Multiway Spherical Clustering via Degree-Corrected Tensor Block Models

Jiaxin Hu
University of Wisconsin – Madison

Miaoyan Wang
University of Wisconsin – Madison

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~~INTRODUCTION

Multiway arrays have been widely collected in various fields including social networks (Anandkumar et al., 2014), neuroscience (Wang et al., 2017), and computer science (Koniusz and Cherian, 2016). 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 (Wang et al., 2019; Hore et al., 2016), where the data tensor consists of expression measurements indexed by (gene, individual, tissue) triplets. Another example is *hypergraph* network ~~Ghoshdastidar and Dukkipati (2017); Ghoshdastidar et al. (201~~ (Ghoshdastidar and Dukkipati, 2017; Ghoshdastidar et al., 2017; Ah 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~~ We study 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.~~Figure 1 illustrates the noisy tensor and the underlying checkerboard structures discovered by multiway clustering methods. ~~The checkerboard structure serves as a meta tool to many popular structures including the low-rankness (Young et al., 2018) , latent space models (Wang and Li, 2020), and isotonic models (Pananjady and Samworth, 2020).~~ 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) ~~Chi et al. (2020); Wang and Zeng (2019)~~ (Wang and Zeng, 2019), which extends the usual matrix stochastic block model (Abbe, 2017) to tensors. ~~The matrix~~ Matrix analysis tools, however, are sub-optimal for higher-order clustering. Developing tensor tools for solving block models has received increased interest recently ~~(Han et al., 2020; Wang and Zeng, 2019)~~ (Wang and Zeng, 2019; Chi et al., 2020; Han et al., 2020) .

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

~~TBM~~ Classical tensor block model 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~~ individual-specific parameters apart from the ~~block effects~~community-specific parameters. However, the exchangeability assumption is often non-realistic. Each node may contribute to the data variation by its own multiplicative effect. We call the unequal node-specific effects the *degree heterogeneity*. Such degree heterogeneity appears commonly in social networks. Ignoring the degree heterogeneity may seriously mislead the clustering results. For example, regular block model fails to model the member affiliation in Karate Club network~~Bickel and Chen (2009) without addressing~~ (Bickel and Chen, 2009) without addressing degree heterogeneity.

~~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~~ The *degree-corrected tensor block model* (dTBM) has been proposed recently to account for the degree heterogeneity (Ke et al., 2019). The dTBM combines a higher-order checkerboard structure with degree parameter $\boldsymbol{\theta} = (\theta(1), \ldots, \theta(p))^T$ $\boldsymbol{\theta} = (\boldsymbol{\theta}(1), \ldots, \boldsymbol{\theta}(p))^T$ to allow heterogeneity among p nodes. ~~Fig.~~Figure 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. To solve dTBM, we project clustering objects to a unit sphere and perform iterative clustering based on angle similarity. We refer to the algorithm as the *spherical* clustering; detailed procedures are in Section 4. The spherical clustering avoids the estimation of nuisance degree heterogeneity. The usage of angle similarity brings new challenges to the theoretical results, and we develop new polar-coordinate based techniques in the proofs.

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~~separability.

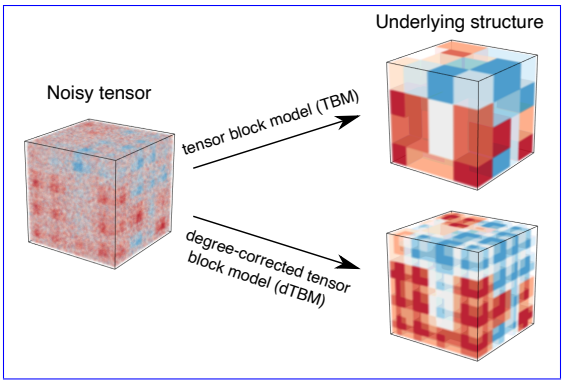

Figure 1: Examples for order-3 TBM with and without degree correction. Both TBM and dTBM have four communities on each mode, while dTBM allows a richer structure with degree heterogeneity.

• We 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~~ among (vector) one-dimensional clustering, (matrix) biclustering, and (tensor) higher-order clustering. Specific SNR thresholds and algorithm ~~behaviours~~ behaviors are depicted in ~~Fig.~~Figure 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.

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~~ Our work is closely related to but also distinct from several lines of existing research. Table 1 summarizes the most relevant models.

Block model. Block models such as stochastic block model (SBM) and degree-corrected ~~model from matrices to tensors. There is a huge literature on degree-corrected matrix models; see~~ SBM have been widely used for matrix clustering problems. See the review paper ~~Abbe (2017)~~ (Abbe, 2017) and the references therein. ~~The tensor counterparts, however, are relatively less understood. Tab. 1 summarizes the most relevant models for~~ The (non-degree) tensor block model (TBM) is a higher-order ~~clustering. Earlier tensor~~

Gao et al Gao et al. (2018)Han et al Han et al. (2020)Ke et al Ke et al. (2019)Ghoshdastidar et al Ghoshdastidar et al. (2017)**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)Comparison between previous methods with our method.

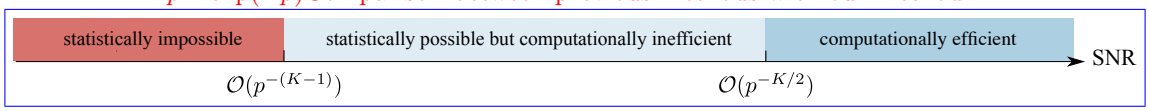

Figure 2: SNR thresholds for statistical and computational limits in order-K dTBM with dimension $(p, ..., p)$ and $K \geq 2$. The SNR gap between statistical possibility and computational efficiency exists only for tensors with $K \geq 3$.

methods either fail to allow degree heterogeneity , or suffer from sub-optimal misclustering rates Ke et al. (2019); Ghoshdastidar et al. (2017); Chi et al. (2020) . In contrast, our method addresses the degree heterogeneity, allows discrete and continuous entries, and achieves exponentially fast rate in clustering tasks. extension of SBM, and its statistical-computational properties are investigated in recent literatures (Wang and Zeng, 2019; Han et al., 2020; Ghoshdastidar et al., 2017) . Extending results from non-degree to degree-corrected model is highly challenging. Our dTBM parameter space is equipped with angle-based similarity and nuisance degree parameters. The extra complexity makes the Cartesian coordinates based analysis (Han et al., 2020) non-applicable to our setting. Towards this goal, we have developed a new polar coordinates based analysis to control the model complexity. We also develop a new angle-based iteration algorithm to achieve optimal clustering rates *without the need of estimating nuisance degree parameters.*

From the algorithm perspective, our *Degree-corrected block model.* The hypergraph degree-corrected block model (hDCBM) and its variant have been proposed in the literature (Ke et al., 2019; Yuan et al., 2018). For this popular model, however, the optimal statistical-computational rates remain an open problem. Our main contribution is to provide a sharp statistical and computational critical phase transition in dTBM literature. In addition, our algorithm results in a faster *exponential* error rate, in contrast to the *polynomial* rate in Ke et al. (2019) . The original hDCBM (Ke et al., 2019) is designed for binary observations only, and we extend the model to both continuous and binary observations. We believe our results are novel and helpful to the community. See Figure 2 for overview of our results.

Global-to-local algorithm strategy. Our methods generalize the recent global-to-local strategy for matrix learning (Gao et al., 2018; Chi et al., 2019; Yun and Proutiere, 2016) to tensors (Han et al., 2020; Ahn et al., 2018; Kim et al., 2018). 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 Kolda and Bader (2009) (De Lathauwer et al., 2000), and we develop a weighted higher-order initialization that relaxes the eigen-gap singular-value 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 *provably* achieves the exact clustering within only polynomial-time complexity.

Notation. We use lower-case letters (e.g., $a, \theta a, b$) 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 $\mathbf{1}_p$ to denote a vector of length p with all entries to be 1. 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 length-p length vector $\boldsymbol{a} = (a_1, \ldots, a_p)$vector $\boldsymbol{a}$, we use $a(i) \in \mathbb{R}$ to denote the i-th entry of $\boldsymbol{a}$, and use $\boldsymbol{a}_I$ to denote the sub-vector by restricting the indices in the set $I \subset [p]$. We use $\|\boldsymbol{a}\| = (\sum_i a_i^2)^{1/2}$ $\|\boldsymbol{a}\| = \sqrt{\sum_i a^2(i)}$ to denote the ℓ_2-norm, $\|\boldsymbol{a}\|_1 = \sum_i |a_i|$ to denote the ℓ_1 norm of $\boldsymbol{a}$. For a matrix $\boldsymbol{Y}$, we use $\boldsymbol{Y}_{i\cdot}$ to denote the i-th row of the matrix. We let $\mathcal{Y} = [\![\mathcal{Y}(i_1, \ldots, i_K)]\!] \in \mathbb{R}^{p_1 \times \cdots \times p_K}$ 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 $\cos(\boldsymbol{a}, \boldsymbol{b}) = \cos(\boldsymbol{a}^T, \boldsymbol{b}^T)$. Let $\mathcal{Y} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$ be an order-K $(p_1, ..., p_K)$-dimensional tensor. We use $\mathcal{Y}(i_1, \ldots, i_K)$ to denote the $(i_1, \ldots, i_K)$-th entry

	Gao et al. (2018)	Han et al. (2020)	Ghoshdastidar et al. (2017)	Ke et al. (2019)	**Ours**
Allow tensors of arbitrary order	×	√	√	√	√
Allow degree heterogeneity	√	×	√	√	√
Singular-value gap-free clustering	√	√	×	×	√
Misclustering rate (for order K^*)	-	$\exp(-p^{K/2})$	p^{-1}	p^{-2}	$\exp(-p^{K/2})$

Table 1: Comparison between previous methods with our method. *We list the result for order-K tensors with $K \geq 3$ and general number of communities $r = \mathcal{O}(1)$.

of $\mathcal{Y}$. 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

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

~~$\mathcal{X}(i_1, \ldots, i_K) = \sum_{(j_1, \ldots, j_K)} \mathcal{S}(j_1, \ldots, j_K) \boldsymbol{M}_1(i_1, j_1) \cdots \boldsymbol{M}_K(i_K, j_K)$.~~ For a matrix $\boldsymbol{Y}$, we use $\boldsymbol{Y}_{i\cdot}$ (respectively, $\boldsymbol{Y}_{\cdot i}$) to denote the i-th row (respectively, i-th column) of the matrix. Similarly, for an order-3 tensor, we use $\mathcal{Y}_{::i}$ to denote the i-th matrix slide of the tensor. 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 $a_p \lesssim b_p$ or $a_p = \mathcal{O}(b_p)$ if $\lim_{p \to \infty} a_p / b_p \leq c$ for some constant $c \geq 0$, $a_p = o(b_p)$ if ~~$\lim_{p \to \infty} a_p / b_p \leq c$~~ $\lim_{p \to \infty} a_p / b_p = 0$, and $a_p = \Omega(b_p)$ if ~~$c b_p \leq a_p \leq C b_p$, for some constants $c, C \geq 0$~~both $b_p \lesssim a_p$ and $a_p \lesssim b_p$. Throughout the paper, we use the terms "community" and "clusters" exchangeably.

2 ~~Model formulation~~MODEL FORMULATION

2.1 ~~Degree-corrected tensor block model~~dTBM

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 ~~Ghoshdastidar et al. (2017); Ke et al. (2019)~~ ~~,~~ (Ke et al., 2019),

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

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~~zero-mean 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:

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

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 ~~$\mathcal{X}$~~ $\mathcal{Y}$ an r-block tensor with degree $\boldsymbol{\theta}$ ~~if $\mathcal{X}$ admitsdTBM~~if $\mathcal{Y}$ admits (1). We are particularly interested in high-dimensional regime where p grows whereas $r = \mathcal{O}(1)$. The extension to general asymmetrical dTBM is obtained via replacing $(\boldsymbol{M}, \boldsymbol{\Theta})$ in (1) by mode-specific parameters $(\boldsymbol{M}_k, \boldsymbol{\Theta}_k)$ for every mode $k \in [K]$. 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 (Wang and Zeng, 2019; Han et al., 2020), which is widely used in previous clustering algorithms ~~Wang and Zeng (2019); Chi et al. (2020). The theoretical results in TBM serve as benchmarks for dTBM.~~ (Wang and Zeng, 2019; Chi et al., 2020)

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 $\sigma^2 = 1/4$.

~~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~~Angle Gap

The goal of clustering is to estimate the partition function z from model (1). For ease of notation, we focus on symmetric tensors; the extension to non-symmetric tensors are similar. We use $\mathcal{P}$ to denote the following parameter space for $(z, \mathcal{S}, \boldsymbol{\theta})$,

$$\mathcal{P} = \Big\{(z, \mathcal{S}, \boldsymbol{\theta}) : \boldsymbol{\theta} \in \mathbb{R}^p_+, \text{ for } a \in [r], \frac{c_1 p}{r} \le |z^{-1}(a)| \le \frac{c_2 p}{r}, \\ c_3 \le \|\mathrm{Mat}(\mathcal{S})_{a:}\| \le c_4, \|\boldsymbol{\theta}_{z^{-1}(a)}\|_1 = |z^{-1}(a)|, a \in [r]\Big\}, \tag{2}$$

where $c_i > 0$'s are universal constants. We briefly describe the rationale of the constraints in (2). 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 allows the trigonometric cos to describe the angle similarity in the Assumption 1 below and Sub-algorithm 2 in Section 4. Note that the positivity constraint can be achieved without sacrificing model flexibility, by using a slightly larger dimension of $\mathcal{S}$ in the factorization (1)~~(see Supplement)~~; see Supplement **??**. Second, ~~the~~ recall that the quantity $|z^{-1}(a)|$ denotes the number of nodes in a-th community. 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 $\mathrm{Mat}(\mathcal{S})_{a:}$ requires no purely zero slide in $\mathcal{S}$, so the core tensor $\mathcal{S}$ is not trivially reduced to a 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. See Supplement **??** for the parameter space comparison with previous works.

We ~~first provide the identifiablity~~ now provide the identifiability 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 ~~$\mathbb{E}\mathcal{Y}$~~ $\mathbb{E}\mathcal{Y}$ in (1) is bounded by, but not necessarily equal to, the number of blocks r (Wang and Zeng, 2019). 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 $\boldsymbol{S} = \mathrm{Mat}(\mathcal{S})$. Assume the minimal gap between normalized rows of $\boldsymbol{S}$ is bounded away from zero~~,~~; i.e., for $r \ge 2$,

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

~~Equivalently,~~

We make the convention $\Delta_{\min} = 1$ for $r = 1$. Equivalently, (3) says that none of the two rows in $\boldsymbol{S}$ are parallel~~,~~; i.e., ~~$\max_{a\neq b\in[r]} \cos(\boldsymbol{S}_{a:}, \boldsymbol{S}_{b:}) = 1 - \Delta^2_{\min}/2 < 1$.~~ $\max_{a\neq b\in[r]} \cos(\boldsymbol{S}_{a:}, \boldsymbol{S}_{b:}) = 1 - \Delta^2_{\min}/2 < 1$. The quantity $\Delta_{\min}$ characterizes the non-redundancy among clusters measured by angle separation. ~~The~~ Denominators involved in definition (3) ~~is~~ are well posed because of the lower bound on $\|\boldsymbol{S}_{a:}\|$ in (2). ~~The following~~ Following theorem shows that the angle ~~gap~~ separation is sufficient and necessary for ~~parameter identifiability under~~ the identifiability of dTBM.

Theorem 1 (Model 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 $K \geq 3$~~THEORETICAL LIMITS

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

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

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

with varying ~~$\gamma > 0$~~ $\gamma \in \mathbb{R}$ 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~~ consider following parameter space,

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

The 1-block dTBM does not belong to the space $\mathcal{P}(\gamma)$ when $\gamma < 0$ by Assumption 1. Our goal is to characterize the clustering accuracy with respect to γ. Let $\hat{z}$ and z be the 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. ~~For technical simplicity, we will present our main theory with a focus on Gaussian tensors. Our algorithm and comparison extend to general sub-Gaussian models with little modification.~~

In Sections 3.1 and 3.2, we provide the lower bounds of $\ell(\hat{z}, z)$ for general Gaussian dTBMs (1) without symmetric assumptions. For general (asymmetric) Gaussian dTBMs, we assume Gaussian noise $\mathcal{E}(i_1, \ldots, i_K) \overset{\text{i.i.d.}}{\sim} N(0, \sigma^2)$, and 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 ~~limitsOur first main result is to show~~ Critical Values

The statistical limit means the minimal SNR required for solving dTBMs with *unlimited computational cost*. Our following result shows the minimax lower bound of SNR for exact recovery in dTBM. ~~We call a dTBM general if no symmetric assumption is imposed to the signal tensor.~~

Theorem 2 (Statistical lower bound)**.** Consider general Gaussian dTBMs under the parameter space $\mathcal{P}(\gamma)$ 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) ~~Wang and Zeng (2019); Ke et al. (2019)~~ (Wang and Zeng, 2019) and trace maximization ~~Ghoshdastidar and Dukkipati (2017). Our derived~~ (Ghoshdastidar and Dukkipati, 2017). As we will show in Section 4, the 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~~Critical Values

The computational limit means the minimal SNR required for exactly recovery with *polynomial-time* computational cost. An important ingredient to establish the computational limits is the *hypergraphic planted clique (HPC) conjecture* (Zhang and Xia, 2018; Brennan and Bresler, 2020). The HPC conjecture 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~~detection conjecture is provided 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 to achieve exact recovery~~ algorithms when $\gamma < -K/2$. Therefore, $\gamma_{\text{comp}} := -K/2$ is the critical value for computational performance of dTBM. In Section 4, we will show the ~~threshold $\gamma \gtrsim -K/2$ is attained by~~ 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 1 (Statistical-computational gaps)**.** Now, we have established the phase transition of exact clustering under ~~dTBM~~ order-K dTBM by combing Theorems 2 and 3. ~~Fig.~~Figure 2 summarizes our results of critical SNRs ~~. We find that the~~ when $K \geq 2$. Particularly, dTBM reduces to matrix degree-corrected model when $K = 2$, and the statistical and computational bounds show the same critical value. When $K = 1$, dTBM reduces to the degree-corrected sub-Gaussian mixture model (GMM). Lu and Zhou (2016) implies that polynomial-time algorithms are able to achieve the statistical minimax lower bound in GMM. Hence, the statistical-to-computational gap emerges only for higher-order tensors with $K \geq 3$~~. The result~~, which reveals the intrinsic distinctions ~~between~~ (among (vector) one-dimensional clustering, (matrix) biclustering, and (tensor) higher-order clustering.

~~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,~~ Besides, we find that the extra complexity from $\boldsymbol{\theta}$ does not render the estimation of z qualitatively harder; see ~~the~~ comparison of our phase transition with non-degree TBM ~~Han et al. (2020).~~ (Han et al., 2020).

4 ~~Polynomial-time algorithm under mild SNR~~ALGORITHM

~~We present a two-stage clustering algorithm~~ In this section, we present an efficient polynomial-time clustering algorithm under mild SNR. The procedure takes a global-to-local approach. See ~~Fig.~~Figure 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.

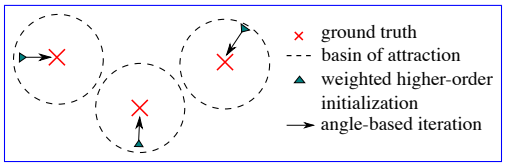

Figure 3: Illustration of our global-to-local algorithm.

4.1 ~~Weighted higher-order initialization~~ Initialization

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

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

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.~~ effects for $K \geq 3$ (Han et al., 2020). 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 ~~Ke et al. (2019)~~ (Ke et al., 2019), we apply the clustering on the unfolded tensor $\hat{\boldsymbol{X}}$ rather than on the factors $\hat{\boldsymbol{U}}$. This strategy relaxes the ~~eigen-gap separation condition Gao et al. (2018); Han et al. (2020).~~ singular-value gap condition (Gao et al., 2018; Han et al., 2020). 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

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

Sub-algorithm 1: Weighted higher-order initialization

Input: Observation $\mathcal{Y} \in \mathbb{R}^{p\times\cdots\times p}$, cluster number 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 $\mathcal{X}_{\text{pre}} = \mathcal{Y}\times_1 \boldsymbol{U}_{\text{pre}}\boldsymbol{U}_{\text{pre}}^T \times_2 \cdots \times_{K-1} \boldsymbol{U}_{\text{pre}}\boldsymbol{U}_{\text{pre}}^T$.

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 $S_0 = \{i \in [p] : \|\hat{\boldsymbol{X}}_{i:}\| = 0\}$. Set $\hat{z}(i)$ randomly in $[r]$ for $i \in S_0$.

4: For all $i \in S_0^c$, 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

$$\sum_{i\in S_0^c} \|\hat{\boldsymbol{X}}_{i:}\|^2 \|\hat{\boldsymbol{X}}^s_{i:} - \hat{\boldsymbol{x}}_{\hat{z}(i)}\|^2 \le \eta \min_{\bar{\boldsymbol{x}}_j, j\in[r], \bar{z}(i), i\in S_0^c} \sum_{i\in S^c} \|\hat{\boldsymbol{X}}_{i:}\|^2 \|\hat{\boldsymbol{X}}^s_{i:} - \bar{\boldsymbol{x}}_{\bar{z}(i)}\|^2.$$

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 $\mathcal{S}^{(t)}(a_1, ..., a_K) = \text{Ave}\{\mathcal{Y}(i_1, \ldots, i_K) : z^{(t)}(i_k) = a_k, k \in [K]\}$.

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 $z^{(T)} \in [r]^p$.

extent of heterogeneity is comparable across clusters ~~in that~~

$$\min_{a\in[r]} \|\boldsymbol{\theta}_{z^{-1}(a)}\| = (1 + o(1)) \max_{a\in[r]} \|\boldsymbol{\theta}_{z^{-1}(a)}\|. \tag{5}$$

Note that, the assumption (5) 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 the general ~~Gaussian dTBM~~ sub-Gaussian dTBM with i.i.d. noise under the parameter space $\mathcal{P}$ and Assumption 1. Assume $\boldsymbol{\theta}$ is balanced and $\min_{i\in[p]} \theta(i) \ge c$ for some constant $c > 0$. Let $z^{(0)}$ denote the output of Sub-algorithm 1. With probability going to 1, we have

$$\ell(z^{(0)}, z) \lesssim r^K p^{-K/2}/\text{SNR}. \tag{6}$$

Remark 2 (Comparison to previous results)**.** For fixed SNR, our initialization error rate with $K = 2$ agrees with the initialization error rate ~~$\mathcal{O}(p)$~~ $\mathcal{O}(p^{-1})$ in matrix models ~~Gao et al. (2018)~~(Gao et al., 2018). Furthermore, in the special case of non-degree TBMs with $\theta_1 = \cdots = \theta_p = 1$, we achieve the same initial ~~misclassification~~ misclustering error $\mathcal{O}(p^{-K/2})$ as in non-degree models (Han et al., 2020). Theorem 4 implies the advantage of our algorithm in achieving both accuracy and model flexibility.

Remark 3 (Failure of conventional tensor HOSVD)**.** If we use conventional HOSVD for tensor denoising; that is, we use $\boldsymbol{U}_{\text{pre}}$ in place of $\hat{\boldsymbol{U}}$ in line 2, then the misclustering rate becomes ~~$\mathcal{O}(p)$~~ $\mathcal{O}(p^{-1})$ for all $K \ge 2$. This rate is substantially worse than our ~~current~~ rate (6).

4.2 Angle-based ~~iteration~~Iteration

Our Theorem 4 has shown the polynomially decaying error rate from our initialization. Now we improve the error rate to exponential decay using local iterations. 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 positive degrees, and $z\colon [p] \mapsto [r]$ is the ~~clustering~~ cluster 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~~ Classical Euclidean-distance based clustering (Han et al., 2020) 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~~ See Sub-algorithm 2 ~~. The following theorem establishes~~ for full procedures.

We now establish the misclustering error rate of iterations under the stability assumption.

Definition 1 (Locally linear stability)**.** Define the ε-neighborhood of z by $\mathcal{N}(z,\varepsilon) = \{\bar{z}\colon \ell(\bar{z},z) \le \varepsilon\}$. Let $\bar{z}\colon [p] \to [r]$ be a clustering function. The degree is ε-locally linearly stable if and only if

$$\sin(\boldsymbol{p}(\bar{z}),\ \boldsymbol{p}_{\boldsymbol{\theta}}(\bar{z})) \lesssim \varepsilon\Delta_{\min}, \text{ for all } \bar{z} \in \mathcal{N}(z,\varepsilon), \quad (7)$$

where $\boldsymbol{p}(\bar{z}) = (|\bar{z}^{-1}(1)|, \ldots, |\bar{z}^{-1}(r)|)^T$ and $\boldsymbol{p}_{\boldsymbol{\theta}}(\bar{z}) = (\|\boldsymbol{\theta}_{\bar{z}^{-1}(1)}\|_1, \ldots, \|\boldsymbol{\theta}_{\bar{z}^{-1}(r)}\|_1)^T$.

Roughly speaking, the vector $\boldsymbol{p}(\bar{z})$ represents the raw cluster sizes, and $\boldsymbol{p}_\theta(\bar{z})$ represents the relative cluster sizes weighted by degrees. The local stability holds trivially for $\varepsilon = 0$ based on the construction of parameter space (2). The condition (7) controls the impact of node degree to the $\boldsymbol{p}_\theta(\cdot)$ with respect to the misclassification rate ε and angle gap.

Theorem 5 (Error for angle-based iteration)**.** Consider the setup as in Theorem 4. Suppose $r = \mathcal{O}(1)$ and ~~SNR $\gtrsim p^{-K/2}$.~~ SNR $\ge \tilde{C}p^{-K/2}\log p$ for some sufficiently large constant $\tilde{C}$. Assume the local linear stability of degree holds in the neighborhood $\mathcal{N}(z,\varepsilon)$ for all $\varepsilon \le E_0$ and some $E_0 \ge \check{C}\log^{-1} p$ with some positive constant $\check{C}$. 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

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

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,

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

for some permutation $\pi \in \Pi$. Hence, our combined algorithm is *computationally efficient* as long as SNR $\gtrsim p^{-K/2}\log p$. Note that, ignoring the logarithmic term, the minimal SNR requirement, $p^{-K/2}$, coincides with the computational lower bound in Theorem 3. Therefore, our algorithm is optimal regarding the signal requirement and lies in the sharpest *computationally efficient* regime in Figure 2.

5 ~~Numerical studies~~NUMERICAL STUDIES

We evaluate the performance of ~~the weighted higher-order initialization and angle-based iteration~~ our algorithm 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 (Meilă, 2012), and ~~a~~ lower CER indicates ~~a~~ better performance.

We generate order-3 tensors with *assortative* ~~Gao et al. (2018)~~ (Gao et al., 2018) 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 $\alpha = 1 + \Omega(p^{\gamma/2})$ with $\gamma < 0$ by Assumption 1 and definition (4). Hence, we easily adjust SNR via varying α. ~~Note that the~~ 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~~ by $\theta(i) = |X_i| + 1 - 1/\sqrt{2\pi}$ 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 of Theoretical Results

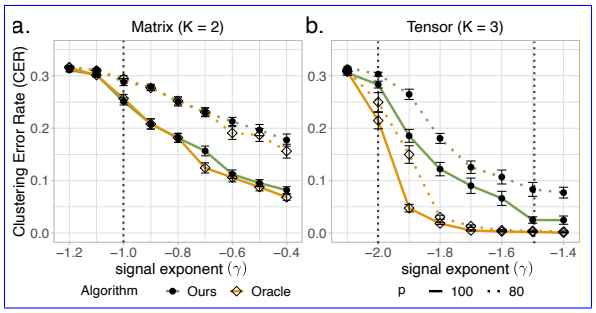

Figure 4: SNR phase transitions for clustering in dTBM with $p = \{80, 100\}, r = 5$ under (a) matrix case with $\gamma \in [-1.2, -0.4]$ and (b) tensor case with $\gamma \in [-2.1, -1.4]$.

5.2 ~~Verification for theoretical results~~

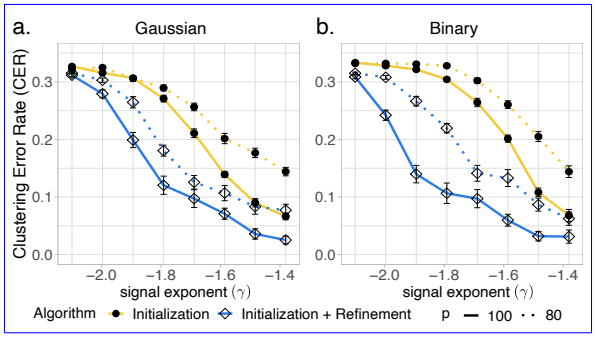

Figure 5: CER versus signal exponent (γ) for initialization only and for combined algorithm. We set $p = \{80, 100\}, r = 5, \gamma \in [-2.1, -1.4]$ under (a) Gaussian models and (b) Bernoulli models.

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.~~Figure 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.~~Figure 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.~~Figure 4 confirms the existence of the statistical-computational gap in our Theorems 2 and 3.

~~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 $\gamma \in [-2.1, -1.4]$.~~

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.~~Figure 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.

~~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 models with $c = 0.5$ and (b) Bernoulli models with $c = 0.25$.~~

5.2 Comparison with ~~other methods~~Other Methods

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

- **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** (Han et al., 2020): High-order ~~Lloyd algorithm and high-order spectral clustering (Han et al., 2020)~~clustering algorithm developed for non-degree TBM;
- **SCORE** (Ke et al., 2019): Tensor-SCORE for clustering ~~(Ke et al., 2019);~~ developed for binary tensors.

Among the four alternative algorithms, the **SCORE** is the closest method to ours. We set the tuning parameters of **SCORE** as in previous literature ~~Ke et al. (2019)~~(Ke et al., 2019). The methods **SCORE** and **HOSVD+** are designed for ~~degree models~~dTBM (1), whereas **HOSVD** and **HLloyd** are designed for non-degree models ~~. serving as~~ benchmarks. We conduct two experiments to assess the impacts of (i) signal strength and (ii) degree heterogeneity ~~, based on~~ under Gaussian and Bernoulli models with $p = 100, r = 5$. We ~~refer to~~ call our algorithm as **dTBM** in the comparison.

We investigate the effects of signal to clustering performance by varying $\gamma \in [-1.5, -1.1]$. ~~Fig.~~Figure 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.~~Figure 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)~~; see Supplement ??. The experiment demonstrates the benefits of addressing heterogeneity in higher-order clustering tasks.

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

~~CER versus shape parameter in degree (denoted $a \in [3, 6]$) for different methods. We set $p = 100, r = 5, c = 0.5, \gamma = -1.2$ under (a) Gaussian and (b) Bernoulli models.~~

The last experiment investigates the effects of degree heterogeneity to clustering performance. We use the same setting as in the first experiment in the Section 5.2, except that we fix the signal exponent $\gamma = -1.2$ and ~~vary the extent of degree heterogeneity. In this experiment, we generate~~ generate the degree heterogeneity $\boldsymbol{\theta}$ 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 the shape parameter $a \in [3, 6]$ and choose b such that ~~$\mathbb{E}[X] = a(a-1)^{-1}b = 1$~~ $\mathbb{E}X = a(a-1)^{-1}b = 1$ 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.~~Figure 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~~

5.3 Peru Legislation Data Analysis

We apply our method to the legislation networks in the Congress of the Republic of Peru (Lee et al., 2017).

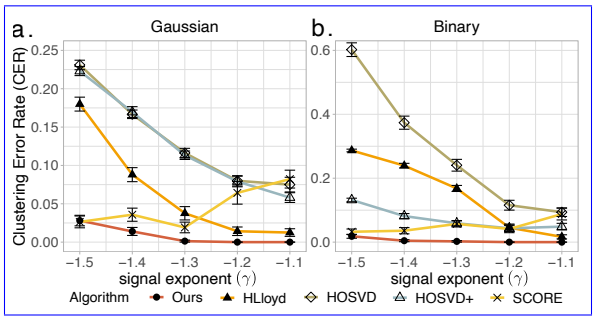

Figure 6: CER versus signal exponent (denoted γ) for different methods. We set $p = 100, r = 5, \gamma \in [-1.5, -1.1]$ under (a) Gaussian and (b) Bernoulli models.

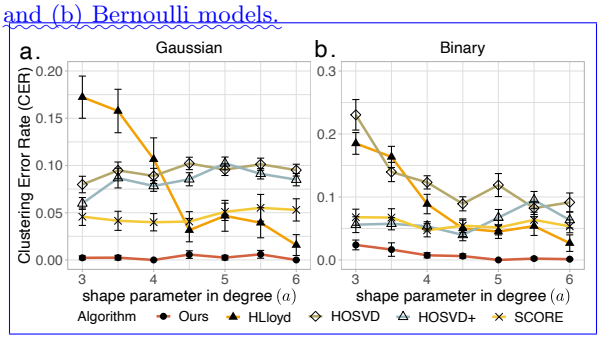

Figure 7: CER versus shape parameter in degree (denoted $a \in [3, 6]$) for different methods with $p = 100, r = 5, \gamma = -1.2$ under (a) Gaussian and (b) Bernoulli models.

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 $\mathcal{Y} \in \{0, 1\}^{116 \times 116 \times 116}$, where $\mathcal{Y}_{ijk} = 1$ if the legislators (i, j, k) have sponsored the same bill, and $\mathcal{Y}_{ijk} = 0$ otherwise. ~~The true~~ True party affiliations of legislators are provided and serve as the ground truth. We apply various higher-order clustering methods to $\mathcal{Y}$ with $r = 5$. ~~Tab.~~Table 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~~, which consists with simulations under strong signal and moderate degree heterogeneity. ~~The comparison suggests that our method **dTBM** is more appealing in real-world applications.~~

6 ~~Conclusion~~

~~We have developed a general degree-corrected tensor block model with a two-step angle-based polynomial-times algorithm. We have, for~~

~~Method dTBM HOSVD+ HLloyd SCORECER 0.116 0.213 0.149 0.199~~

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

Table 2: Clustering errors (measured by CER) for various methods in the analysis of Peru Legislation dataset.

~~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 Peru Legislation data analysis confirm the potential of our method for practical applications.~~

Acknowledgments

This research is supported in part by NSF grants DMS-1915978, DMS-2023239, EF-2133740, and funding from the Wisconsin Alumni Research foundation. We thank Zheng Tracy Ke, Rungang Han, Yuetian Luo for helpful discussions and for sharing software packages.

References

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

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.

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

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.

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.

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

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.

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.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278.

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.

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

Ghoshdastidar, D. et al. (2017). Consistency of spectral hypergraph partitioning under planted partition model. *The Annals of Statistics*, 45(1):289–315.

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*.

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.

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

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*.

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

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.

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.

Lu, Y. and Zhou, H. H. (2016). Statistical and computational guarantees of lloyd's algorithm and its variants. *arXiv preprint arXiv:1612.02099*.

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

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

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

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.

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.

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

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.

Yuan, M., Liu, R., Feng, Y., and Shang, Z. (2018). Testing community structures for hypergraphs. *arXiv preprint arXiv:1810.04617*.

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.

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