Gaussian tensor matching

Jiaxin Hu

May 20, 2022

This note summarizes the main results in Gaussian tensor matching.

1 Model

1.1 Notations

- Let the lowercase letters (e.g., a, c) denote the scalar; the bold lowercase letters (e.g, $\boldsymbol{\omega}, \boldsymbol{v}$) denote the vector; the uppercase letters with sup-script in parentheses (e.g., $P^{(m)}, Q^{(m)}$) denote the dimension-m index sets with elements in the form $(i_1, \ldots, i_m) \in \mathbb{Z}_+^m$, and drop the sup-script when $m = 1$ for simplicity; the bold uppercase letters (e.g., $\boldsymbol{P}, \boldsymbol{Q}$) denote the matrix; and the calligraphy letters (e.g., $\mathcal{A}, \mathcal{B}$) denote the tensor of order three or greater .
- For a positive integer n, let $[n]$ denote the index set $\{1, \ldots, n\}$.
- For an index set S and a positive integer m, let S^m denote the dimension-m vector space of S, where $S^m = \{(i_1, \ldots, i_m) \colon i_k \in S, \text{ for all } k \in [m]\}$.
- For two index sets S and T, we call the function $\pi \colon S \mapsto T$ *the perfect matching between S and T* if π is an one-to-one function; i.e., $\pi(i_1) = \pi(i_2)$ if and only if $i_1 = i_2$ for any $i_1, i_2 \in S$. When $T = S$, we call the π *the permutation on S*.
- For a perfect matching $\pi \colon S \mapsto T$, we call the dimension-2 index set $P^{(2)} = \{(i, \pi(i)) : i \in S\}$ *the set corresponding to* π, and we call π *the perfect matching corresponding to* $P^{(2)}$.
- For a perfect matching $\pi \colon S \mapsto T$ and an index set $S_0 \subset S$, let $\pi|_{S_0} \colon S_0 \mapsto T_0$ denote the sub-matching of π for the nodes in S_0, where $T_0 = \{\pi(i) : i \in S_0\} \subset T$.
- For a perfect matching $\pi \colon S \mapsto T$ and a dimension-m vector $\boldsymbol{v} = (v_1, \ldots, v_m) \in S^m$, let $\pi \circ \boldsymbol{v} = (\pi(v_1), \ldots, \pi(v_m)) \in T^m$ denote the permutation of the vector $\boldsymbol{v}$.
- Let $\mathcal{A} \in \mathbb{R}^{n^{\otimes m}}$ denote an order-m real tensor of dimension n on each mode. For the vector $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m) \in [n]^m$, we use $\mathcal{A}_{\boldsymbol{\omega}}$ to denote the $(\omega_1, \ldots, \omega_m)$-th entry of $\mathcal{A}$.
- We call a tensor $\mathcal{A} \in \mathbb{R}^{n^{\otimes m}}$ *super-symmetric* if for all $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m) \in [n]^m$ we have $\mathcal{A}_{\boldsymbol{\omega}} = \mathcal{A}_{\pi \circ \boldsymbol{\omega}}$ for all permutations π on the set $\{\omega_1, \ldots, \omega_m\}$.
- Let $\times_k$ denote the tensor-by-matrix multiplication on the k-th mode.
- Let $\|\cdot\|_F$ denote the Frobenius norm for tensors. Let $\langle \cdot, \cdot \rangle$ denote the inner product for tensors.

1.2 Correlated Gaussian observations

Consider two order-m random tensor observations $\mathcal{A}, \mathcal{B}' \in \mathbb{R}^{n^{\otimes m}}$ and use $\boldsymbol{\omega} \in [n]^m$ to index the entries in $\mathcal{A}$ and $\mathcal{B}$. Suppose that for all $\boldsymbol{\omega} \in [n]^m$ and some $\rho \in (0,1)$

$$\begin{pmatrix} \mathcal{A}_{\boldsymbol{\omega}} \\ \mathcal{B}'_{\boldsymbol{\omega}} \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \quad \text{and} \quad \begin{pmatrix} \mathcal{A}_{\boldsymbol{\omega}} \\ \mathcal{B}'_{\boldsymbol{\omega}} \end{pmatrix} \text{ is independent with } \begin{pmatrix} \mathcal{A}_{\boldsymbol{\omega}'} \\ \mathcal{B}'_{\boldsymbol{\omega}'} \end{pmatrix} \text{ for all } \boldsymbol{\omega} \neq \boldsymbol{\omega}'.$$

Let π^* be a permutation on $[n]$ with corresponding permutation matrix $\Pi^* \in \{0,1\}^{n \times n}$, and consider the permuted observation $\mathcal{B}$ such that for all $\boldsymbol{\omega} \in [n]^m$

$$\mathcal{B}_{\boldsymbol{\omega}} = \mathcal{B}'_{\pi^* \circ \boldsymbol{\omega}}, \quad \text{or equivalently} \quad \mathcal{B} = \mathcal{B}' \times_1 \Pi^* \times_2 \cdots \times_m \Pi^*.$$

We call the observation $(\mathcal{A}, \mathcal{B})$ follow the *order-m correlated Gaussian model (CGM)* with correlation $\rho^* \in [0,1]$ and permutation π^*, denoted as $CGM_m(n, \rho^*, \pi^*)$. Our goal is to recover π^* (or equivalently Π^*) observing $\mathcal{A}, \mathcal{B}$.

Remark 1 (Symmetry)**.** Note that $(\mathcal{A}, \mathcal{B}) \sim CGM_m(n, \rho^*, \pi^*)$ **are not** super-symmetric tensors. If we add super-symmetric constrain to $(\mathcal{A}, \mathcal{B})$, then $(\mathcal{A}, \mathcal{B})$ follows the higher-order Winger model mentioned in note 0402, and we call $(\mathcal{A}, \mathcal{B})$ follows symmetric $CGM_m(n, \rho^*, \pi^*)$. All theoretical results following are not rigorously proved under symmetric case but rigorously valid with non-symmetric case.

2 MLE and its phase transition

2.1 MLE

Theorem 1 (MLE for CGM)**.** *Suppose that the order-m tensor observation $(\mathcal{A}, \mathcal{B}) \sim CGM_m(n, \rho^*, \pi^*)$. The MLE of the true permutation π^*, denoted $\hat{\pi}$ satisfies*

$$\hat{\Pi} = \underset{\Pi \in \mathcal{P}_n}{\arg\max} \left\langle \mathcal{A} \times_1 \Pi \times_2 \cdots \times_m \Pi, \mathcal{B} \right\rangle,$$

where $\hat{\Pi}$ is the permutation matrix corresponding to $\hat{\pi}$, and $\mathcal{P}_n$ is the collection for all possible permutation matrices on $[n]$.

Proof of Theorem 1. See note 0402 for the proof of Theorem 1. The symmetry of observations does not affect the MLE result. □

2.2 MLE phase transition

Theorem 2 (MLE phase transition under CGM)**.** *Consider the observations $(\mathcal{A}, \mathcal{B}) \sim CGM_m(n, \rho, \pi^*)$. Assume n is large enough and*

$$\rho^2 \geq \frac{C_0 \log n}{n^{m-1}},$$

for some $C_0 > 0$. Then, the MLE $\hat{\pi}_{MLE}$ exactly recovers true permutation π^*; i.e., $\hat{\pi}_{MLE} = \pi^*$ with probability tends to 1.

Conversely, if we have

$$\rho^2 \le \frac{(C_0 - \varepsilon)\log n}{n^{m-1}},$$

for some positive constant $C_0 > 0$ and small constant ε. Then, the MLE $\hat{\pi}_{MLE}$ exactly recovers true permutation π^* with probability $o(1)$.

Proof of Theorem 2. See note 0517 for the rigorous proof for the positive part. The negative part remains to be the conjecture; see note 0519 for the proof idea. The constant C_0 should be carefully and precisely claimed in the future. □

3 Algorithm

3.1 Unseeded matching using empirical distribution

Consider tensor observations $\mathcal{A}, \mathcal{B} \sim CGM_m(n, \rho^*, \pi^*)$. For each pair $(i,k) \in [n]^2$, we define the L-distance with uniform partition $\{I_l\}_{l\in[L]}$ over $[-1/2, 1/2]$ as

$$d_{ik} = \sum_{l\in[L]} |F_n^i(I_l) - G_n^k(I_l)|, \tag{1}$$

where

$$F_n^i(I_l) = \frac{1}{n^{m-1}} \sum_{\boldsymbol{\omega}\in[n]^{m-1}} \mathbb{1}\{\mathcal{A}_{i,\boldsymbol{\omega}} \in I_l\}, \quad G_n^k(I_l) = \frac{1}{n^{m-1}} \sum_{\boldsymbol{\omega}\in[n]^{m-1}} \mathbb{1}\{\mathcal{B}_{k,\boldsymbol{\omega}} \in I_l\}$$

are empirical distributions of the slices in $\mathcal{A}$ and $\mathcal{B}$.

Algorithm 1 Gaussian tensor matching via empirical distribution

Input: Gaussian tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n^{\otimes m}}$, tuning parameter L.

1: Calculate the L-distance matrix $D = [\![d_{ik}]\!] \in \mathbb{R}^{n\times n}$, where d_{ik} is defined in (1).

2: Obtain the estimated permutation $\hat{\pi}$ on $[n]$ such that

$$\hat{\pi} = \underset{\pi \text{ is a permutation on [n]}}{\arg\min} \sum_{i\in[n]} d_{i\pi(i)}.$$

Output: Estimated permutation $\hat{\pi}$.

Theorem 3 (Guarantee for Algorithm 1)**.** *Assume $\sigma < \frac{c_\sigma}{\log n}$, $L = c_L \log n$ where c_σ, c_L are some positive constants. The output of Algorithm 1, $\hat{\pi}$, is equal to the true permutation π^* with probability tends to 1 as n tends to ∞.*

Proof of Theorem 1. See note 0423 for the proof of Theorem 3. □

3.2 Seeded matching

We improve the Algorithm 1 with seeded algorithm. Our seeded algorithm involves two steps: (1) generating seeds that reveals the true permutation π^* for a subset of nodes; (2) recovering the full permutation with the seed.

Specifically, we consider the seed that involves high-similarity and high-degree pairs in step (1). We measure the similarity by the L-distance defined in (1). The "degree" of node i in $\mathcal{A}$ and node k in $\mathcal{B}$ for all $i, k \in [n]$ are represented as

$$a_i = \frac{1}{n^{(m-1)/2}} \sum_{\boldsymbol{\omega} \in [n]^{m-1}} \mathcal{A}_{i,\boldsymbol{\omega}}, \quad \text{and} \quad b_k = \frac{1}{n^{(m-1)/2}} \sum_{\boldsymbol{\omega} \in [n]^{m-1}} \mathcal{B}_{k,\boldsymbol{\omega}}. \tag{2}$$

Hence, we consider the following seed set $Q^{(2)}$ with given thresholds ξ, ζ

$$Q^{(2)} = \{(i,k) \in [n]^2 \colon a_i, b_k \geq \xi, d_{ik} \leq \zeta\}. \tag{3}$$

Algorithm 2 Improved Gaussian tensor matching with seeded algorithm

Step 1: Seeds generation

Input: Gaussian tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n^{\otimes m}}$, thresholds ξ and ζ, tuning parameter L.

1: Calculate the sup-norm distances d_{ik} as (1) for all pairs $(i,k) \in [n]^2$ and the degrees a_i and b_i as (2) for all $i \in [n]$.
2: Obtain the seed set $Q^{(2)}$ as (3) with ξ and ζ.
3: **if** there exists a perfect matching $\pi_0 : S \mapsto T$ corresponding to $Q^{(2)}$ **then**
4: Output π_0.
5: **else**
6: Stop the entire Algorithm 2 immediately and output error.
7: **end if**

Output: Perfect matching π_0 or error.

Step 2: Seeded matching

Input: Gaussian tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n^{\otimes m}}$ and the perfect matching $\pi_0 : S \mapsto T$ from **Step 1**.

8: Calculate the analogy of sample covariance H_{ik} for all pairs (i,k) such that $i \in S^c$ and $k \in T^c$, where $S^c = [n]/S, T^c = [n]/T$, and

$$H_{ik} = \sum_{\boldsymbol{\omega} \in S^{m-1}} \mathcal{A}_{i,\boldsymbol{\omega}} \mathcal{B}_{k,\pi_0 \circ \boldsymbol{\omega}}.$$

9: Find the optimal perfect matching $\tilde{\pi}_1 \colon S^c \mapsto T^c$ such that

$$\tilde{\pi}_1 = \underset{\pi \colon S^c \mapsto T^c}{\arg\max} \sum_{i \in S^c} H_{i\pi(i)}.$$

10: Concatenate the matching π_0 and $\tilde{\pi}_1$ to a permutation π_1 on $[n]$ such that $\pi_1|_S = \pi_0$ and $\pi_1|_{S^c} = \tilde{\pi}_1$.

Output: Estimated permutation $\hat{\pi} = \pi_1$.

Theorem 4 (Guarantee for Algorithm 2)**.** *Let* $\sigma^2 = \sqrt{1-\rho^2}$. *Suppose* $\sigma \leq c \log^{-1/3(m-1)} n$ *for some sufficiently small positive constant* c. *Choose thresholds* $\xi \geq c_1 \sqrt{\log^{1/(m-1)} n}$ *and* $\zeta \leq c_2 \sqrt{\sigma / n^{m-1}}$ *for some positive constants* c_1, c_2. *Algorithm 2 exactly recovers the true permutation* π^* *with probability tends to 1 as* $n \to \infty$.

Proof of Theorem 2. The guarantee of **Step 1**, i.e., enough seeds are included in $Q^{(2)}$ is in note 0327 or follows the Section 2.3 in Ding et al. (2021). The guarantee of **Step 2**, i.e., $\hat{\pi}$ exactly recovers π^* with seeds in $Q^{(2)}$ is in note 0323. □

Remark 2 (No non-iterative clean up)**.** Compared with note 0402, we do not include the non-iterative clean up in Algorithm 2. Because note 0409 indicates that non-iterative clean up can not improve the noise condition in terms of the order. The output of **Step 2** achieves exact recovery under the noise condition with the same order when we do not adapt non-iterative clean up.

4 Q & A

Red color refers to my concerns.

1. **What is the main contribution of the work?**

   This work shows the MLE phase transition under the Gaussian tensor matching problem and provides an efficient matching algorithm for Gaussian tensor observations with state-of-art guarantees.

2. **The Algorithms 1 and 2 seem to be the natural higher-order extensions of Ding et al. (2021). Are there any novelties in the algorithm design and the theoretical results?**

   It is not trivial to extend the work in Ding et al. (2021).

   For unseeded algorithm, the analysis in Section 2.2 Ding et al. (2021) is not sufficient to support the exact recovery of Gaussian matrix matching. We adapt a probability density function (pdf) based method for Gaussian tensor matching, which is rigorously proved to achieve the exact recovery.

   For seeded algorithm, we show the noise condition is relaxed from $\sigma = \mathcal{O}(\log^{-1/3} n)$ to $\sigma = \mathcal{O}(\log^{-1/3(m-1)} n)$ as the order m increases, which indicates the benefit from matrix to tensor. In addition, we also show the unnecessity of non-iterative clean up proposed in Ding et al. (2021) for seeded matching, which leads to a simpler seeded algorithm.

   (In fact, our method is a natural extension of the Z-distance proposed by Ding et al. (2021) for Bernoulli matching. It is equivalent to unfold the tensor observation on one mode and use Z-distance to measure the pairwise similarity between the rows. We do not use the tensor algebra or multilinear properties to obtain the theoretical results, and Theorem 3 for unseeded algorithm does not show the improvement from matrix to tensor. Also, we do not use tensor-specific techniques for seeded algorithm and the non-iterative clean up is just a minor step.)

3. **Are there any theoretical novelties in the MLE phase transition?**

   The MLE phase transition shows the benefit from matrix to tensor observations; the sharp threshold for MLE to exact recovery is relaxed from $\rho^2 \geq \mathcal{O}(\log n/n)$ to $\rho^2 \geq \mathcal{O}(\log n/n^{m-1})$ as m increases.

   (Technically, the extension of MLE phase transition is straightforward when the observations are non-symmetric. When the observations are super-symmetric, we may need complex tensor-specific techniques to find the relationship between node and edge permutation. Because super-symmetry is hard to handle when $m \geq 3$.)

References

Ding, J., Ma, Z., Wu, Y., and Xu, J. (2021). Efficient random graph matching via degree profiles. *Probability Theory and Related Fields*, 179(1):29–115.