# Breaking the $1/\sqrt{n}$ Barrier: Faster Rates for Permutation-based Models in Polynomial Time

**Cheng Mao** MAOCHENG@MIT.EDU
*Massachusetts Institute of Technology, Cambridge, MA, USA*

**Ashwin Pananjady** ASHWINPM@BERKELEY.EDU
*University of California, Berkeley, CA, USA*

**Martin J. Wainwright** WAINWRIG@BERKELEY.EDU
*University of California, Berkeley, CA, USA*

**Editors:** Sébastien Bubeck, Vianney Perchet and Philippe Rigollet

## Abstract

Many applications, including rank aggregation and crowd-labeling, can be modeled in terms of a bivariate isotonic matrix with unknown permutations acting on its rows and columns. We consider the problem of estimating such a matrix based on noisy observations of a subset of its entries, and design and analyze a polynomial-time algorithm that improves upon the state of the art. In particular, our results imply that any such $n \times n$ matrix can be estimated efficiently in the normalized Frobenius norm at rate $\widetilde{\mathcal{O}}(n^{-3/4})$, thus narrowing the gap between $\widetilde{\mathcal{O}}(n^{-1})$ and $\widetilde{\mathcal{O}}(n^{-1/2})$, which were hitherto the rates of the most statistically and computationally efficient methods, respectively.

**Keywords:** permutation-based models, ranking, pairwise comparisons, crowd-labeling, statistical-computational gap, shape-constrained estimation.

## 1. Introduction

Structured matrices with entries in the range $[0, 1]$ and unknown permutations acting on their rows and columns arise in multiple applications, including estimation from pairwise comparisons and crowd-labeling. Traditional parametric models (Bradley and Terry, 1952; Luce, 1959; Thurstone, 1927; Dawid and Skene, 1979) assume that these matrices are obtained from rank-one matrices via a known link function. With the goal of increasing model flexibility, a recent line of work has studied the class of *permutation-based* models (Chatterjee, 2015; Shah et al., 2017, 2016a). This class of models imposes only shape constraints on the matrix, such as monotonicity, before unknown permutations act on the its rows and columns. As a result, it reduces modeling bias compared to its parametric counterparts while, perhaps surprisingly, producing models that can be estimated at rates that differ only by logarithmic factors from parametric models. On the negative side, these advantages of permutation-based models are accompanied by significant computational challenges. Except for simple models such as the noisy sorting model (Braverman and Mossel, 2008; Mao et al., 2018) where polynomial-time algorithms achieve near-optimal rates, results from many recent papers show a non-trivial statistical-computational gap in estimation rates for models with latent permutations (Shah et al., 2017; Chatterjee and Mukherjee, 2016; Shah et al., 2016a; Flammarion et al., 2016; Pananjady et al., 2017b).

. Extended abstract. Full version appears as arXiv:1802.09963, v3.

In particular, the class of matrices satisfying the *strong stochastic transitivity* condition, or SST for short, contains all $n \times n$ bivariate isotonic matrices with unknown permutations acting on their rows and columns, with an additional skew-symmetry constraint. While the minimax rate of estimating a matrix in the SST class with $\Theta(n^2)$ Bernoulli observations is $\widetilde{\Theta}(n^{-1})$ in the normalized Frobenius norm (Shah et al., 2017), the fastest computationally efficient rate is only $\widetilde{\mathcal{O}}(n^{-1/2})$, achieved by spectral methods (Chatterjee, 2015; Shah et al., 2017) and variants of the Borda count estimator (Shah et al., 2016b; Chatterjee and Mukherjee, 2016; Pananjady et al., 2017a).

Our main contribution in the current work is to tighten this statistical-computational gap. More precisely, we study the problem of estimating a bivariate isotonic matrix with unknown permutations acting on its rows and columns, given noisy, partial observations of its entries. Our polynomial-time algorithm provably achieves the rate of estimation $\widetilde{\mathcal{O}}(n^{-3/4})$ uniformly over the SST class.

## 2. Problem setup

We define $\mathbb{C}_{\mathsf{BISO}}$ to be the class of matrices in $[0,1]^{n_1 \times n_2}$ with nondecreasing rows and nondecreasing columns, where we assume $n_1 \geq n_2$ for readability. Given a matrix $M \in \mathbb{R}^{n_1 \times n_2}$ and permutations[1] $\pi \in \mathfrak{S}_{n_1}$ and $\sigma \in \mathfrak{S}_{n_2}$, we define the matrix $M(\pi, \sigma) \in \mathbb{R}^{n_1 \times n_2}$ by specifying its entries as

$$[M(\pi,\sigma)]_{i,j} = M_{\pi(i),\sigma(j)} \text{ for } i \in [n_1], j \in [n_2].$$

Also define the class $\mathbb{C}_{\mathsf{BISO}}(\pi,\sigma) := \{M(\pi,\sigma) : M \in \mathbb{C}_{\mathsf{BISO}}\}$ as the set of matrices that are bivariate isotonic when viewed along the row permutation $\pi$ and column permutation $\sigma$, respectively. The class of matrices that we are interested contains bivariate isotonic matrices whose rows and columns are acted upon by unknown, and possibly different, permutations:

$$\mathbb{C}_{\mathsf{Perm}} := \bigcup_{\substack{\pi \in \mathfrak{S}_{n_1} \\ \sigma \in \mathfrak{S}_{n_2}}} \mathbb{C}_{\mathsf{BISO}}(\pi,\sigma).$$

Letting $\mathsf{Poi}(\lambda)$ denote a Poisson random variable of mean $\lambda$, suppose that $N' = \mathsf{Poi}(N)$ noisy entries[2] are sampled independently and uniformly with replacement from all entries of $M^* \in \mathbb{C}_{\mathsf{Perm}}$. More precisely, let $E^{(i,j)}$ denote the $n_1 \times n_2$ matrix with 1 in the $(i,j)$-th entry and 0 elsewhere, and suppose that $X_\ell$ is a random matrix sampled independently and uniformly from the set $\{E^{(i,j)} : i \in [n_1],\, j \in [n_2]\}$. We observe $N'$ independent pairs $\{(X_\ell, y_\ell)\}_{\ell=1}^{N'}$ from the model

$$y_\ell = \mathsf{tr}(X_\ell^\top M^*) + z_\ell,$$

where the observations are contaminated by independent, centered, sub-Gaussian noise $z_\ell$ with variance parameter $\varsigma^2$. Now given $N'$ observations $\{(X_\ell, y_\ell)\}_{\ell=1}^{N'}$, let us define the matrix of observations $Y = Y\big(\{(X_\ell, y_\ell)\}_{\ell=1}^{N'}\big)$, with entry $(i,j)$ given by

$$Y_{i,j} = \frac{1}{p_{\mathsf{obs}}} \frac{1}{1 \vee \sum_{\ell=1}^{N'} \mathbf{1}\{X_\ell = E^{(i,j)}\}} \sum_{\ell=1}^{N'} y_\ell \, \mathbf{1}\{X_\ell = E^{(i,j)}\}. \tag{1}$$

In words, the rescaled entry $p_{\mathsf{obs}} Y_{i,j}$ is the average of all the noisy realizations of $M^*_{i,j}$ that we have observed, or zero if the entry goes unobserved.

1. We let $\mathfrak{S}_n$ represent the set of permutations on the set $[n] := \{1, 2, \ldots, n\}$ .
2. The rates obtained from such a *Poissonized* observation model are the same as those obtained without Poissonization up to constant factors, so the rates stated here also hold for the observation model with exactly $N$ noisy samples.

## 3. Algorithms and results

Our main algorithm relies on two distinct steps: first, we estimate the unknown permutations, and then project onto the class of matrices that are bivariate isotonic when viewed along the estimated permutations. The formal meta-algorithm is described below.

**Algorithm 1 (meta-algorithm)**

- Step 0: Split the observations into two disjoint parts, each containing $N'/2$ observations, and construct the matrices $Y^{(1)} = Y\left(\{X_\ell, y_\ell\}_{\ell=1}^{N'/2}\right)$ and $Y^{(2)} = Y\left(\{X_\ell, y_\ell\}_{\ell=N'/2+1}^{N'}\right)$.
- Step 1: Use $Y^{(1)}$ to obtain the permutation estimates $(\widehat{\pi}, \widehat{\sigma})$.
- Step 2: Return the matrix estimate $\widehat{M}(\widehat{\pi}, \widehat{\sigma}) := \arg\min_{M \in \mathbb{C}_{\mathsf{BISO}}(\widehat{\pi}, \widehat{\sigma})} \|Y^{(2)} - M\|_F^2$.

We now present our main permutation estimation procedure that can be plugged into Step 1 of this meta-algorithm. We first define a certain blocking sub-routine that helps us estimate the row permutation by ordering entries according to an estimate of the column permutation (and vice versa). For a partition $\mathsf{bl} = (\mathsf{bl}_1, \ldots, \mathsf{bl}_K)$ of the set $[n_2]$, we group the columns of a matrix $Y \in \mathbb{R}^{n_1 \times n_2}$ into $K$ blocks according to their indices in $\mathsf{bl}$, and refer to $\mathsf{bl}$ as a partition or *blocking* of the columns of $Y$. Given a data matrix $Y \in \mathbb{R}^{n_1 \times n_2}$, the following blocking subroutine returns a column partition $\mathsf{BL}(Y)$.

**Subroutine 1 (blocking)**

- Step 1: Compute the column sums $\{C(j)\}_{j=1}^{n_2}$ of the matrix $Y$ as

$$C(j) = \sum_{i=1}^{n_1} Y_{i,j}.$$

  Let $\widehat{\sigma}_{\mathsf{pre}}$ be the permutation along which the sequence $\{C(\widehat{\sigma}_{\mathsf{pre}}(j))\}_{j=1}^{n_2}$ is nondecreasing.

- Step 2: Set $\tau = 16(\zeta+1)\Big(\sqrt{\frac{n_1^2 n_2}{N}\log(n_1 n_2)} + \frac{n_1 n_2}{N}\log(n_1 n_2)\Big)$ and $K = \lceil n_2/\tau \rceil$. Partition the columns of $Y$ into $K$ blocks by defining

$$\begin{aligned}
\mathsf{bl}_1 &= \{j \in [n_2] : C(j) \in (-\infty, \tau)\}, \\
\mathsf{bl}_k &= \big\{j \in [n_2] : C(j) \in \big[(k-1)\tau, k\tau\big)\big\} \text{ for } 1 < k < K, \text{ and} \\
\mathsf{bl}_K &= \{j \in [n_2] : C(j) \in [(K-1)\tau, \infty)\}.
\end{aligned}$$

  Note that each block is contiguous when the columns are permuted by $\widehat{\sigma}_{\mathsf{pre}}$.

- Step 3 (aggregation): Set $\beta = n_2\sqrt{\frac{n_1}{N}\log(n_1 n_2)}$. Call a block $\mathsf{bl}_k$ "large" if $|\mathsf{bl}_k| \geq \beta$ and "small" otherwise. Aggregate small blocks in $\mathsf{bl}$ while leaving the large blocks as they are, to obtain the final partition $\mathsf{BL}$.

  More precisely, consider the matrix $Y' = Y(\mathsf{id}, \widehat{\sigma}_{\mathsf{pre}})$ having nondecreasing column sums and contiguous blocks. Call two small blocks "adjacent" if there is no other small block between them. Take unions of adjacent small blocks to ensure that the size of each resulting block is in the range $[\frac{1}{2}\beta, 2\beta]$. If the union of all small blocks is smaller than $\frac{1}{2}\beta$, aggregate them all.

  Return the resulting partition $\mathsf{BL}(Y) = \mathsf{BL}$.

Finally, we are in a position to describe our main two-dimensional sorting algorithm.

**Algorithm 2 (two-dimensional sorting)**

- Step 0: Split the observations into two independent subsamples of equal size, and form the corresponding matrices $Y^{(1)}$ and $Y^{(2)}$ according to equation (1).
- Step 1: Apply Subroutine 1 to the matrix $Y^{(1)}$ to obtain a partition $\mathsf{BL} = \mathsf{BL}(Y^{(1)})$ of the columns. Let $K$ be the number of blocks in $\mathsf{BL}$.
- Step 2: Using the second sample $Y^{(2)}$, compute the row sums

$$S(i) = \sum_{j \in [n_2]} Y^{(2)}_{i,j} \text{ for each } i \in [n_1],$$

and the partial row sums within each block

$$S_{\mathsf{BL}_k}(i) = \sum_{j \in \mathsf{BL}_k} Y^{(2)}_{i,j} \text{ for each } i \in [n_1], k \in [K].$$

Create a directed graph $G$ with vertex set $[n_1]$, where an edge $u \to v$ is present if either

$$S(v) - S(u) > 16(\zeta+1)\left(\sqrt{\frac{n_1 n_2^2}{N}\log(n_1 n_2)} + \frac{n_1 n_2}{N}\log(n_1 n_2)\right), \text{ or} \tag{2a}$$

$$S_{\mathsf{BL}_k}(v) - S_{\mathsf{BL}_k}(u) > 16(\zeta+1)\left(\sqrt{\frac{n_1 n_2}{N}|\mathsf{BL}_k|\log(n_1 n_2)} + \frac{n_1 n_2}{N}\log(n_1 n_2)\right) \text{ for some } k \in [K]. \tag{2b}$$

- Step 3: Compute a topological sort $\widehat{\pi}_{\mathsf{tds}}$ of the graph $G$; if none exists, set $\widehat{\pi}_{\mathsf{tds}} = \mathsf{id}$.
- Step 4: Repeat Steps 1–3 with $(Y^{(i)})^\top$ replacing $Y^{(i)}$ for $i = 1, 2$, the roles of $n_1$ and $n_2$ switched, and the roles of $\pi$ and $\sigma$ switched, to compute the permutation estimate $\widehat{\sigma}_{\mathsf{tds}}$.
- Step 5: Return the permutation estimates $(\widehat{\pi}_{\mathsf{tds}}, \widehat{\sigma}_{\mathsf{tds}})$.

Recall that a permutation $\pi$ is called a topological sort of $G$ if $\pi(u) < \pi(v)$ for every directed edge $u \to v$. The construction of the graph $G$ in Step 2 dominates the computational complexity, and takes time $\mathcal{O}(n_1^2 n_2/\beta) = \mathcal{O}(n_1^2 n_2^{1/2})$. We have the following guarantee for the two-dimensional sorting algorithm.

**Theorem 1** *For any matrix $M^* \in \mathbb{C}_{\mathsf{Perm}}$, we have*

$$\frac{1}{n_1 n_2}\left\|\widehat{M}(\widehat{\pi}_{\mathsf{tds}}, \widehat{\sigma}_{\mathsf{tds}}) - M^*\right\|_F^2 \lesssim (\zeta^2 \vee 1)\left[\left(\frac{n_1 \log n_1}{N}\right)^{3/4} + \frac{n_1 \log^2 n_1}{N}\right]$$

*with probability at least $1 - 9(n_1 n_2)^{-3}$.*

In particular, setting $N = n_1 n_2$, we have proved that our efficient estimator enjoys the rate

$$\frac{1}{n_1 n_2}\left\|\widehat{M}(\widehat{\pi}_{\mathsf{tds}}, \widehat{\sigma}_{\mathsf{tds}}) - M^*\right\|_F^2 = \widetilde{O}\left(n_2^{-3/4}\right),$$

which is the main theoretical guarantee established in this paper for permutation-based models.

## Acknowledgments

This work was supported in part by grants NSF CAREER DMS-1541099, NSF DMS-1541100, NSF-DMS-1612948 and DOD ONR-N00014.

## References

Ralph A. Bradley and Milton E. Terry. Rank analysis of incomplete block designs. I. The method of paired comparisons. *Biometrika*, 39:324–345, 1952.

Mark Braverman and Elchanan Mossel. Noisy sorting without resampling. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 268–276. ACM, New York, 2008.

Sabyasachi Chatterjee and Sumit Mukherjee. On estimation in tournaments and graphs under monotonicity constraints. *arXiv preprint arXiv:1603.04556*, 2016.

Sourav Chatterjee. Matrix estimation by universal singular value thresholding. *Ann. Statist.*, 43(1): 177–214, 2015.

Alexander Philip Dawid and Allan M. Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics*, pages 20–28, 1979.

Nicolas Flammarion, Cheng Mao, and Philippe Rigollet. Optimal rates of statistical seriation. *arXiv preprint arXiv:1607.02435*, 2016.

R. Duncan Luce. *Individual choice behavior: A theoretical analysis*. John Wiley & Sons, Inc., New York; Chapman & Hall, Ltd., London, 1959.

Cheng Mao, Jonathan Weed, and Philippe Rigollet. Minimax rates and efficient algorithms for noisy sorting. In Firdaus Janoos, Mehryar Mohri, and Karthik Sridharan, editors, *Proceedings of Algorithmic Learning Theory*, volume 83 of *Proceedings of Machine Learning Research*, pages 821–847. PMLR, 07–09 Apr 2018.

Ashwin Pananjady, Cheng Mao, Vidya Muthukumar, Martin J. Wainwright, and Thomas A. Courtade. Worst-case vs average-case design for estimation from fixed pairwise comparisons. *arXiv preprint arXiv:1707.06217*, 2017a.

Ashwin Pananjady, Martin J. Wainwright, and Thomas A. Courtade. Denoising linear models with permuted data. In *Information Theory (ISIT), 2017 IEEE International Symposium on*, pages 446–450. IEEE, 2017b.

Nihar B. Shah, Sivaraman Balakrishnan, and Martin J. Wainwright. A permutation-based model for crowd labeling: Optimal estimation and robustness. *arXiv preprint arXiv:1606.09632*, 2016a.

Nihar B. Shah, Sivaraman Balakrishnan, and Martin J. Wainwright. Feeling the Bern: Adaptive estimators for Bernoulli probabilities of pairwise comparisons. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pages 1153–1157. IEEE, 2016b.

Nihar B. Shah, Sivaraman Balakrishnan, Adityanand Guntuboyina, and Martin J. Wainwright. Stochastically transitive models for pairwise comparisons: statistical and computational issues. *IEEE Trans. Inform. Theory*, 63(2):934–959, 2017.

Louis L. Thurstone. A law of comparative judgment. *Psychological review*, 34(4):273, 1927.