Gaussian Tensor Matching

Jiaxin Hu

1 Problem Formulation and Model

Consider two random tensors $\mathcal{A}, \mathcal{B}' \in \mathbb{R}^{d^{\otimes m}}$, where $\mathcal{A}(\omega)$ and $\mathcal{B}'(\omega)$ denote the tensor entry indexed by $\omega = (i_1, \ldots, i_m) \in [d]^m$. Suppose $\mathcal{A}$ and $\mathcal{B}'$ are super-symmetric; i.e., $\mathcal{A}(\omega) = \mathcal{A}(f(\omega)), \mathcal{B}(\omega) = \mathcal{B}'(f(\omega))$ for any function f permutes the indices in ω for all $\omega \in [d]^m$. Consider the bivariate generative model that for the entries $\{\omega : 1 \leq i_1 \leq \cdots \leq i_m \leq d\}$

$$(\mathcal{A}(\omega), \mathcal{B}'(\omega)) \sim \mathcal{N}\left(0, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}\right), \quad \text{and} \quad (\mathcal{A}(\omega), \mathcal{B}'(\omega)) \perp (\mathcal{A}(\omega'), \mathcal{B}'(\omega')), \text{ for all } \omega \neq \omega',$$

where the correlation $\rho \in (0,1)$ and $\perp$ denote the statistical independence. We call $\mathcal{A}$ and $\mathcal{B}'$ as two correlated Wigner tensors.

Suppose we observe the tensor pair $\mathcal{A}$ and $\mathcal{B} \stackrel{\text{def}}{=} \mathcal{B}' \circ \pi$, where $\pi : [d] \mapsto [d]$ denotes a permutation on $[d]$, and by definition $\mathcal{B}(i_1, \ldots, i_m) = \mathcal{B}'(\pi(i_1), \ldots, \pi(i_m))$ for all $(i_1, \ldots, i_m) \in [d]^m$.

This work aims to recover the true matching π given the noisy observations $\mathcal{A}, \mathcal{B}$.

2 Gaussian Tensor Matching

In this section, we develop the matching strategies for two correlated Wigner tensors.

2.1 Matching via Empirical Distributions

The main idea for correlated Wigner tensor matching is to use the empirical distribution of each slices and construct a distance statistics between the distributions. Specifically, for node $i \in [d]$ and tensor $\mathcal{A}$, we define the empirical distribution

$$\mu_i = \frac{1}{d^{m-1}} \sum_{(i_2, \ldots, i_m) \in [d]^{m-1}} \delta_{\mathcal{A}_{i, i_2, \ldots, i_m}},$$

where δ_x is the point mass at x. Similarly, we define

$$\nu_k = \frac{1}{d^{m-1}} \sum_{(i_2, \ldots, i_m) \in [d]^{m-1}} \delta_{\mathcal{B}_{k, i_2, \ldots, i_m}}$$

with tensor $\mathcal{B}$. Intuitively, the distance between the empirical distributions μ_i and ν_k are small if i and k forms a true pair, and the distance is large, otherwise. Also, we define the empirical CDFs as

$$F_d^i(t) = \frac{1}{d^{m-1}} \sum_{(i_2,\dots,i_m)\in[d]^{m-1}} \mathbb{1}\{\mathcal{A}_{i,i_2,\dots,i_m} \le t\}, \text{ and } G_d^k(t) = \frac{1}{d^{m-1}} \sum_{(i_2,\dots,i_m)\in[d]^{m-1}} \mathbb{1}\{\mathcal{B}_{k,i_2,\dots,i_m} \le t\}.$$

We construct the distance statistic to measure the similarity between μ_i and ν_k as

$$d_p(\mu_i,\nu_k) = \left(\int_{\mathbb{R}} dt |F_d^i(t) - G_d^k(t)|^p\right)^{1/p},$$

for $p \in [1,\infty)$. Particularly, when $p=1$, $d_p(\mu_i,\nu_k)$ is equivalent to the 1-Wasserstein distance, where

$$d_1(\mu_i,\nu_k) = \sum_{j=1}^{d^{m-1}} |\mathrm{vec}(\mathcal{A}^i)_{(j)} - \mathrm{vec}(\mathcal{B}^k)_{(j)}|, \tag{1}$$

where $\mathcal{A}^i$ denotes the i-th slice of $\mathcal{A}$, $\mathrm{vec}(\mathcal{A}^i)_{(j)}$ denotes the j-th largest entry in the i-th slice of $\mathcal{A}$, and $\mathcal{B}^k, \mathrm{vec}(\mathcal{B}^k)_{(j)}$ have similar definitions. Hence, we develop a Gaussian tensor matching algorithm with the distance statistics (1). See Algorithm 1.

Algorithm 1 Gaussian tensor matching via empirical distribution with 1-Wasserstein distance

Input: Gaussian tensors $\mathcal{A},\mathcal{B} \in \mathbb{R}^{d^{\otimes m}}$.

1: Calculate the distance statistics $d_1(\mu_i,\nu_k)$ in (1) for each pair of $(i,k)\in[d]\times[d]$.
2: Sort $\{d_1(\mu_i,\nu_k) : (i,k)\in[d]\times[d]\}$ and let S be the set of indices of the smallest d elements.
3: **if** there exists a permutation $\hat{\pi}$ such that $S = \{(i,\hat{\pi}(i)) : i\in[d]\}$ **then**
4: Output $\hat{\pi}_1$ and $\hat{\pi}_2$
5: **else**
6: Output error.
7: **end if**

Output: Estimated permutations $\hat{\pi}$ or error.

The theoretical guarantee for the success of Algorithm 1 is below.

Theorem 2.1 (Guarantee of Algorithm 1)**.** *Let $\rho = \sqrt{1-\sigma^2}$. Suppose $\sigma \le \frac{c}{\log d}$ for sufficiently small constant c. Algorithm 1 recover the true permutation π with probability tends to 1.*

2.2 Improvement with Seeded Matching

Previous strategy in Section 2.1 aims to find the matching in one shot. In this section, we improve the Algorithm 1 by seeded matching. The seeded matching includes two steps: (1) use Algorithm 1 to find the seeds with enough true pairs; (2) apply a seeded bipartite matching with the seeds.

For the first step, we consider the seeds consisting of high-degree pairs. We define the slice sum

$$a_i = \frac{1}{\sqrt{d^{m-1}}} \sum_{\omega\in[d]^{\otimes m-1}} \mathcal{A}_{i,\omega}, \quad b_k = \frac{1}{\sqrt{d^{m-1}}} \sum_{\omega\in[d]^{\otimes m-1}} \mathcal{B}_{k,\omega},$$

where a_i and b_k are considered as the counterparts of "degrees" for Gaussian tensors. By Ding et al. (2021), we have

$$\mathbb{P}(a_i \geq \xi, b_k \geq \xi) = \begin{cases} Q(\xi)^2 & \text{if } (i,k) \text{ is a fake pair} \\ Q(\xi)\exp(-C\sigma^2\xi^2) & \text{if } (i,k) \text{ is a true pair,} \end{cases}$$

where C is a positive constant, $Q(\cdot)$ is the complementary CDF of standard normal distribution, and ξ serves as the threshold for high-degree. Consider the high-degree set

$$S = \{(i,k) \in [d]^2 \colon a_i, b_k \geq \xi, d_1(\mu_i, \nu_k) \leq \zeta\}. \tag{2}$$

with given thresholds ξ and ζ. Suppose we need s seeds for bipartite matching success. We need

(1) S has enough true pairs, i.e.,

$$dQ(\xi)\exp(-C\sigma^2\xi^2) \geq s.$$

(2) no fake pairs involved in S, i.e,

$$d^2 Q(\xi)^2 \exp(-C\sigma^{-1}) = o(1),$$

where C is a positive constant and the term $\exp(-C\sigma^{-1})$ follows from the property of distance statistics $d_p(\mu_i, \nu_k)$ for the fake pair (i,k) in (3), and d^2 is the order of total pairs.

Choose the threshold $\xi = \mathcal{O}(\sqrt{s})$. We have $Q(\xi) = \Omega(s/d)\mathcal{O}(\exp \sigma^2 s)$ by (1), and with (2) we obtain

$$\sigma \leq \frac{c}{s^{1/3}},$$

for some constant c. Hence, if the number of true pairs in the seed s is smaller than $\log^3 d$, we release the condition from $\sigma \leq c/\log d$ to $\sigma \leq c/s^{1/3}$.

For the second step, the main idea for seeded matching is to use the prior information in seed to describe distance between the unseeded pairs. Let $\pi_0 : S \mapsto T$ denotes the seeds, where $S, T \subset [n]$ and $\pi_0(j) = \pi(j)$ for all $j \in S$. Define the sets

$$\mathcal{N} = \{(i_2, \ldots, i_m) : i_l \in S, \text{ for all } l = 2, \ldots, m\}$$

with $|\mathcal{N}| = |S|^{m-1}$, and define $\pi_0(\mathcal{N})$ by replacing i_l to $\pi_0(i_l)$ in the definition of $\mathcal{N}$ for all $l = 2, \ldots, m$. Then, we define the distance for the unseeded pairs (i,k) as

$$H_{p,ik} = \left(\int_{\mathbb{R}} dt \left| \frac{1}{|\mathcal{N}|} \sum_{\omega \in \mathcal{N}} \mathbb{1}\{\mathcal{A}_{i,\omega} \leq t\} - \frac{1}{|\pi_0(\mathcal{N})|} \sum_{\omega \in \pi_0(\mathcal{N})} \mathbb{1}\{\mathcal{B}_{k,\omega} \leq t\} \right|^p \right)^{1/p},$$

for some $p \geq 1$. Intuitively, the term $\frac{1}{|\mathcal{N}|}\sum_{\omega \in \mathcal{N}} \delta_{\mathcal{A}_{i,\omega}}$ describes the empirical distribution of edges in $\mathcal{A}$ related to the unseeded node i and the seeded nodes. The term $H_{i,k}$ indicates the difference between the empirical distributions related to the seeds with unseeded nodes i and k in $\mathcal{A}$ and $\mathcal{B}$, respectively.

See the improved matching strategy in Algorithm 2 with seeded matching as subroutine in Algorithm 3.

The theoretical guarantee for Algorithm 2 is below.

Algorithm 2 Gaussian tensor matching with seed improvement

Input: Gaussian tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{d^{\otimes m}}$, threshold ξ, ζ.

1: Calculate the distance statistics $d_1(\mu_i, \nu_k)$ in (1) for each pair of $(i,k) \in [d] \times [d]$.
2: Obtain the high-degree set S in (2).
3: **if** there exists a permutation π_0 such that $S = \{(i, \pi_0(i)) : i \in [d]\}$ **then**
4: Run bipartite Algorithm with seed π_0 and output $\hat{\pi}$
5: **else**
6: Output error.
7: **end if**

Output: Estimated permutations $\hat{\pi}$ or error.

Algorithm 3 Seeded Gaussian tensor matching

Input: Gaussian tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{d^{\otimes m}}$, seed $\pi_0 : S \mapsto T$.

1: For $i \in S^c$ and $k \in T^c$, obtain the distance $H_{1,ik}$.
2: Find the optimal bipartite permutation π_1' such that

$$\pi_1' = \arg\min_{\pi} \sum_{i \in S^c} H_{i\pi(i)}.$$

Let $\hat{\pi}$ denote the matching on $[d]$ such that $\hat{\pi}|_S = \pi_0$ and $\hat{\pi}|_{S^c} = \pi_1'$.
3: **if** $\hat{\pi}$ is a perfect matching on $[d]$ such that $\hat{\pi}$ is an one-to-one function from $[d]$ to $[d]$. **then**
4: Output $\hat{\pi}$.
5: **else**
6: Output error.
7: **end if**

Output: Estimated permutations $\hat{\pi}$ or error.

Theorem 2.2 (Conjecture: guarantee for Algorithm 2)**.** *Let $\rho = \sqrt{1-\sigma^2}$. Suppose $\sigma \le \frac{c}{\log^{1/3(m-1)} d}$ for sufficiently small constant c. Algorithm 1 recover the true permutation π with probability tends to 1.*

Remark 1 (From matrix matching to tensor matching)**.** The improvement of tensor matching with increasing order m is mainly indicated in the seeded algorithm. Intuitively, in tensor cases, we need less seeds to obtain the description of the unseeded pairs with the same accuracy, which results in a looser upper bound of σ. Note that a larger σ indicates a smaller correlation between two tensors and thereof a weaker "signal" in the matching problem. Therefore, we allow a weaker signal assumption $\sigma = \mathcal{O}(\frac{1}{\log^{1/3(m-1)} d})$ as m increases.

3 Proof Sketches

Proof Sketch of Theorem 2.1. Without loss of generality, we assume the true permutation π is the identity mapping; i.e., $\pi(i) = i$ for all $i \in [d]$. For simplicity, let d_{ik} denote the distance statistics $d_1(\mu_i, \nu_j)$ in (1). To guarantee the Algorithm 1 outputs the true permutation with probability , it suffices to show

$$\min_{i \neq k \in [d]} d_{ik} > \max_{i \in [d]} d_{ii}$$

with probability tends to 1.

According to Ding et al. (2021), for all $i \in [d]$ we have

$$\mathbb{P}\left(d_{ii} \geq \sqrt{\frac{\sigma}{d^{m-1}}}\right) \leq \exp\left(-\frac{C_1}{\sigma}\right), \text{(needs to be verified)}$$

and for all $i \neq k \in [d]$

$$\mathbb{P}\left(d_{ik} \leq \sqrt{\frac{\sigma}{d^{m-1}}}\right) \leq \exp\left(-\frac{C_2}{\sigma}\right). \tag{3}$$

Hence, we have

$$\mathbb{P}\left(\max_{i\in[d]} d_{ii} < \sqrt{\frac{\sigma}{d^{m-1}}}\right) \geq [1 - \exp(-C_1/\sigma)]^d,$$

and

$$\mathbb{P}\left(\min_{i\neq k\in[d]} d_{ik} > \sqrt{\frac{\sigma}{d^{m-1}}}\right) \geq [1 - d\exp(-C_2/\sigma)]^d,$$

where C_1, C_2 are two positive constants. Take $\sigma \leq \frac{c}{\log d}$ for sufficiently small c. Then, we have

$$\min_{i\neq k\in[d]} d_{ik} > \sqrt{\frac{\sigma}{d^{m-1}}} > \max_{i\in[d]} d_{ii},$$

with probability $\mathcal{O}([1 - 1/d^2]^d)$ that tends to 1 as $d \to \infty$.

$\square$

4 Numerical Experiments

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.