p} \min_{\pi \in \Pi} \sum_{i \in [p]} \mathbb{1}\{\hat{z}(i) \neq \pi \circ z(i)\},$$

where $\pi : [r] \mapsto [r]$ is a permutation of cluster labels, $\circ$ denotes the composition operation, and Π denotes the collection of all possible permutations. The infinitum over all permutations accounts for the ambiguity in cluster label permutation.

In Sections 3.1-3.2, we provide the lower bounds of $\ell(\hat{z}, z)$ for general Gaussian dTBMs without symmetric assumptions. For general (asymmetric) dTBMs, we extend the parameter space (2) to allow K clustering functions $(z_k)_{k \in [K]}$, one for each mode. For notational simplicity, we still use z and $\mathcal{P}(\gamma)$ for this general (asymmetric) model. All lower bounds should be interpreted as the worst-case results across K modes.

3.1 Statistical limits

Our first main result is to show the minimax lower bound of SNR for exact recovery in dTBM.

Theorem 2 (Statistical lower bound)**.** Consider general Gaussian dTBMs under the parameter space $\mathcal{P}(\gamma)$ (5) with $K \geq 1$. Assume $r \lesssim p^{1/3}$. If the signal exponent satisfies $\gamma < -(K-1)$, then, every estimator $\hat{z}_{\text{stat}}$ obeys

$$\sup_{(z, \mathcal{S}, \boldsymbol{\theta}) \in \mathcal{P}(\gamma)} \mathbb{E}\left[p\ell(\hat{z}_{\text{stat}}, z)\right] \geq 1.$$

Theorem 2 demonstrates the impossibility of exact recovery of the assignment when $\gamma < -(K-1)$ in the high-dimensional regime $p \to \infty$ for fixed r. The proof is information-theoretical, and therefore the results apply to all statistical estimators, including but not limited to, maximum likelihood estimation (MLE) [25, 15] and trace maximization [12]. Our derived SNR threshold $-(K-1)$ is also a minimax upper bound, because MLE achieves exact recovery when $\gamma > -(K-1)$. Hence, the boundary $\gamma_{\text{stat}} := -(K-1)$ is the critical value for statistical performance of dTBM.

3.2 Computational limits

An important ingredient to establish the computational limits is the *hypergraphic planted clique (HPC) conjecture* [28, 6]. The HPC indicates the impossibility of fully recovering the planted cliques with polynomial-time algorithm when the clique size is less than the number of vertices in the hypergraph. The formal statement of HPC conjecture can be found in Supplement. Under the HPC conjecture, we establish the SNR lower bound that is necessary for any *polynomial-time* estimator to achieve exact clustering.

Theorem 3 (Computational lower bound)**.** Consider general Gaussian dTBMs under the parameter space $\mathcal{P}(\gamma)$ with $K \geq 2$. Assume HPC conjecture holds. If the signal exponent $\gamma < -K/2$, then, every *polynomial-time estimator* $\hat{z}_{\text{comp}}$ obeys

$$\liminf_{p \to \infty} \sup_{(z, \mathcal{S}, \boldsymbol{\theta}) \in \mathcal{P}(\gamma)} \mathbb{E}\left[p\ell(\hat{z}_{\text{comp}}, z)\right] \geq 1.$$

Theorem 3 indicates the impossibility of exact recovery by polynomial-time algorithms when $\gamma < -K/2$. In Section 4, we will show the condition $\gamma > -K/2$ suffices for our proposed polynomial-time estimator. Thus, $\gamma_{\text{comp}} := -K/2$ is the critical value for computational performance of dTBM.

Remark 2 (Statistical-computational gaps)**.** We have established the phase transition of exact clustering under dTBM combing Theorems 2 and 3. Fig. 2 summarizes our results of critical SNRs. We find that the statistical-to-computational gap emerges only for higher-order tensors with $K \geq 3$. This result reveals the intrinsic distinctions between matrix biclustering and higher-order clustering.

Remark 3 (Comparison with non-degree models)**.** We compare our results to non-degree tensor models. The allowance of degree heterogeneity $\boldsymbol{\theta}$ makes the model more flexible, but it incurs extra statistical and computational complexity. Fortunately, we find that the extra complexity does not render the estimation of z qualitatively harder; see the comparison of our phase transition with non-degree TBM [13].

4 Polynomial-time algorithm under mild SNR

We present a two-stage clustering algorithm. The procedure takes a global-to-local approach. See Fig. 3 for illustration. The global step finds the basin of attraction with polynomial miclustering error, whereas the local iterations improve the initial clustering to exact recovery. Both steps are critical to obtain a satisfactory algorithm output.

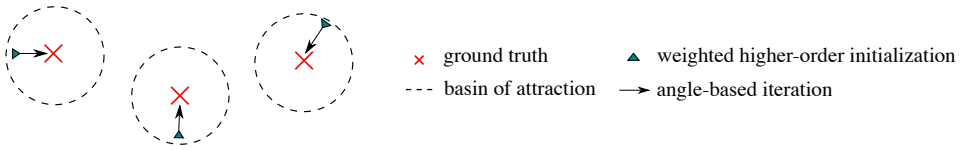

Figure 3: Illustration of our global-to-local algorithm.

4.1 Weighted higher-order initialization

We start with weighted higher-order clustering algorithm as initialization. To gain insights, we use an order-3 symmetric tensor as a working example. Consider noiseless case with $\mathcal{X} = \mathbb{E}\mathcal{Y}$ and $\boldsymbol{X} = \mathrm{Mat}(\mathcal{X})$. By model (1), for all $i \in [p]$, we have

$$\theta(i)^{-1}\boldsymbol{X}_{i:} = [\mathrm{Mat}(\mathcal{S} \times_2 \boldsymbol{\Theta}\boldsymbol{M} \times_3 \boldsymbol{\Theta}\boldsymbol{M})]_{z(i):}.$$

This implies that, all node i belonging to a-th community (i.e., $z(i) = a$) share the same normalized mean vector $\theta(i)^{-1}\boldsymbol{X}_{i:}$, and vice versa. Intuitively, one can apply k-means clustering to the vectors $\{\theta(i)^{-1}\boldsymbol{X}_{i:}\}_{i\in[p]}$, which leads to main idea of our Sub-algorithm 1. Specifically, our initialization consists of denoising step and clustering step. The denoising step (lines 1-2 in Sub-algorithm 1) estimates $\mathcal{X}$ from $\mathcal{Y}$ by a double projection spectral method. The double projection improves usual matrix spectral methods in order to alleviate the noise tensor. The clustering step (lines 3-5 in Sub-algorithm 1) performs the weighted k-means clustering. The choice of weights is to bound the k-means objective function by the Frobenius-norm accuracy of $\hat{\mathcal{X}}$. Unlike existing clustering algorithm [15], we apply the clustering on the unfolded tensor $\hat{\boldsymbol{X}}$ rather than on the factors $\hat{\boldsymbol{U}}$, which relaxes the eigen-gap separation condition [10, 13]. Full procedures are provided in Sub-algorithm 1.

We now establish the misclustering error rate of initialization. We call $\boldsymbol{\theta}$ is balanced, if the relative extent of heterogeneity is comparable across clusters in that

$$\min_{a\in[r]}\|\boldsymbol{\theta}_{z^{-1}(a)}\| = (1 + o(1)) \max_{a\in[r]}\|\boldsymbol{\theta}_{z^{-1}(a)}\|. \tag{6}$$

Note that, the assumption (6) does not preclude degree heterogeneity. Indeed, within each of the clusters, the highest degree can be $\theta(i) = \Omega(p)$, whereas the lowest degree can be $\theta(i) = \mathcal{O}(1)$.

Theorem 4 (Error for weighted higher-order initialization)**.** Consider Gaussian dTBMs under the parameter space $\mathcal{P}$ and Assumption 1. Assume $\boldsymbol{\theta}$ is balanced and $\min_{i\in[p]}\theta(i) \geq c$ for some constant $c > 0$. Let $z^{(0)}$ denote the output of Sub-algorithm 1. With probability going to 1,

$$\ell(z^{(0)}, z) \lesssim \frac{r^K p^{-K/2}}{\mathrm{SNR}}. \tag{7}$$

Remark 4 (Comparison to previous results)**.** For fixed SNR, our initialization error rate with $K = 2$ agrees with the initialization error rate $\mathcal{O}(p)$ in matrix models [10]. Furthermore, in the special case of non-degree TBMs, we achieve the same initial misclassification error $\mathcal{O}(p^{-K/2})$ as in [13]. The result demonstrates the advantage of our algorithm in achieving both accuracy and model flexibility.

Remark 5 (Failure of conventional tensor HOSVD)**.** If we use conventional HOSVD for tensor denoising; that is, we use $\boldsymbol{U}_{\mathrm{pre}}$ in place of $\hat{\boldsymbol{U}}$ in line 2, then the misclustering rate becomes $\mathcal{O}(p)$ for all $K \geq 2$. This rate is substantially worse than our current rate (7).

4.2 Angle-based iteration

We propose an angle-based local iteration to improve the outputs from Sub-algorithm 1. To gain the intuition, consider an one-dimensional degree-corrected clustering problem with data vectors $\boldsymbol{x}_i = \theta(i)\boldsymbol{s}_{z(i)} + \boldsymbol{\epsilon}_i, i \in [p]$, where $\boldsymbol{s}_i$'s are known cluster centroids, $\theta(i)$'s are unknown degrees, and

Algorithm: Multiway spherical clustering for degree-corrected tensor block model

Sub-algorithm 1: Weighted higher-order initialization

Input: Observation $\mathcal{Y} \in \mathbb{R}^{p\times\cdots\times p}$, number of cluster r, relaxation factor $\eta > 1$ in k-means clustering.

1: Compute factor matrix $\boldsymbol{U}_{\text{pre}} = \text{SVD}_r(\text{Mat}(\mathcal{Y}))$ and the $(K-1)$-mode projection $\mathcal{X}_{\text{pre}} = \mathcal{Y}\times_1 \boldsymbol{U}_{\text{pre}}^T \times_2 \cdots \times_{K-1} \boldsymbol{U}_{\text{pre}}^T$.
2: Compute factor matrix $\hat{\boldsymbol{U}} = \text{SVD}_r(\text{Mat}(\mathcal{X}_{\text{pre}}))$ and denoised tensor $\hat{\mathcal{X}} = \mathcal{Y} \times_1 \hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T \times_2 \cdots \times_K \hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T$.
3: Let $\hat{\boldsymbol{X}} = \text{Mat}(\hat{\mathcal{X}})$ and $I_0 = \{i \in [p] : \|\hat{\boldsymbol{X}}_{i:}\| = 0\}$. Set $\hat{z}(i)$ randomly in $[r]$ for $i \in I_0$.
4: For all $i \in I_0^c$, compute normalized rows $\hat{\boldsymbol{X}}^s_{i:} := \|\hat{\boldsymbol{X}}_{i:}\|^{-1}\hat{\boldsymbol{X}}_{i:}$.
5: Solve the clustering $\hat{z}\colon [p] \to [r]$ and centroids $(\hat{\boldsymbol{x}}_j)_{j\in[r_k]}$ using weighted k-means, such that

$$\sum_{i\in I_0^c} \|\hat{\boldsymbol{X}}_{i:}\|^2 \|\hat{\boldsymbol{X}}^s_{i:} - \hat{\boldsymbol{x}}_{\hat{z}(i)}\|^2 \le \eta \min_{\bar{\boldsymbol{x}}_j, j\in[r], \hat{z}(i), i\in I_0^c} \sum_{i\in S^c} \|\hat{\boldsymbol{X}}_{i:}\|^2 \|\hat{\boldsymbol{X}}^s_{i:} - \bar{\boldsymbol{x}}_{\hat{z}(i)}\|^2.$$

Output: Initial clustering $z^{(0)} \leftarrow \hat{z}$.

Sub-algorithm 2: Angle-based iteration

Input: Observation $\mathcal{Y} \in \mathbb{R}^{p\times\cdots\times p}$, initialization $z^{(0)}\colon [p] \to [r]$ from Sub-algorithm 1, iteration number T.

6: **for** $t = 0$ to $T-1$ **do**
7: Update the block tensor $\mathcal{S}^{(t)}$ via $\mathcal{S}^{(t)}(i_1, ..., i_K) = \text{Ave}\{\mathcal{Y}(i_1, \ldots, i_K) : z^{(t)}(i_k) = j_k, k \in [K]\}$.
8: Calculate reduced tensor $\mathcal{Y}^{\text{d}} \in \mathbb{R}^{p\times r\times\cdots\times r}$ via

$$\mathcal{Y}^{\text{d}}(i, a_2, \ldots, a_K) = \text{Ave}\{\mathcal{Y}(i, i_2, \ldots, i_K) : z^{(t)}(i_k) = a_k, k \ne 1\}.$$

9: Let $\boldsymbol{Y}^{\text{d}} = \text{Mat}(\mathcal{Y}^{\text{d}})$ and $J_0 = \{i \in [p] : \|\boldsymbol{Y}^{\text{d}}_{i:}\| = 0\}$. Set $z^{(t+1)}(i)$ randomly in $[r]$ for $i \in J_0$.
10: Let $\boldsymbol{S}^{(t)} = \text{Mat}(\mathcal{S}^{(t)})$. For all $i \in J_0^c$ update the cluster assignment by

$$z(i)^{(t+1)} = \arg\max_{a\in[r]} \cos\left(\boldsymbol{Y}^{\text{d}}_{i:},\ \boldsymbol{S}^{(t)}_{a:}\right).$$

11: **end for**

Output: Estimated clustering $z^{(T)} \in [r]^p$.

$z\colon [p] \mapsto [r]$ is the clustering assignment of interest. The angle-based k-means algorithm estimates the assignment z by minimizing the angle between data vectors and centroids; i.e.,

$$z(i) = \arg\max_{a\in[r]} \cos(\boldsymbol{x}_i,\ \boldsymbol{s}_a), \quad \text{for all } i \in [p].$$

The classical Euclidean-distance based clustering [13] fails to recover z in the presence of degree heterogeneity, even under noiseless case. In contrast, the proposed angle-based k-means achieves accurate recovery without explicit estimation of $\boldsymbol{\theta}$. Our Sub-algorithm 2 shares the same spirit as angle-based k-means, except that we use estimated centroids $\boldsymbol{s}_a^{(t)}$ in place of $\boldsymbol{s}_a$ based on estimated assignment in previous iterations. Full procedures for our angle-based iteration are described in Sub-algorithm 2. The following theorem establishes the misclustering error rate of iterations.

Theorem 5 (Error for angle-based iteration)**.** Consider the setup as in Theorem 4. Suppose $r = \mathcal{O}(1)$ and $\text{SNR} \gtrsim p^{-K/2}\log p$. Let $z^{(t)}$ denote the t-th iteration output in Sub-algorithm 2 with initialization $z^{(0)}$ from Sub-algorithm 1. With probability going to 1, there exists a contraction parameter $\rho \in (0, 1)$ such that

$$\ell(z, \hat{z}^{(t+1)}) \lesssim \underbrace{\text{SNR}^{-1}\exp\left(-\frac{p^{K-1}\text{SNR}}{r^{K-1}}\right)}_{\text{statistical error}} + \underbrace{\rho^t \ell(z, z^{(0)})}_{\text{computational error}}.$$

The iteration error is decomposed into two parts: statistical error and computational error. The statistical error is unavoidable with noisy data regardless t, whereas the computational error decays in an exponential rate as the number of iterations $t \to \infty$. Theorem 5 implies that, with probability going to 1, our estimate $z^{(T)}$ achieves exact recovery within polynomial iterations; more precisely, for some permutation $\pi \in \Pi$,

$$z^{(T)} = \pi \circ z, \quad \text{for all } T \gtrsim \log_{1/\rho} p.$$

5 Numerical studies

We evaluate the performance of the weighted higher-order initialization and angle-based iteration in this section. We report average errors and standard deviations across 30 replications in each

experiment. Clustering accuracy is assessed by clustering error rate (CER, i.e., one minus rand index). Note that CER between $(\hat{z}, z)$ is equivalent to misclustering error $\ell(\hat{z}, z)$ up to constant multiplications [20], and a lower CER indicates a better performance.

We generate order-3 tensors with *assortative* [10] core tensors to control SNR; i.e., we set $\mathcal{S}_{aaa} = s_1$ for $a \in [r]$ and others be s_2, where $s_1 > s_2 > 0$. Let $\alpha = s_1/s_2$. We set α close to 1 such that $1 - \alpha = o(p)$. In particular, we have $\alpha = 1 + \Omega(p^{\gamma/2})$ by Assumption 1 and definition (4). Hence, we easily adjust SNR via varying α. Note that the assortative setting is proposed for simulations, and our algorithm is applicable for general tensors in practice. The cluster assignment z is randomly generated with equal probability across r clusters for each mode. Without further explanation, we generate degree heterogeneity $\boldsymbol{\theta}$ from absolute normal distribution as $\theta(i) = |X_i| + 1 - 1/\sqrt{2\pi}$ with $|X_i| \overset{\text{i.i.d.}}{\sim} N(0,1), i \in [p]$ and normalize $\boldsymbol{\theta}$ to satisfy (2). We set $\sigma^2 = 1$ for Gaussian data.

5.1 Verification for theoretical results

The first experiment verifies statistical-computational gap described in Section 3. Consider the Gaussian model with $p = \{80, 100\}$, $r = 5$. We vary γ in $[-1.2, -0.4]$ and $[-2.1, -1.4]$ for matrix $(K = 2)$ and tensor $(K = 3)$ clustering, respectively. Note that finding MLE under dTBM is computationally intractable. We approximate MLE using an oracle estimator; i.e., the output of Sub-algorithm 2 initialized from true assignment. Fig. 4a shows that both our algorithm and oracle estimator start to decrease around the critical value $\gamma_{\text{stat}} = \gamma_{\text{comp}} = -1$ in matrix case. In contrast, Fig. 4b shows a significant gap in the phase transitions between the algorithm estimator and oracle estimator in tensor case. The oracle error rapidly decreases to 0 when $\gamma_{\text{stat}} = -2$, whereas the algorithm estimator tends to achieve exact clustering when $\gamma_{\text{comp}} = -1.5$. Fig. 4 confirms the existence of the statistical-computational gap in our Theorems 2 and 3.

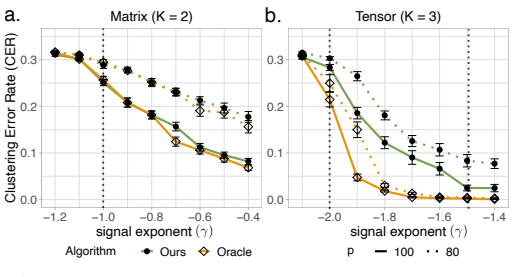

Figure 4: SNR phase transitions for clustering in dTBM with $p = \{80, 100\}, r = 5$ under (a) matrix case with $\gamma \in [-1.2, -0.4]$ and (b) tensor case with $\gamma \in [-2.1, -1.4]$.

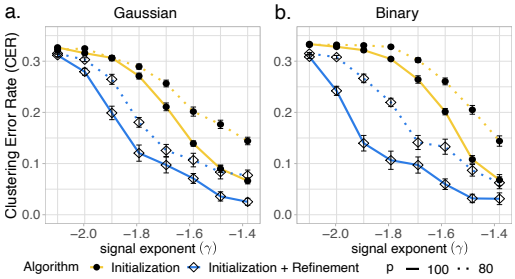

Figure 5: CER versus signal exponent (γ) for initialization only and for combined algorithm. We set $p = \{80, 100\}, r = 5, \gamma \in [-2.1, -1.4]$ under (a) Gaussian and (b) Bernoulli models.

The second experiment verifies the performance guarantees of two algorithms: (i) weighted higher-order initialization; (ii) combined algorithm of weighted higher-order initialization and angle-based iteration. We consider both the Gaussian and Bernoulli models with $p = \{80, 100\}$, $r = 5$, $\gamma \in [-2.1, -1.4]$. Fig. 5 shows the substantial improvement of combined algorithm over initialization, especially under weak and intermediate signals. This phenomenon agrees with the error rates in Theorems 4 and 5 and confirms the necessity of the local iterations.

5.2 Comparison with other methods

We compare our algorithm with following higher-order clustering methods:

- **HOSVD** : HOSVD on data tensor and k-means on the rows of the factor matrix;
- **HOSVD+**: HOSVD on data tensor and k-means on the ℓ_2-normalized rows of the factor matrix;
- **HLloyd**: High-order Lloyd algorithm and high-order spectral clustering [13];
- **SCORE**: Tensor-SCORE for clustering [15];

Among the four alternative algorithms, the **SCORE** is the closest method to ours. We set the tuning parameters of **SCORE** as in previous literature [15]. The methods **SCORE** and **HOSVD+** are designed for degree models, whereas **HOSVD** and **HLloyd** are designed for non-degree models. We conduct two experiments to assess impacts of (i) signal strength and (ii) degree heterogeneity, based on Gaussian and Bernoulli models with $p = 100, r = 5$. Our algorithm is referred to as **dTBM** in the comparison.

We investigate the effects of signal to clustering performance by varying $\gamma \in [-1.5, -1.1]$. Fig. 6 shows the consistent outperformance of our method **dTBM** among all algorithms. The sub-optimality of **SCORE** and **HOSVD+** indicates the necessity of local iterations on the clustering. Furthermore, Fig. 6 shows the inadequacy of non-degree algorithms in the presence of mild degree heterogeneity. The only exception is the slightly better performance of **HLloyd** over **HOSVD+** under Gaussian model. However, we find the advantage of **HLloyd** disappears with higher degree heterogeneity (see extra simulation results in Supplement). The experiment demonstrates the benefits of addressing heterogeneity in higher-order clustering tasks.

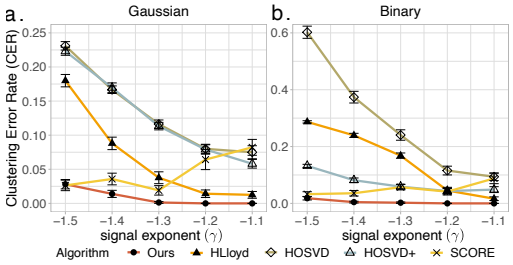

Figure 6: CER versus signal exponent (γ) for different methods. We set $p = 100, r = 5, \gamma \in [-1.5, -1.1]$ under (a) Gaussian and (b) Bernoulli models.

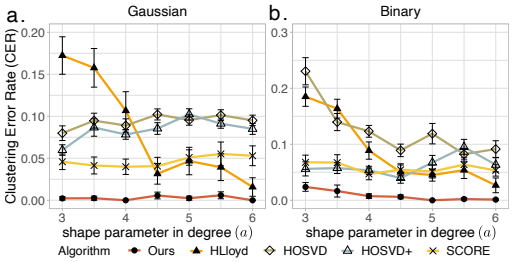

Figure 7: CER versus shape parameter in degree (a) for different methods. We set $p = 100, r = 5, \gamma = -1.2$ under (a) Gaussian and (b) Bernoulli models.

The last experiment investigates the effects of degree heterogeneity to clustering performance. We fix the signal exponent $\gamma = -1.2$ and vary the extent of degree heterogeneity. In this experiment, we generate $\boldsymbol{\theta}$ from Pareto distribution prior to normalization. The density function of Pareto distribution is $f(x|a,b) = ab^a x^{-(a+1)} \mathbb{1}\{x \geq b\}$, where a is called *shape* parameter. We vary $a \in [3, 6]$ and choose b such that $\mathbb{E}[X] = a(a-1)^{-1}b = 1$ for X following Pareto(a, b). Note that a smaller a leads to a larger variance in $\boldsymbol{\theta}$ and hence a larger degree heterogeneity. Fig. 7 demonstrates the stability of degree-corrected algorithms (**dTBM**, **SCORE**, **HOSVD+**) over the entire range of degree heterogeneity under consideration. In contrast, non-degree algorithms (**HLloyd**, **HOSVD**) show poor performance with large heterogeneity, especially in Bernoulli cases. This experiment, again, highlights the benefit of addressing degree heterogeneity in higher-order clustering.

5.3 Peru Legislation data analysis

We consider the legislation networks in the Congress of the Republic of Peru [19]. Because of the frequent political power shifts in the Peruvian Congress during 2006-2011, we choose to focus on the data for the first half of 2006-2007 year. The dataset records the co-sponsorship of 116 legislators from top 5 parties and 802 bill proposals. We reconstruct legislation network as an order-3 binary tensor $\mathcal{Y} \in \{0,1\}^{116\times116\times116}$, where $\mathcal{Y}_{ijk} = 1$ if the legislators (i, j, k) have sponsored the same bill, and $\mathcal{Y}_{ijk} = 0$ otherwise. The true party affiliations of legislators are provided and serve as ground truth. We apply various higher-order clustering methods to $\mathcal{Y}$ with $r = 5$. Tab. 2 shows that our **dTBM** achieves the best performance compared to others. The second best method is the two-stage algorithm **HLloyd**, followed by the spectral methods **SCORE** and **HOSVD+**. This result is consistent with our simulations under strong signal and moderate degree heterogeneity. The comparison suggests that our method **dTBM** is more appealing in real-world applications.

Method	**dTBM**	**HOSVD**	**HOSVD+**	**HLloyd**	**SCORE**
CER	**0.116**	0.22	0.213	0.149	0.199

Table 2: Clustering errors (measured by CER) for various methods with Peru Legislation dataset.

6 Conclusion

We have developed a general degree-corrected tensor block model with a two-step angle-based polynomial-times algorithm. We have, for the first time, characterized the statistical and computational behaviors of the degree-corrected tensor block model under different signal-to-noise ratio regimes. Simulations and real data analysis confirm the potential of our method for practical applications.

References

[1] Abbe, E. (2017). Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531.

[2] Ahn, K., Lee, K., and Suh, C. (2018). Hypergraph spectral clustering in the weighted stochastic block model. *IEEE Journal of Selected Topics in Signal Processing*, 12(5):959–974.

[3] Ahn, K., Lee, K., and Suh, C. (2019). Community recovery in hypergraphs. *IEEE Transactions on Information Theory*, 65(10):6561–6579.

[4] Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M., and Telgarsky, M. (2014). Tensor decompositions for learning latent variable models. *The Journal of Machine Learning Research*, 15(1):2773–2832.

[5] Bickel, P. J. and Chen, A. (2009). A nonparametric view of network models and newman–girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106(50):21068–21073.

[6] Brennan, M. and Bresler, G. (2020). Reducibility and statistical-computational gaps from secret leakage. In *Conference on Learning Theory*, pages 648–847. PMLR.

[7] Chi, E. C., Gaines, B. J., Sun, W. W., Zhou, H., and Yang, J. (2020). Provable convex co-clustering of tensors. *Journal of Machine Learning Research*, 21(214):1–58.

[8] Chi, Y., Lu, Y. M., and Chen, Y. (2019). Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67(20):5239–5269.

[9] Chien, I., Lin, C.-Y., and Wang, I.-H. (2018). Community detection in hypergraphs: Optimal statistical limit and efficient algorithms. In *International Conference on Artificial Intelligence and Statistics*, pages 871–879. PMLR.

[10] Gao, C., Ma, Z., Zhang, A. Y., and Zhou, H. H. (2018). Community detection in degree-corrected block models. *The Annals of Statistics*, 46(5):2153–2185.

[11] Ghoshdastidar, D. and Dukkipati, A. (2017a). Consistency of spectral hypergraph partitioning under planted partition model. *The Annals of Statistics*, 45(1):289–315.

[12] Ghoshdastidar, D. and Dukkipati, A. (2017b). Uniform hypergraph partitioning: Provable tensor methods and sampling techniques. *The Journal of Machine Learning Research*, 18(1):1638–1678.

[13] Han, R., Luo, Y., Wang, M., and Zhang, A. R. (2020). Exact clustering in tensor block model: Statistical optimality and computational limit. *arXiv preprint arXiv:2012.09996*.

[14] Hore, V., Viñuela, A., Buil, A., Knight, J., McCarthy, M. I., Small, K., and Marchini, J. (2016). Tensor decomposition for multiple-tissue gene expression experiments. *Nature genetics*, 48(9):1094.

[15] Ke, Z. T., Shi, F., and Xia, D. (2019). Community detection for hypergraph networks via regularized tensor power iteration. *arXiv preprint arXiv:1909.06503*.

[16] Kim, C., Bandeira, A. S., and Goemans, M. X. (2018). Stochastic block model for hypergraphs: Statistical limits and a semidefinite programming approach. *arXiv preprint arXiv:1807.02884*.

[17] Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review*, 51(3):455–500.

[18] Koniusz, P. and Cherian, A. (2016). Sparse coding for third-order super-symmetric tensor descriptors with application to texture recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5395–5403.

[19] Lee, S. H., Magallanes, J. M., and Porter, M. A. (2017). Time-dependent community structure in legislation cosponsorship networks in the congress of the republic of peru. *Journal of Complex Networks*, 5(1):127–144.

[20] Meilă, M. (2012). Local equivalences of distances between clusterings—a geometric perspective. *Machine Learning*, 86(3):369–389.

[21] Pananjady, A. and Samworth, R. J. (2020). Isotonic regression with unknown permutations: Statistics, computation, and adaptation. *arXiv preprint arXiv:2009.02609*.

[22] Wang, L., Durante, D., Jung, R. E., and Dunson, D. B. (2017). Bayesian network–response regression. *Bioinformatics*, 33(12):1859–1866.

[23] Wang, M., Fischer, J., and Song, Y. S. (2019). Three-way clustering of multi-tissue multi-individual gene expression data using semi-nonnegative tensor decomposition. *The Annals of Applied Statistics*, 13(2):1103–1127.

[24] Wang, M. and Li, L. (2020). Learning from binary multiway data: Probabilistic tensor decomposition and its statistical optimality. *Journal of Machine Learning Research*, 21(154):1–38.

[25] Wang, M. and Zeng, Y. (2019). Multiway clustering via tensor block models. In *Advances in Neural Information Processing Systems*, pages 713–723.

[26] Young, J.-G., St-Onge, G., Desrosiers, P., and Dubé, L. J. (2018). Universality of the stochastic block model. *Physical Review E*, 98(3):032309.

[27] Yun, S.-Y. and Proutiere, A. (2016). Optimal cluster recovery in the labeled stochastic block model. *Advances in Neural Information Processing Systems*, 29:965–973.

[28] Zhang, A. and Xia, D. (2018). Tensor SVD: Statistical and computational limits. *IEEE Transactions on Information Theory*, 64(11):7311–7338.

[29] Zhang, Q. and Tan, V. Y. (2021). Exact recovery in the general hypergraph stochastic block model. *arXiv preprint arXiv:2105.04770*.