Proceedings of Machine Learning Research vol 125:1–50, 2020 33rd Annual Conference on Learning Theory

Better Algorithms for Estimating Non-Parametric Models in Crowd-Sourcing and Rank Aggregation

Allen Liu CLIU568@MIT.EDU

Ankur Moitra MOITRA@MIT.EDU

Massachusetts Institute of Technology

Editors: Jacob Abernethy and Shivani Agarwal

Abstract

Motivated by applications in crowd-sourcing and rank aggregation, a recent line of work has studied the problem of estimating an $n \times n$ bivariate isotonic matrix with an unknown permutation acting on its rows (and possibly another unknown permutation acting on its columns) from partial and noisy observations. There are wide and persistent computational vs. statistical gaps for this problem. It is known that the minimax optimal rate is $\widetilde{O}(n^{-1})$ when error is measured in average squared Frobenius norm. However the best known polynomial time computable estimator due to Mao et al. (2018) achieves the rate $\widetilde{O}(n^{-\frac{3}{4}})$, and this is the natural barrier to approaches based on using local statistics to figure out the relative order of pairs of rows without using information from the rest of the matrix.

Here we introduce a framework for exploiting global information in shape-constrained estimation problems. In the case when only the rows are permuted, we give an algorithm that achieves error rate $O(n^{-1+o(1)})$, which essentially closes the computational vs. statistical gap for this problem. When both the rows and columns are permuted, we give an improved algorithm that achieves error rate $O(n^{-\frac{5}{6}+o(1)})$. Additionally, all of our algorithms run in nearly linear time.

Keywords: Shape-constrained estimation, bivariate isotonic matrix, non-parametric model, stochastic transitivity

1. Introduction

1.1. Background

Consider the following problem in crowd-sourcing: There are n workers and m tasks and there is an unknown matrix M where the entry $M_{i,j}$ represents the probability that worker i completes task j correctly. We are interested in the calibration problem of estimating the entries of M from partial and noisy observations. These observations could come from auditing the performance of the workers on some random tasks. The key to this problem is making some assumption about the *shape* of M. The Dawid-Skene model Dawid and Skene (1979) is a popular model in crowd-sourcing Snow et al. (2008); Raykar et al. (2010); Karger et al. (2011) that makes a particularly strong assumption about the shape of M – it assumes that each worker has some unknown but fixed probability of completing any particular task correctly. This means that all the rows of M are scalar multiples of each other, and hence it has rank one.

The problem with the Dawid-Skene model Dawid and Skene (1979) is that it precludes the very realistic possibility that some tasks are inherently more difficult than others. Recently Shah et al.

Shah et al. (2016b) proposed a challenging non-parametric[1] extension of the Dawid-Skene model to address this issue. We say that a matrix A is *bivariate isotonic* if each of its rows and columns are non-decreasing. In the context of crowd-sourcing, this type of shape constraint is natural because it translates to the assumption that there is an inherent ordering of which tasks are easier than others and which workers are more capable than others, and that the probability of completing the task correctly is a monotone increasing function of the easiness of the task and capability of the worker. Of course, we do not know the ordering of the workers (or perhaps, the ordering of the tasks either). Thus Shah et al. Shah et al. (2016b) proposed modeling M as the result of applying unknown permutations to the rows and/or columns of a bivariate isotonic matrix.

Estimating shape constrained matrices also arises in the context of noisy sorting and aggregating ranking information Shah et al. (2016b). We say that a matrix M is *strongly stochastically transitive* (SST) if $M_{i,j} + M_{j,i} = 1$ and after applying some unknown permutation to its rows and columns we get a matrix whose rows and columns are non-decreasing. The interpretation is that $M_{i,j}$ represents the probability of ranking element i above element j when we compare them. If we can estimate M from partial and noisy observations, it would allow us to accurately predict the outcomes of subsequent comparisons. This model is more flexible than making the parametric assumption that there is some fixed probability that each comparison gives the correct answer, regardless of the pair of elements being compared, as in earlier work Braverman and Mossel (2008). It is also supported by a variety of empirical studies Davidson and Marschak (1959); McLaughlin and Luce (1965); Tversky (1972). Notably, Ballinger and Wilcox Ballinger and Wilcox (1997) studied a number of models for noisy rankings and found that the assumption of being strongly stochastically transitive was the only one to usually survive their scrutiny.

In either of these problems, most works seek to find an estimate $\widehat{M}$ that minimizes

$$\frac{1}{nm}\|\widehat{M} - M\|_F^2$$

However there are currently wide and persistent computational vs. statistical gaps for non-parametric shape constrained estimation. Chatterjee and Mukherjee Chatterjee and Mukherjee (2019) gave the first efficient algorithms based on spectral methods. He showed that they achieve the rate $\widetilde{O}(n^{-1/4})$. Shah et al. Shah et al. (2016a) showed that the minimax optimal rate is $\widetilde{O}(n^{-1})$ but computing their estimator requires brute-force search over the set of all permutations. Shah et al. Shah et al. (2016a) also improved the analysis of the spectral estimator to obtain the rate $\widetilde{O}(n^{-1/2})$. Later Shah et al. Shah et al. (2019) gave an alternative estimator based on Borda counts that also achieves the same rate. Finally, Mao et al. Mao et al. (2018) gave an estimator that works by approximately sorting the rows and columns of the matrix of observations and then applying isotonic regression. They proved that their estimator achieves the rate $\widetilde{O}(n^{-3/4})$. However this is the natural barrier to a broad class of approaches that are based on figuring out the relative order of pairs of rows from local statistics. Any approach that makes a polynomial improvement in the error rate must inherently use information about the entire matrix to figure out the relative ordering of pairs of rows that cannot be separated using only local information, as we explain later (see Theorem 3).

Question 1 *Is it possible to exploit global information to get better estimators for shape-constrained estimation?*

1. The term non-parametric has multiple uses within statistics. Here, as in earlier work on this topic, we use it to refer to models where the number of parameters is at least as large as the number of observations.

Various authors Flammarion et al. (2019); Shah et al. (2019) have conjectured that there are fundamental computational vs. statistical gaps for shape-constrained estimation. The natural question is: Have we already reached the algorithmic limits of what is possible?

1.2. Our Results

Here we introduce a framework for exploiting global information in shape-constrained estimation. As we alluded to earlier, what confounds existing approaches is that there are pairs of rows r_1 and r_2 that are far apart but whose relative ordering we cannot figure out from information just about them. In fact, if all the other rows in the matrix were 0, then it is information theoretically impossible to figure out whether r_1 is above r_2 in the unknown permutation, or if it is the other way around. But if that were the case, getting their relative order wrong would not contribute much to our overall estimation error. The difficulty is: Can there be many such pairs of rows that drive our overall estimation error up? Our key insight is that if this were the case, we could use information from the other rows in the matrix to figure out the relative ordering of r_1 and r_2 (see Section 3 for a more detailed explanation).

We introduce a new family of test functions. Rather than summing entries of M (or rather, noisy estimates of the entries) over contiguous intervals as in Mao et al. Mao et al. (2018), we use unions of contiguous intervals. In fact, Mao et al. Mao et al. (2018) conjecture that no algorithm that uses only information about the sums of entries along contiguous intervals can break the $\widetilde{O}(n^{-3/4})$ barrier. A major technical challenge in our work is that we need a superconstant number of steps to recover all the information about the unknown permutations that we can. Roughly, we use information from the columns to refine our estimate of the correct order of the rows, which we then use in turn to further refine our estimate for the correct order of the columns, and so on. We need many carefully chosen definitions to track how our iterative algorithm makes progress. These complications were not present in earlier works that estimate the permutation on rows and columns using a one-shot procedure.

Our first main result shows that there is actually no computational vs. statistical gap when only the rows are permuted:

Theorem 1 *There is an estimator $\widehat{M}$ computable in time $O\left(n^{2+o(1)}\right)$ so that for any bivariate isotonic matrix with an unknown permutation σ applied to its rows M, given $\Theta(n^2)$ Bernoulli observations of its entries has*

$$\mathbb{E}[\frac{1}{n^2}\|\widehat{M} - M\|_F^2] \leq Cn^{-1+o(1)}$$

Prior to our work, the best known error rates for polynomial time estimators were still stuck at $\widetilde{O}(n^{-3/4})$ Mao et al. (2018) (even in the case when only the rows are permuted) which is quite far from the minimax optimal error rate. Moreover the algorithm in Mao et al. (2018) runs in time $\widetilde{O}(n^{2.5})$ which is worse than ours, which is essentially optimal and runs in time that is nearly linear in the size of the output.

We also use our framework in the setting where there are unknown permutations acting on the rows and columns (we remark that the special case of strong stochastically transitive matrices is only easier than what we consider here, because it corresponds to making the further assumption that the permutations acting on the rows and columns are the same and that $M_{i,j} + M_{j,i} = 1$). Our second main result is:

Theorem 2 *There is an estimator $\widehat{M}$ computable in time $O\left(n^{2+o(1)}\right)$ so that for any bivariate isotonic matrix with unknown permutations π and σ applied to its rows and columns M, given $\Theta(n^2)$ Bernoulli observations of its entries has*

$$\mathbb{E}[\frac{1}{n^2}\|\widehat{M} - M\|_F^2] \leq Cn^{-5/6+o(1)}$$

There is a close connection between the error rate and how many entries of M we need to observe to get some desired average error. In particular, suppose we fix a target accuracy $\epsilon = o(1)$ and we are working in the model where an unknown permutation is acting on the rows. Further suppose we choose uniformly random entries of M and get to observe the outcome of a Bernoulli random variable whose probability of being one is the unknown value $M_{i,j}$. Then our first main result implies that we only need to observe $O(n^{1+o(1)})$ random entries to get the average error of our estimator down to ϵ. See Section 2.1 for a discussion of how changing the parameter in the sub-Gaussian bound on the noise affects the overall estimation error. When both the rows and columns are permuted we need only $O(n^{7/6+o(1)})$ observations. Finally we implement our algorithms and show that, on a natural benchmark studied in prior work, our algorithm empirically achieves better error rates too.

More broadly, our algorithms point to the likely difficulty of predicting where computational vs. statistical gaps for shape-constrained estimation fit in. Using simple local statistics is not necessarily the limit of what can be done with efficient algorithms, although it does seem (to us) that there ought to be some inherent limits to what is possible.

2. Problem Setup

We let $\mathsf{Mat}_{n\times n}$ be the set of $n \times n$ matrices with entries between 0 and 1. We let $\mathsf{BISO}_{n\times n}$ be the subset of bivariate isotonic matrices, i.e. the subset of $\mathsf{Mat}_{n\times n}$ with entries sorted in nondecreasing order in each row and column. We use $\mathsf{Perm}_{n\times n}$ to denote the subset of $\mathsf{Mat}_{n\times n}$ consisting of all matrices for which the rows and columns can be permuted to obtain a bivariate isotonic matrix. For a matrix M, we let $M_{\pi,\sigma}$ denote applying the permutation π to its rows and the permutation σ to its columns. If only the rows are being permuted, we may write M_π for short.

All of the matrices we deal with are $n \times n$ and for simplicity, we assume that n is a power of 2. Throughout we will let $l = \log_2 n$. We will often work with sets of consecutive integers, so when the context is clear, we will use $(a, b]$ to denote the set of integers $\{a+1, \ldots, b\}$. Given a matrix, we use the term *block* to refer to groups of consecutive rows and *cluster* to refer to groups of consecutive columns. In general, we will view the rows and columns of our matrices as n-dimensional vectors.

2.1. Sampling Model

Let $M \in \mathsf{BISO}_{n\times n}$. We use the same sampling model as Mao et al. (2018) where we observe noisy entries of a matrix $M_{\eta,\gamma}$ for some unknown permutations η, γ. More precisely, we receive $N \sim n^2$ observations of the form

$$y_{i_l,j_l} = M_{\eta(i_l),\gamma(j_l)} + z \tag{1}$$

where for $1 \leq l \leq N$, i_l, j_l are chosen independently and uniformly at random from $[n]$ and z is chosen from some sub-Gaussian noise distribution with mean 0. The goal is to construct an estimate $\widehat{M}$ that minimizes $||\widehat{M} - M_{\eta,\gamma}||_2^2$.

We use the same poissonization trick as Mao et al. (2018) where we assume that we receive $N' = \mathsf{Poi}(N)$ samples of the form given by (1). Let $p_{obs} = 1 - e^{-\frac{N}{n^2}}$ be the probability that we see at least one observation of a fixed entry. From our samples, we construct a matrix M' whose entries are

$$M'_{ij} = \frac{1}{p_{obs}} \cdot \frac{\sum_{(i_l,j_l)=(i,j)} y_{i_l j_l}}{\sum_{(i_l,j_l)=(i,j)} 1}$$

and $M'_{ij} = 0$ if the denominator in the above expression is 0.

Note we can write $M' = M_{\eta,\gamma} + E$ where E has entries drawn independently and at random from some sub-Gaussian distribution. From now on, we will assume that we receive observations of the form $M' = M_{\eta,\gamma} + E$ and that the entries of E are drawn from some sub-Gaussian noise distribution with sub-Gaussian parameter 1. Note that changing the sub-Gaussian parameter of E by a sub-polynomial factor affects our Frobenius squared error by a sub-polynomial factor since we can simply re-scale our observations so that E has sub-Gaussian parameter 1. Since we receive $N' = \mathsf{Poi}(N)$ total samples, we can split the samples into $n^{o(1)}$ parts and assume that we actually receive $n^{o(1)}$ independent observations M' (at a sub-polynomial cost to the Frobenius error).

To help with clarity of notation, we use $M'_{\eta,\gamma}$ to denote the observed matrix. we use id to denote the identity permutation and when there is ambiguity we write $M_{\mathsf{id}} = M$. We also use M'_{id} to denote the noisy version of M_{id} obtained by sorting the rows and columns of $M'_{\eta,\gamma}$. In general we will use r_i to denote the i^{th} row of M and r'_i to denote the same row with added noise.

3. Technical Overview

Before delving into the technical definitions and algorithm descriptions, we attempt to give some intuition towards why the algorithm in Mao et al. (2018) encounters a barrier at $\tilde{O}\left(n^{-\frac{3}{4}}\right)$ and how our techniques circumvent this barrier. Similar to Mao et al. (2018) our algorithm first attempts to estimate the unknown row and column permutation by sorting the rows and columns, and then runs isotonic regression to recover the original matrix. The error from isotonic regression is $\tilde{O}\left(n^{-1}\right)$ which matches the minimax optimal rate. We give a better algorithm for sorting the rows and columns which allows us to break the $\tilde{O}\left(n^{-\frac{3}{4}}\right)$ barrier.

3.1. Understanding the $n^{-3/4}$-Barrier

The key limitation of the algorithm in Mao et al. (2018) is that when sorting the rows it only compares two rows at a time in a vacuum, without using information from the other rows. The authors show that even in the case when the columns are perfectly sorted, there could be pairs of rows that are $\tilde{O}\left(n^{1/4}\right)$ apart in Frobenius squared error that cannot be placed in the correct order with high probability. This is captured in the theorem below.

Theorem 3 (Restated from Mao et al. (2018)) *Say we receive noisy observations of a matrix M_π where $M \in \mathsf{BISO}_{n\times n}$ is an unknown matrix and π is an unknown permutation. Let the rows of M be $r_1, \ldots, r_n$. For any algorithm that estimates the permutation $\hat{\pi}$, there must be some instance M, π such that*

$$\mathbb{E}\left[\max_{i\in[n]} ||r_{\pi(i)} - r_{\hat{\pi}(i)}||_2^2\right] \geq \tilde{O}\left(n^{1/4}\right)$$

Note a row-wise error of $\tilde{O}\left(n^{1/4}\right)$ leads to a normalized Frobenius squared error of $\tilde{O}\left(n^{-3/4}\right)$, which is exactly the barrier that Mao et al. (2018) hits. Thus, in order to break this barrier, we must use information from many rows simultaneously.

It is also worth noting that (by the above theorem) there may be rows r_i, r_j which are $\tilde{O}\left(n^{1/4}\right)$ apart in Frobenius squared error and such that their relative order cannot be determined even when using information from the rest of the matrix. However, if this occurs, we show that the error from failing to distinguish r_i and r_j can be amortized by lower error in estimating the rest of the matrix.

Another limitation of the algorithm of Mao et al. (2018) is in how it compares two rows. Given noisy observations r_i', r_j' of two rows, a natural way to compare them would be to take a subset $S \subset [n]$ and compare the sum of the entries indexed by S in each row. The algorithm in Mao et al. (2018) only considers partial sums, i.e. when S is a contiguous interval. They conjecture that algorithms which only exploit partial row and column sums cannot break the $\tilde{O}\left(n^{-3/4}\right)$ barrier. We do not tackle this conjecture. However, we give some intuition for why partial sums are limited and explain how to make more precise comparisons by looking at a more general family of subsets.

Say we have two rows r_i, r_j whose entries are sorted in increasing order and such that all entries of r_i are at most all entries of r_j. Also for the sake of simplicity, assume that all entries of r_i and r_j are integer multiples of $\frac{1}{k}$ for some integer k. Let $S \subset [n]$ be the set of indices a, such that the a^{th} entry of r_i and r_j are not equal. It is not difficult to show that S can be written as the union of at most k disjoint contiguous intervals (see Lemma 11). The main intuition from this example is that the locations where two rows are different must "concentrate". While one contiguous interval does not suffice to capture the "difference", looking at unions of small numbers of contiguous intervals gives us improved distinguishing power.

3.2. Our Method For Comparing Rows

In the previous section, we outlined three key aspects that are crucial to breaking the $\tilde{O}\left(n^{-3/4}\right)$ barrier in the case when only the rows are permuted. They are

- Use information from many rows simultaneously
- Amortize the error from being unable to distinguish two rows r_i, r_j that are far apart in Frobenius norm
- For determining the relative order of rows, look at the entries indexed by a subset S where S is a union of a small number of contiguous intervals

Now we give more detailed intuition for how our algorithm accomplishes the above. Return to the setup where we have two rows r_i and r_j whose entries are integer multiples of $\frac{1}{k}$. A few key intuitions are as follows. These intuitions will be explained in more detail in the proceeding paragraphs.

- We need information from $\sim k$ rows to identify differences of size $\frac{1}{k}$
- If there are less than k rows between two rows r_i, r_j whose differences have size $\sim \frac{1}{k}$ then the error from not distinguishing r_i and r_j can be amortized
- To select S, we analyze contiguous rectangles and compute the mean of all of the entries in each rectangle

In our example, assume for the sake of simplicity that the corresponding entries of r_i and r_j either differ by 0 or $\frac{1}{k}$. Note if there are fewer than k^2 locations where r_i and r_j are different, then the Frobenius squared error between r_i and r_j is less than 1 and we do not actually need to be able to distinguish them in order to reach the minimax optimal error rate. Essentially, the threshold for which our algorithm needs to work is when there are $\sim k^2$ differences between r_i and r_j. If there are no rows between r_i and r_j in the sorted order then we may still run into the same issue that pairwise comparisons only allow us to guarantee normalized Frobenius error down to $\tilde{O}\left(n^{-3/4}\right)$. However, observe the following. The L^1 distance between r_i and r_j is $\sim k$. The L^1 distance between the smallest and largest row is at most n. Thus, very roughly, the threshold that our algorithm needs to handle is when there are $\sim k$ rows in between r_i and r_j in the sorted order. If there are fewer rows between r_i and r_j, then our algorithm makes up for the error of not distinguishing r_i and r_j by having smaller error elsewhere. The intuition that we need k rows to identify differences of size $\frac{1}{k}$ is crucial in the actual algorithm and proof (for instance, see the second clause of Lemma 18 and the third clause of Definition 21).

Now, making one more simplification, we arrive at the following "core instance". We have $\frac{k}{2}$ copies of r_i and $\frac{k}{2}$ copies of r_j that form a $k \times n$ matrix. We are given noisy observations of their entries and we need to separate the two "types". Partition $[n]$ into intervals of size k, say $I_1 = \{1, 2, \ldots, k\}, I_2 = \{k+1, \ldots, 2k\}, \ldots, I_{\frac{n}{k}} = \{n-k+1, \ldots, n\}$. We further simplify the instance by assuming that r_i and r_j are constant on each interval. To see why this is a reasonable assumption, note that there are $\sim k^2$ differences and the set of differences is the union of at most k distinct intervals. Thus, it suffices to look at intervals of length k. Our goal is to identify the intervals where r_i and r_j are different.

We now show a technique for the special case outlined above and provide some intuition for how our algorithm works in the general case. Say we have three consecutive intervals I_{a-1}, I_a, I_{a+1} such that r_i and r_j differ on I_a but not I_{a-1} or I_{a+1}. Say on I_a the entries of r_i are $\frac{x}{k}$ and the entries of r_j are $\frac{x+1}{k}$. Then the mean of the entries in I_{a-1} across all of the rows is at most $\frac{x}{k}$ while the mean of the entries in I_{a+1} across all of the rows is at least $\frac{x+1}{k}$. This suggests that to identify the differences between r_i and r_j, we should look at the means of all of the entries in each of the intervals $I_1, \ldots, I_{\frac{n}{k}}$. Note that in one interval, there are k^2 entries (since there are k rows), so a difference of $\frac{1}{k}$ can be detected after adding entry-wise noise. Note it is the step of computing means of contiguous blocks that allows us to aggregate information from many rows simultaneously.

In the general case there are many complications. Even in the special case above, there is an additional complication when the intervals where r_i and r_j differ are consecutive (i.e. r_i, r_j differ on I_a and I_{a+1}). Our full algorithms require many additional procedures to deal with such complications, which will not be discussed here. we hope the simplified example above illuminates the motivation for one crucial step of our algorithm.

3.3. Our Method for 2D Sorting

In the case when both the rows and columns of the matrix are permuted, there is one more aspect of our algorithm that allows us to beat the $\tilde{O}\left(n^{-3/4}\right)$ barrier. This is the number of rounds of "adaptivity". In particular, our algorithm iteratively sorts the rows and columns, using an improved estimate of the row permutation to help sort the columns and vice versa. While the algorithm in Mao et al. (2018) only performs 2 adaptive rounds, our algorithm performs $n^{o(1)}$ adaptive rounds, and we believe this is necessary for obtaining better error rates.

4. Basic Definitions

We will use L to denote the set $\{1, 2, 4, \ldots, 2^l\}$, the set of powers of 2 between 1 and n. For a matrix M or vector v, let $\mu(M)$ (respectively $\mu(v)$) denote the mean of its entries and for a vector v, we may use v_i or $v^{(i)}$ interchangeably to denote its i^{th} entry. For two vectors u, v, we say $u \leq v$ if every entry of u is at most as large as the corresponding entry of v. When u and v are rows in a matrix $M \in \mathsf{Perm}_{n\times n}$, we say v is bigger than u or u is smaller than v if $u \leq v$.

Throughout this paper, when we say that an event X_n occurs with negligible probability, we mean that for any constant c, as $n \to \infty$, X_n occurs with probability less than $\frac{1}{n^c}$.

Definition 4 *Let V_n be the set of n-dimensional vectors with all entries between 0 and 1. Let $U_n \subset V_n$ be the subset of V_n containing all vectors with entries sorted in nondecreasing order. We call vectors in U_n well-sorted. Note that the rows and columns of M_{id} are well-sorted.*

Definition 5 *For a vector $v \in R^n$ and a set of indices $S \subset [n]$, we define*

$$\sigma(v, S) = \frac{1}{\sqrt{|S|}} \sum_{i \in S} v_i$$

Definition 6 *For a set of vectors $v_1, \ldots, v_k$, define the multidimensional variance*

$$V(\{v_1, \ldots, v_k\}) = \frac{1}{k} \sum_{1 \leq i < j \leq k} ||v_i - v_j||_2^2$$

Note that if the vectors are 1-dimensional (i.e. real numbers) then the above coincides with the usual definition of variance.

Definition 7 *Given a matrix M, a set S of rows and a set T of columns, let $R(M, S, T)$ denote the restriction of M to the rows in S and the columns in T. If T is the set of all columns, we may write $R(M, S)$ for simplicity. Define $\mu(M, S)$ as the mean of all of the entries in the corresponding restriction and define $\mu(M, S, T)$ similarly.*

Definition 8 *Let $\mathsf{In}_{a,b}$ denote the set of integers in the interval $(ab, a(b+1)]$. Let $D_{i,j} = \mathsf{In}_{2^i,j}$ denote the set of integers in the interval $\left(j \cdot 2^i, (j+1) \cdot 2^i\right]$. We call intervals of the form $D_{i,j}$ dyadic. We call a set of consecutive rows indexed by $D_{i,j}$ a dyadic block and a set of columns indexed by $D_{i,j}$ a dyadic cluster.*

5. Algorithms

5.1. Meta Algorithm

Our algorithm for the general case when both the rows and columns are permuted consists of three high-level steps, similar to the previous algorithm given in Mao et al. (2018).

Algorithm 1 Meta Algorithm

1: Split the observations into $n^{o(1)}$ parts
2: Run 2D MULTISCALE SORT (described below) to obtain estimates for the hidden permutations $\hat{\eta}, \hat{\gamma}$.
3: Let $\widehat{M}$ be the matrix in the family $\{M_{\hat{\eta},\hat{\gamma}} | M \in \mathsf{BISO}_{n\times n}\}$ that minimizes $||\widehat{M} - M'||_2^2$. Output $\widehat{M}$.

Note that the last step is a convex optimization problem and can be solved in almost-linear time Kyng et al. (2015); Bril et al. (1984). The main contribution in this paper is an improved algorithm for the sorting step which we describe in the proceeding sections. To see why the problem of estimating the original matrix reduces to sorting the rows and columns, we recall Proposition 1 in Mao et al. (2018).

Theorem 9 *[Restated from Mao et al. (2018)] Let $M \in \mathsf{BISO}_{n\times n}$ and let $M'_{\eta,\gamma} = M_{\eta,\gamma} + E$ where η, γ are permutations on $[n]$ and E is a matrix with entries drawn from a sub-Gaussian noise distribution with variance ζ^2.*

Let $\hat{\eta}$ and $\hat{\gamma}$ be estimates for η, γ and let $\widehat{M}$ be the matrix in the family $\{M_{\hat{\eta},\hat{\gamma}} | M \in \mathsf{BISO}_{n\times n}\}$ that minimizes $||\widehat{M} - M'_{\eta,\gamma}||_2^2$. Then with at least $1 - \frac{1}{n^6}$ probability

$$||\widehat{M} - M_{\eta,\gamma}||_2^2 \leq O\left(max(\zeta^2, 1)\left(n \log^2 n + ||M_{\eta,\hat{\gamma}} - M_{\eta,\gamma}||_2^2 + ||M_{\hat{\eta},\gamma} - M_{\eta,\gamma}||_2^2\right)\right)$$

5.2. 2D Multiscale Sort

The goal of the 2D MULTISCALE SORT algorithm is to obtain estimates of η, γ from observing a matrix $M'_{\eta,\gamma}$. More specifically, we will attempt to sort the rows and columns in increasing order. The output of our sorting algorithm will be a matrix $M'_{\hat{\eta}^{-1}\eta,\hat{\gamma}^{-1}\gamma}$ and $\hat{\eta}, \hat{\gamma}$ will be our estimates of the hidden permutations.

In light of Theorem 9, if we let $\pi = \hat{\eta}^{-1}\eta$ and $\sigma = \hat{\gamma}^{-1}\gamma$ it suffices to bound $||M_{\pi,\mathsf{id}} - M_{\mathsf{id}}||_2^2$ and $||M_{\mathsf{id},\sigma} - M_{\mathsf{id}}||_2^2$

The MULTISCALE SORT algorithm will iteratively call several subroutines which we describe below. At a high level, in MULTISCALE SORT we iteratively sort the rows and the columns. To sort the rows, we split the rows into two halves using BLOCK SORTING such that the rows in one half are larger than the rows in the other half. We then recurse to further sort the rows within each half.

Algorithm 2 2D MULTISCALE SORT Overview

Input matrix $M'_{\eta,\gamma}$;
  for i *in* $[l^{10}]$ **do**
    for j *in* $\{0, 1, \ldots, l-1\}$ **do**
      Run BLOCK SORTING on all dyadic blocks of size $\frac{n}{2^j}$;
    end
  Transpose matrix to swap rows and columns;
  end

Algorithm 3 BLOCK SORTING Overview

Input block $R(M', (a, b])$;
Initialize lower set $X_1 = \emptyset$, upper set $X_2 = \emptyset$, count $= 0$; **while** $|X_1| + |X_2| < b - a$ *and* *count* $< n^{o(1)}$ **do**
  Run 2D PIVOTING ALGORITHM to add some subset of rows to X_1 or X_2;
  Consider restriction of M' to rows that have not yet been added to either X_1 or X_2;
  count $=$ count $+ 1$;
  end
Add rows that have not been added to X_1 or X_2 arbitrarily so that

$$|X_1| = \lfloor \frac{b-a}{2} \rfloor, |X_2| = \lceil \frac{b-a}{2} \rceil$$

;
Permute rows in $R(M', (a, b])$ so that all rows in X_2 appear above all rows in X_1;

As mentioned in Section 2, we can assume that we have access to a sub-polynomial number of independent samples $M'_{\eta,\gamma}$ (i.e. the noise is drawn independently for each of the samples).

5.3. Block Sorting

Goal: The block sorting subroutine splits the rows in a block of the observed matrix $R(M', (a, b])$ into two parts, an upper and lower half. The goal is to obtain a partition that almost satisfies the property that all rows in the upper half are larger than all rows in the lower half. More formally, our goal is to obtain a partition $X_1 \cup X_2 = (a, b]$ such that

- X_1 and X_2 are disjoint
- $|X_1| - |X_2| \in \{0, 1\}$
- For any $i \in X_1$ and $j \in X_2$, $r_i < r_j$.

In the above, X_1 is the lower half and X_2 is the upper half. We will not be able to satisfy the last property exactly, but we will show that our algorithm obtains a partition that is close to satisfying the desired property, where closeness is quantified in terms of Frobenius error.

Overview: To sort a block of the observed matrix, we will iteratively add rows, for which we are confident about their position, to the upper and lower half. We will do this using the 2D PIVOTING ALGORITHM described below. To sort the entire block, we iteratively apply the pivoting algorithm, remove the rows that have been added to the upper or lower half, and then recurse on the remaining rows.

5.3.1. 2D PIVOTING ALGORITHM

Goal: Given a block and an index, determine for some subset of the rows whether their rank is higher or lower than the given index. More formally, we will work with a block $R(M', (a, b])$ and also a pivot index, say $0 \le d \le k$ where $k = b - a$. The goal is to create two sets of rows X_1 and X_2 such that the following holds with high probability.

- For any $r'_i \in X_1$, r_i is among the d smallest rows in the set $\{r_{a+1}, \ldots, r_b\}$.

- For any $r'_i \in X_2$, r_i is among the $k-d$ largest rows in the set $\{r_{a+1}, \ldots, r_b\}$.

We call X_1 the lower set and X_2 the upper set.

Algorithm Description: We will draw a fresh set of samples for each run of the pivoting algorithm. Now we choose three integers $r, t, w \in L$. We will call r the cluster size and t the difference size. Note there are at most l^3 possible choices for r, t, w. For each tuple (r, t, w) we run the three steps below.

STEP 1

Overview: We will find a set of candidate locations that will be used to distinguish the rows in the given block $R(M', (a, b])$. The set we construct will be a union of disjoint intervals of length r. We construct the set by comparing the means of the entries in various contiguous rectangles. Note how this idea was motivated in Section 3.

Description: For each (r, t) we build an auxiliary matrix $M'(a, b, r, t)$ as follows. Note the entire block in consideration is a $k \times n$ matrix. We can break this into $\frac{n}{r}$ matrices of size $k \times r$. Say these matrices are, in order from left to right, $A_1, \ldots, A_{\frac{n}{r}}$. Let S_0 be the subset of $\{1, \ldots, \frac{n}{r}\}$ containing indices i such that either $|\mu(A_i) - \mu(A_{i+1})| \geq \frac{1}{30t}$ or $|\mu(A_i) - \mu(A_{i-1})| \geq \frac{1}{30t}$. Let $\mu\left(A_{\frac{n}{r}+1}\right) = 1$ and $\mu(A_0) = 0$ (as these indices are "out of bounds"). Let

$$S = \{1, \ldots, \frac{n}{r}\} \cap \left(\bigcup_{i \in S_0} \{i-1, i, i+1\} \right)$$

Note S is the union of S_0 and the set of all indices adjacent to some element of S_0. If the set S consists of more than $1000t$ elements, don't proceed to the second step. We call S_0 the starter-set and S the preliminary-set.

STEP 2

Overview: We will further refine the set constructed in the previous step. For each of the disjoint intervals of length r in the set from the previous step, we look at the corresponding $r \times k$ rectangle of $R(M', (a, b])$ and take the mean of the entries in each row to get a vector $v \in \mathbb{R}^k$. We keep the intervals for which $||v||_2$ is large and throw away the rest.

Description: We have a set S with $|S| \leq 1000t$. Say $S = \{i_1, \ldots, i_a\}$. For each $i \in \{1, 2, \ldots, \frac{n}{r}\}$, let c_i be the column vector obtained by taking the mean of the entries in each row of A_i. Note c_i has dimension k. Now let c'_i be the column vector obtained by subtracting $\mu(c_i)$ from each entry of c_i (so c'_i is a k-dimensional vector such that the mean of its entries is 0). We will now construct $T \subset S$ as follows. If $a \leq w$, set $T = S$. Otherwise, let T be a set of w indices among $\{i_1, \ldots, i_a\}$ corresponding to the w columns where the vector c'_{i_j} has the largest magnitude.

STEP 3

Overview: Given a set $R \subset [n]$ of candidate locations from the previous step, we sort the rows based on $\sigma(r'_i, R)$, the sum of their entries in locations indexed by R. If the sum is too small, the row is added to the lower set and if the sum is too large, the row is added to the upper set.

Description: Define the test-set $R = \bigcup_{j \in T}(r(j-1), rj]$. Draw a fresh sample M' (i.e. the noise added to each entry is resampled from the noise distribution). Sort the rows in the current block $r'_1, \ldots, r'_k$ according to $\sigma(r'_i, R)$. Let λ be a permutation on k elements such that $\sigma(r'_{\lambda(1)}, R) \leq \cdots \leq \sigma(r'_{\lambda(k)}, R)$. Let $\tau \approx 10l$ be a chosen threshold. Let $Y_{1|(r,t,w)}$ be the set of all $1 \leq i \leq k$ such that

$$\sigma(r'_{\lambda(d+1)}, R) - \sigma(r'_i, R) \geq \tau$$

and let $Y_{2|(r,t,w)}$ be the set of all $1 \leq i \leq k$ such that

$$\sigma(r'_i, R) - \sigma(r'_{\lambda(d)}, R) \geq \tau$$

For edge cases, if $d = 0$ let $Y_{1|(r,t,w)} = \{1, 2, \ldots, k\}$ and $Y_{2|(r,t,w)}$ be empty and vice versa if $d = k$. Now take X_1 to be the union of the set $Y_{1|(r,t,w)}$ over all choices of r, t, w and similarly let X_2 be the union of $Y_{2|(r,t,w)}$ over all choices of r, t, w. As we will show later on, with high probability the sets X_1 and X_2 will be disjoint.

5.3.2. BLOCK SORTING USING THE PIVOTING ALGORITHM

We now explain how to split a block $R(M', (a, b])$ into two halves by iteratively running the pivoting algorithm. First run the pivoting algorithm on the entire block with $d = \lfloor \frac{k}{2} \rfloor$. we obtain two sets X_1 and X_2. Add all elements of X_1 to L and all elements of X_2 to U. Now set $d = \lfloor \frac{k}{2} \rfloor - |X_1|$ and run the pivoting algorithm on the matrix formed by the remaining rows, $R(M', (a, b]/(X_1 \cup X_2))$. Repeat this process $l^{l^{0.51}}$ times or until all rows have been added to either the upper or lower half. If any rows remain, add them to either half arbitrarily so that U and L have the correct size.

5.4. Omitted Algorithms

A complete description of all relevant algorithms in our paper can be found in Appendix B. This includes a full description of 2D MULTISCALE SORT and a variant, 1D MULTISCALE SORT, that allows us to obtain a better error guarantee when only the rows are permuted.

References

T Parker Ballinger and Nathaniel T Wilcox. Decisions, error and heterogeneity. *The Economic Journal*, 107(443):1090–1105, 1997.

Mark Braverman and Elchanan Mossel. Noisy sorting without resampling. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 268–276. Society for Industrial and Applied Mathematics, 2008.

Gordon Bril, Richard Dykstra, Carolyn Pillers, and Tim Robertson. Algorithm as 206: isotonic regression in two independent variables. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 33(3):352–357, 1984.

Sabyasachi Chatterjee and Sumit Mukherjee. Estimation in tournaments and graphs under monotonicity constraints. *IEEE Transactions on Information Theory*, 2019.

Donald Davidson and Jacob Marschak. Experimental tests of a stochastic decision theory. *Measurement: Definitions and theories*, 17:233–269, 1959.

Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28 (1):20–28, 1979.

Nicolas Flammarion, Cheng Mao, Philippe Rigollet, et al. Optimal rates of statistical seriation. *Bernoulli*, 25(1):623–653, 2019.

David R Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. In *Advances in neural information processing systems*, pages 1953–1961, 2011.

Rasmus Kyng, Anup Rao, and Sushant Sachdeva. Fast, provable algorithms for isotonic regression in all l_p-norms. In *Advances in neural information processing systems*, pages 2719–2727, 2015.

Cheng Mao, Ashwin Pananjady, and Martin J Wainwright. Towards optimal estimation of bivariate isotonic matrices with unknown permutations. *arXiv preprint arXiv:1806.09544*, 2018.

Don H McLaughlin and R Duncan Luce. Stochastic transitivity and cancellation of preferences between bitter-sweet solutions. *Psychonomic Science*, 2(1-12):89–90, 1965.

Vikas C Raykar, Shipeng Yu, Linda H Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning from crowds. *Journal of Machine Learning Research*, 11 (Apr):1297–1322, 2010.

Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. In *Proceedings of the International Congress of Mathematicians 2010 (ICM 2010) (In 4 Volumes) Vol. I: Plenary Lectures and Ceremonies Vols. II–IV: Invited Lectures*, pages 1576–1602. World Scientific, 2010.

Nihar Shah, Sivaraman Balakrishnan, Aditya Guntuboyina, and Martin Wainwright. Stochastically transitive models for pairwise comparisons: Statistical and computational issues. In *International Conference on Machine Learning*, pages 11–20, 2016a.

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*, 2016b.

Nihar B Shah, Sivaraman Balakrishnan, and Martin J Wainwright. Feeling the bern: Adaptive estimators for bernoulli probabilities of pairwise comparisons. *IEEE Transactions on Information Theory*, 2019.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics, 2008.

Amos Tversky. Elimination by aspects: A theory of choice. *Psychological review*, 79(4):281, 1972.

Appendix A. Experimental Results

We implemented a simplified version of our algorithm and a version of the algorithm described in Mao et al. (2018). Below we plot the performance of both algorithms. Note the steeper slope of the blue lines indicates a better dependence on n in the error of the estimator.

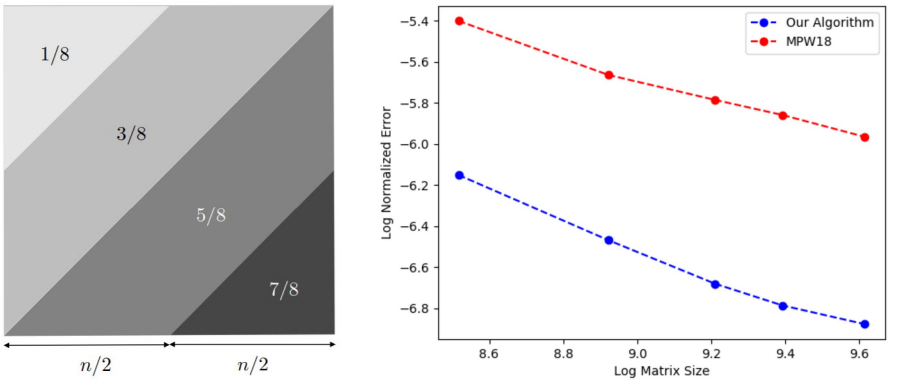

Figure 1: The diagram on the left shows the family of matrices that we work with. The plot on the right shows the log-normalized Frobenius error, $\frac{1}{n^2}||\widehat{M} - M||_2^2$, of the two estimators for various values of n.

Appendix B. Omitted Algorithm Descriptions

B.1. Full 2D MULTISCALE SORT Algorithm

Here we complete the description of the 2D MULTISCALE SORT algorithm, using BLOCK SORTING as a subroutine.

B.1.1. FULL ROW SORTING

We will iteratively use the block sorting algorithm to sort all of the rows of the observed matrix. First we run the block sorting algorithm on the entire matrix to split the set of rows into two halves of size $\sim \frac{n}{2}$. We will permute the rows of M' so that all rows indexed by U appear above all rows indexed by L. We call the set of rows indexed by U the upper block and the set of rows indexed by L the lower block. We will then run the block sorting algorithm on the upper and lower block to split each of the two blocks obtained in the first step into an upper and lower block of size $\sim \frac{n}{4}$. We recurse again on the blocks of size $\frac{n}{4}$. In total we will run l phases. Note that the number of independent samples we need in one step of the recursion is upper bounded by the largest number of samples required by the block sorting algorithm (since the blocks we are sorting are independent, we can split a fresh sample of the entire matrix into one sample for each block).

B.1.2. FULL ALGORITHM

The full MULTISCALE SORT algorithm works by first sorting the rows using the method described above. Then we fix the row permutation and sort the columns using the same method. We repeatedly

sort the rows and then columns for $O(l^{10})$ iterations and output the resulting matrix. It is not difficult to see that we need at most $l^{l^{0.52}} = n^{o(1)}$ independent samples.

B.2. 1D Multiscale Sort

We now show a different algorithm, called 1D MULTISCALE SORT that (when plugged into our meta algorithm in place of 2D MULTISCALE SORT) gives a better error guarantee when the column permutation is known to be the identity. The overall algorithm will involve one pass of the full row sorting algorithm, except we will use a different pivoting algorithm. Note that we do not change the block sorting algorithm and full row sorting algorithm except for the different pivoting subroutine described below.

B.2.1. 1D PIVOTING ALGORITHM

The goal and setup of the 1D PIVOTING ALGORITHM is the same as for the 2D PIVOTING ALGORITHM. Say we are given a pivoting index d. As in the general case, we will draw a fresh set of samples for each run of the pivoting algorithm. Choose parameters r, t as in the general pivoting algorithm.

STEP 1

Construct the preliminary set as in the 2D PIVOTING ALGORITHM.

STEP 2

Overview: Given the set from the previous step, we will compute weights for the elements. We will compute the weights by constructing an auxiliary matrix and using its principal component. Note we can view a weighted set as a vector v.

Description: We have a set with $|S| \leq 1000t$. Say $S = \{i_1, \ldots, i_a\}$. For each $i \in \{1, 2, \ldots, \frac{n}{r}\}$, let c_i be the column vector obtained by taking the mean of the entries in each row of A_i. Let c_i' be obtained by demeaning c_i (i.e. subtracting the mean of the entries of c_i from each entry). Now let N be the $k \times a$ matrix that has columns $c_{i_1}', \ldots, c_{i_a}'$. Let $v \in \mathbb{R}^a$ be a unit vector that maximizes $||Nv||_2$. Let v^+ be the vector obtained by taking the positive entries of v and zeroing out the other entries and let v^- be the vector obtained by taking the negative entries of v and zeroing out the other entries. Note $v = v^+ - v^-$.

STEP 3

Overview: We will sort the rows based on their inner product $r_i \cdot v$ where v is the vector constructed in the previous step. Rows for which the inner product is too small are added to the lower set and rows for which the inner product is too large are added to the upper set.

Description: Draw a fresh sample M' and use it to construct the matrix N (i.e. we construct the same matrix N as before but the noise is effectively resampled). Let the rows of N be $s_1', \ldots, s_k'$. If $||Nv^+||_2 \geq ||Nv^-||_2$ let $v_{\text{test}} = \frac{v^+}{||v^+||}$ and otherwise set $v_{\text{test}} = \frac{-v^-}{||v^-||}$. We essentially imitate the third step of the general pivoting algorithm but project onto v_{test} to sort the rows. In particular, let

λ be a permutation on $[k]$ such that $s'_{\lambda(1)} \cdot v_{\text{test}} \leq \cdots \leq s'_{\lambda(k)} \cdot v_{\text{test}}$ and let $\tau \approx 10l$ be a chosen threshold. Let $Y_{1|(r,t)}$ be the set of all $1 \leq i \leq k$ such that

$$s'_{\lambda(d+1)} \cdot v_{\text{test}} - s'_i \cdot v_{\text{test}} \geq \frac{\tau}{\sqrt{r}}$$

and let $Y_{2|(r,t)}$ be the set of all $1 \leq i \leq k$ such that

$$s'_i \cdot v_{\text{test}} - s'_{\lambda(d)} \cdot v_{\text{test}} \geq \frac{\tau}{\sqrt{r}}$$

Edge cases $d = 0$ and $d = k$ are dealt with as in the general case. Now let X_1 be the union of $Y_{1|(r,t)}$ over all choices of r, t and let X_2 be the union of $Y_{2|(r,t)}$ over all choices of r, t.

Appendix C. Block Differences

The analysis of our algorithms will rely on understanding certain structures in the rows and differences between rows of the underlying matrix. In this section, we formally introduce several tools that will be used extensively later on.

Definition 10 *Given vectors u, v, we say u is (x, y)-above v if $u \geq v$ and there are x locations where the entry in u is at least $\frac{1}{y}$ more than the corresponding entry in v.*

Lemma 11 *Say we have two vectors $v = (v_1, \ldots, v_n)$ and $u = (u_1, \ldots, u_n)$ with $u \geq v$ and $u, v \in U_n$. Let $t_i = u_i - v_i$. Then for any subsequence $i_1, \ldots, i_k$ of $1, 2, \ldots, n$,*

$$|t_{i_2} - t_{i_1}| + \cdots + |t_{i_k} - t_{i_{k-1}}| \leq 2$$

Proof We have

$$|t_{i_{j+1}} - t_{i_j}| = |(u_{i_{j+1}} - v_{i_{j+1}}) - (u_{i_j} - v_{i_j})| \leq |u_{i_{j+1}} - u_{i_j}| + |v_{i_{j+1}} - v_{i_j}| = (u_{i_{j+1}} - u_{i_j}) + (v_{i_{j+1}} - v_{i_j})$$

Plugging this in we get,

$$|t_{i_2} - t_{i_1}| + \cdots + |t_{i_k} - t_{i_{k-1}}| \leq (u_{i_k} - u_{i_1}) + (v_{i_k} - v_{i_1}) \leq 2$$

■

Lemma 12 *Say we have two vectors $u, v \in V_n$ such that $u \geq v$ and $||u - v||^2 \geq \frac{2}{n}$. Then there exists $x, y \in L$ such that u is (x, y)-above v and $\frac{x}{y^2} \geq \frac{||u-v||^2}{16l}$.*

Proof For each $1 \leq i \leq l$, let x_i be the number of locations where the entry in u is at least $\frac{1}{2^i}$ larger than the corresponding entry in v. Let x'_i be the smallest power of 2 that is at least x_i (or 0 if $x_i = 0$). Note by the minimality of x'_i, u is $\left(\lceil \frac{x'_i}{2} \rceil, 2^i\right)$-above v for every i. Next note that

$$||u - v||^2 \leq x'_1 + x'_2 \left(\frac{1}{2}\right)^2 + \cdots + x'_l \left(\frac{1}{2^{l-1}}\right)^2 + \frac{1}{n}$$

This is because all entries of $u - v$ are at most 1. The number of entries that are between $\frac{1}{2}$ and 1 is at most x'_1, the number of entries that are between $\frac{1}{4}$ and $\frac{1}{2}$ is at most x'_2 and so on. For the last term, the total L^2 error contributed by entries where $u - v$ is less than $\frac{1}{2^l}$ is at most $\frac{1}{n}$. Therefore, there exists some i such that

$$x'_i \left(\frac{1}{2^{i-1}}\right)^2 \geq \frac{||u-v||^2 - \frac{1}{n}}{l} \geq \frac{||u-v||^2}{2l}$$

Since u is $\left(\lceil \frac{x'_i}{2} \rceil, 2^i\right)$-above v, choosing $x = \lceil \frac{x'_i}{2} \rceil$ and $y = 2^i$, we are done (note $\lceil \frac{x'_i}{2} \rceil$ is a power of 2 since x'_i is, the ceiling is only there for the case when $x'_i = 1$).

■

Claim Say we have a set of row vectors $r_1 \leq \cdots \leq r_{k+1} \in V_n$ such that r_{i+1} is (x, y)-above r_i for all $1 \leq i \leq k$. Also assume $k \leq n$. Then there must exist x', y' such that

- x' is a power of 2 between 1 and n
- y' is a power of 2
- r_{k+1} is (x', y') above r_1
- $x' \geq \frac{x}{2l}$
- $\frac{y}{n} \leq y' \leq 2y$
- $\frac{x'}{y'^2} \geq \frac{1}{8l} \cdot \frac{kx}{y^2}$

Proof For each i, place a marker on the locations among $\{1, 2, \ldots n\}$ where the entry in r_{i+1} is at least $\frac{1}{y}$ larger than the entry of r_i. For each $j \in [n]$, let α_j be the total number of markers on location j and for a constant c let β_c be the number of locations with at least c markers. Using a standard argument, there exists an integer c such that

$$c\beta_c \geq \frac{\alpha_1 + \cdots + \alpha_n}{l} \geq \frac{kx}{l}$$

Note that r_{k+1} is $\left(\beta_c, \frac{y}{c}\right)$-above r_1. We can now set x' to be the largest power of 2 at most β_c and y' to be the smallest power of 2 at least $\frac{y}{c}$. We now verify the remaining conditions. Note $\frac{y}{n} \leq y' \leq 2y$ is clear.

$$x' \geq \frac{\beta_c}{2} \geq \frac{kx}{2cl} \geq \frac{x}{2l}$$

$$\frac{x'}{y'^2} \geq \frac{c^2\beta_c}{8y^2} \geq \frac{c\beta_c}{8y^2} \geq \frac{kx}{8ly^2}$$

■

Definition 13 *Let $M \in \mathsf{Mat}_{n\times n}$. We say two sets of rows $(r_1, \ldots, r_k), (s_1, \ldots, s_k)$ of M form a (k, x, y) difference if $r_1 \leq \cdots \leq r_k$, $s_1 \leq \cdots \leq s_k$, and s_1 is (x, y)-above r_k.*

Definition 14 *Let $M \in \mathsf{Mat}_{n\times n}$. We say two sets of rows $(r_1,\dots,r_k),(s_1,\dots,s_k)$ in M form a (k,x,y)-error if they form a (k,x,y)-difference and all of $(s_1,\dots,s_k)$ are below $(r_1,\dots,r_k)$. We say the size of a (k,x,y) error is $\frac{kx}{y^2}$.*

Definition 15 *We say that a matrix $M \in \mathsf{Mat}_{n\times n}$ contains a type (a,b,x,y)-error if the following holds. There exist a sets of $2b$ rows $S_1,\dots,S_a$ such that all rows of S_i are below all rows of S_j for $i<j$ and each set forms a (b,x,y)-error. We define an (a,b,x,y)-difference similarly except we require that each set forms a (b,x,y)-difference. We say the size of a (a,b,x,y) error or difference is $\frac{abx}{y^2}$.*

Lemma 16 *Let $M \in \mathsf{Mat}_{n\times n}$ and let $r_1,\dots,r_k$ be some of its rows such that $r_1 \le \cdots \le r_k$ and $V(\{r_1,\dots,r_k\}) \ge 2\log_2 k$. There exist $c,x,y \in L$ such that $\frac{cx}{y^2} \ge \frac{V(\{r_1,\dots,r_k\})}{32l^2}$ and $(r_{k-c+1},\dots,r_k)$ and $(r_1,\dots r_c)$ form a (c,x,y)-difference.*

Proof First note that

$$V(\{r_1,\dots,r_k\}) = \frac{1}{k}\left(\sum_{1\le i<j\le k}||r_i-r_j||_2^2\right)$$

$$= \frac{1}{k}\left(||r_{\frac{k}{2}+1}-r_{\frac{k}{2}}||_2^2 + \sum_{i'=0}^{\log_2 k-1}\sum_{\substack{i<j\\ 2^{i'}\le min(i,k+1-j)<2^{i'+1}}}||r_i-r_j||_2^2\right)$$

$$\le \frac{1}{k}\left(\sum_{i'=0}^{\log_2 k-1}2^{i'+1}k||r_{k+1-2^{i'}}-r_{2^{i'}}||_2^2\right) = \sum_{i'=0}^{\log_2 k-1}2^{i'+1}||r_{k+1-2^{i'}}-r_{2^{i'}}||_2^2$$

In particular for some index i_0,

$$2^{i_0}||r_{k+1-2^{i_0}}-r_{2^{i_0}}||_2^2 \ge \frac{V(\{r_1,\dots,r_k\})}{2\log_2 k}$$

We set $c=2^{i_0}$. Note that the above gives us $||r_{k+1-2^{i_0}}-r_{2^{i_0}}||_2^2 \ge \frac{1}{2^{i_0}} \ge \frac{2}{n}$ so applying Lemma 12, there exist $x,y\in L$ such that $r_{k+1-2^{i_0}}$ is (x,y)-above $r_{2^{i_0}}$ and $\frac{x}{y^2} \ge \frac{||r_{k+1-2^{i_0}}-r_{2^{i_0}}||_2^2}{16l}$. We get that $\frac{cx}{y^2} \ge \frac{V(\{r_1,\dots,r_k\})}{32l^2}$. Also note that $(r_{k-c+1},\dots,r_k)$ and $(r_1,\dots r_c)$ form a (c,x,y)-difference so we are done. ■

C.1. Detectable Block Differences

Now that we have introduced basic tools for dealing with difference structures, the main goal in this section is to show that it "suffices" to consider (a,b,x,y) errors where the parameters a,b,x,y are in a range that can be detected by our algorithm. In other words, we show that once we control (a,b,x,y) errors for a certain regime of a,b,x,y we have an upper bound on the L^2 error. Below, $f(n)$ will be some function with $1<f(n)<n$. We will set it more precisely later on.

Definition 17 *For a permutation π on $\{1, 2, \ldots, n\}$, its dyadic decomposition is a set of $l+1$ permutations $(\pi_0, \pi_1, \pi_2, \ldots, \pi_l)$ defined as follows. Consider the list $\{\pi(1), \ldots, \pi(n)\}$ (which contains each integer between 1 and n exactly once). To obtain π_i, consider the dyadic intervals $D_{l-i,0} \ldots, D_{l-i,2^i-1}$ and for each dyadic interval, sort the elements of the list within that dyadic interval in increasing order.*

Example 1 *For $\pi = (4, 5, 1, 6, 3, 8, 7, 2)$, we have $\pi_0 = \mathsf{id} = (1, 2, 3, 4, 5, 6, 7, 8)$, $\pi_1 = (1, 4, 5, 6, 2, 3, 7, 8)$, $\pi_2 = (4, 5, 1, 6, 3, 8, 2, 7)$, $\pi_3 = \pi = (4, 5, 1, 6, 3, 8, 7, 2)$.*

Lemma 18 *Let $M \in \mathsf{BISO}_{n\times n}$ and π be a permutation. If $||M_\pi - M_{\mathsf{id}}||^2 \geq E \geq nl^{1000l^{0.5}}$ then there must exist an (a, b, x, y) error in M_π where a, b, x, y are powers of 2 with the following properties.*

- $ab\frac{x}{y^2} \geq \frac{E}{l^{0.5}}$
- $y \leq b$

Consider the dyadic decomposition of π, $(\pi_0, \pi_1, \ldots, \pi_l)$. There must exist an index $0 \leq i \leq l-1$ such that $||M_{\pi_i} - M_{\pi_{i+1}}||^2 \geq \frac{E}{l}$. We will first focus on each dyadic block $D_{l-i,j}$. Let $k = \frac{n}{2^{i+1}}$. The block $R(M_{\pi_i}, D_{l-i,0})$ consists of $2k$ rows, say $r_1, \ldots, r_k, r_{k+1}, \ldots, r_{2k}$ in order. We know $r_1 \leq r_2 \leq \cdots \leq r_{2k}$ by definition. The block $R(M_{\pi_{i+1}}, D_{l-i,0})$ consists of the same rows but in a different permutation. We use the term first half to refer to the first k rows and second half to refer to the second k rows of $R(M_{\pi_{i+1}}, D_{l-i,0})$. There must exist an integer c such that exactly c rows are swapped between the two halves when compared to $R(M_{\pi_i}, D_{l-i,0})$. Let $\{i_1, \ldots, i_c\} \subset \{1, 2, \ldots, k\}$ and $\{i'_1, \ldots, i'_c\} \subset \{k+1, \ldots, 2k\}$ such that $r_{i_1}, \ldots, r_{i_c}$ occur in the second half of$R(M_{\pi_{i+1}}, D_{l-i,0})$ and $r_{i'_1}, \ldots, r_{i'_c}$ occur in the first half of $R(M_{\pi_{i+1}}, D_{l-i,0})$. Also WLOG $i_1 < \cdots < i_c$ and $i'_1 < \cdots < i'_c$. we first prove the following inequality. Claim

$$||R(M_{\pi_i}, D_{l-i,0}) - R(M_{\pi_{i+1}}, D_{l-i,0})||^2 \leq 2\sum_{j=1}^{c} ||r_{i_j} - r_{i'_{c+1-j}}||^2$$

Proof First note

$$||R(M_{\pi_i}, D_{l-i,0}) - R(M_{\pi_{i+1}}, D_{l-i,0})||^2 = ||R(M_{\pi_i}, D_{l-i-1,0}) - R(M_{\pi_{i+1}}, D_{l-i-1,0})||^2 + ||R(M_{\pi_i}, D_{l-i-1,1}) - R(M_{\pi_{i+1}}, D_{l-i-1,1})||^2$$

For a permutation σ on k elements, let $R_\sigma(M_{\pi_{i+1}}, D_{l-i-1,0})$ denote permuting the rows in $R(M_{\pi_{i+1}}, D_{l-i-1,0})$ according to σ. Over all permutations σ, the identity permutation minimizes $||R(M_{\pi_i}, D_{l-i-1,0}) - R_\sigma(M_{\pi_{i+1}}, D_{l-i-1,0})||^2$ since when σ is the identity permutation, the rows of both matrices are sorted in increasing order. It is clear that there exists a permutation σ such that

$$||R(M_{\pi_i}, D_{l-i-1,0}) - R_\sigma(M_{\pi_{i+1}}, D_{l-i-1,0})||^2 = \sum_{j=1}^{c} ||r_{i_j} - r_{i'_{c+1-j}}||^2$$

Thus, we have

$$||R(M_{\pi_i}, D_{l-i-1,0}) - R(M_{\pi_{i+1}}, D_{l-i-1,0})||^2 \leq \sum_{j=1}^{c} ||r_{i_j} - r_{i'_{c+1-j}}||^2$$

Using a similar argument for the dyadic block $D_{l-i-1,1}$ and adding the two inequalities, we get the desired. ■

The next step in the proof of Lemma 18 will be to upper bound the quantity $\sum_{j=1}^{c} ||r_{i_j} - r_{i'_{c+1-j}}||^2$ in terms of (a, b, x, y) differences. Let 2^f be the largest power of 2 at most c. Note

$$\sum_{j=1}^{2^f} ||r_{i_j} - r_{i'_{c+1-j}}||^2 \geq \frac{1}{2} \sum_{j=1}^{c} ||r_{i_j} - r_{i'_{c+1-j}}||^2$$

since in the sum on the left hand side, we are taking the largest terms. Now note that since the terms in the sum are (weakly) decreasing,

$$S = \sum_{j=1}^{2^f} ||r_{i_j} - r_{i'_{c+1-j}}||^2 \leq ||r_{i_1} - r_{i'_c}||^2 + 2||r_{i_2} - r_{i'_{c-2}}||^2 + \cdots + 2^f ||r_{i_{2^f}} - r_{i'_{c+1-2^f}}||^2$$

Therefore, there exists some index $0 \leq g \leq f$ such that $2^g ||r_{i_{2^g}} - r_{i'_{c+1-2^g}}||^2 \geq \frac{S}{f+1}$. We will use the above and apply Lemma 12 on the rows $r_{i_{2^g}}$ and $r_{i'_{c+1-2^g}}$ to deduce the following. Claim If $||R(M_{\pi_i}, D_{l-i,0}) - R(M_{\pi_{i+1}}, D_{l-i,0})||^2 \geq \frac{8k(f+1)}{n}$ then there exist three integers $b_0, x_0, y_0 \in L$ such that $R(M_\pi, D_{l-i,0})$ contains a (b_0, x_0, y_0)-error and

$$\frac{b_0 x_0}{y_0^2} \geq \frac{||R(M_{\pi_i}, D_{l-i,0}) - R(M_{\pi_{i+1}}, D_{l-i,0})||^2}{100l^2}$$

.

Proof By the computations in the previous paragraph

$$||R(M_{\pi_i}, D_{l-i,0}) - R(M_{\pi_{i+1}}, D_{l-i,0})||^2 \leq 2 \sum_{j=1}^{c} ||r_{i_j} - r_{i'_{c+1-j}}||^2 \leq 4 \sum_{j=1}^{2^f} ||r_{i_j} - r_{i'_{c+1-j}}||^2$$
$$\leq 4(f+1)2^g ||r_{i_{2^g}} - r_{i'_{c+1-2^g}}||^2$$

In particular $||r_{i_{2^g}} - r_{i'_{c+1-2^g}}||^2 \geq \frac{8k(f+1)}{n} \frac{1}{4(f+1)2^g} \geq \frac{2}{n}$ so we can apply Lemma 12 and find integers x_0, y_0 such that $r_{i'_{c+1-2^g}}$ is (x_0, y_0)-above $r_{i_{2^g}}$ and $\frac{x_0}{y_0^2} \geq \frac{||r_{i_{2^g}} - r_{i'_{c+1-2^g}}||^2}{16l}$. Set $b_0 = 2^g$. The rows $(r_{i_1}, \ldots, r_{i_{2^g}}), (r_{i'_c}, \ldots, r_{i'_{c+1-2^g}})$ form a (b_0, x_0, y_0) error in M_π. We have

$$\frac{b_0 x_0}{y_0^2} \geq 2^g \frac{||r_{i_{2^g}} - r_{i'_{c+1-2^g}}||^2}{16l} \geq \frac{||R(M_{\pi_i}, D_{l-i,0}) - R(M_{\pi_{i+1}}, D_{l-i,0})||^2}{4(f+1) \cdot 16l}$$
$$\geq \frac{||R(M_{\pi_i}, D_{l-i,0}) - R(M_{\pi_{i+1}}, D_{l-i,0})||^2}{100l^2}$$

■

So far we have only worked with the dyadic block $D_{l-i,0}$ to find a (b, x, y)-error. Clearly Claim C.1 also applies to other dyadic blocks $D_{l-i,j}$. If we "amortize" the above inequality over all dyadic blocks at level i, we will be able to find an (a, b, x, y)-error of sufficient size.

Claim There exist $a, b, x, y \in L$ such that M_π contains an (a, b, x, y)-error and $\frac{abx}{y^2} \geq \frac{E}{400l^6}$

Proof Note

$$||M_{\pi_i} - M_{\pi_{i+1}}||^2 = \sum_{j=0}^{2^i-1} ||R(M_{\pi_i}, D_{l-i,j}) - R(M_{\pi_{i+1}}, D_{l-i,j})||^2$$

Let S be the set of indices j such that $||R(M_{\pi_i}, D_{l-i,j}) - R(M_{\pi_{i+1}}, D_{l-i,j})||^2 \geq \frac{8k(f+1)}{n}$. We have

$$\sum_{j \notin S} ||R(M_{\pi_i}, D_{l-i,j}) - R(M_{\pi_{i+1}}, D_{l-i,j})||^2 \leq \frac{8k(f+1)}{n}|S| \leq 4(f+1)$$

Since we assumed $E > n$,

$$\sum_{j \in S} ||R(M_{\pi_i}, D_{l-i,j}) - R(M_{\pi_{i+1}}, D_{l-i,j})||^2 \geq ||M_{\pi_i} - M_{\pi_{i+1}}||^2 - 4(f+1) \geq \frac{E}{l} - 4(f+1) \geq \frac{E}{2l}$$

Now for each $j \in S$, by Claim C.1, we can find $b_j, x_j, y_j \in L$ such that $R(M_\pi, D_{l-i,j})$ contains a (b_j, x_j, y_j)-error and

$$\frac{b_j x_j}{y_j^2} \geq \frac{||R(M_{\pi_i}, D_{l-i,j}) - R(M_{\pi_{i+1}}, D_{l-i,j})||^2}{100l^2}$$

There are only l^3 possibilities for the triple (b_j, x_j, y_j). Thus, there exists some (b, x, y) such that

$$\begin{aligned}\sum_{j \in S | (b_j, x_j, y_j) = (b,x,y)} \frac{b_j x_j}{y_j^2} &\geq \sum_{j \in S | (b_j, x_j, y_j) = (b,x,y)} \frac{||R(M_{\pi_i}, D_{l-i,j}) - R(M_{\pi_{i+1}}, D_{l-i,j})||^2}{100l^2} \\ &\geq \frac{1}{100l^2} \frac{\sum_{j \in S} ||R(M_{\pi_i}, D_{l-i,j}) - R(M_{\pi_{i+1}}, D_{l-i,j})||^2}{l^3} \geq \frac{E}{200l^6}\end{aligned}$$

Let $a' = |\{j \in S | (b_j, x_j, y_j) = (b, x, y)\}|$. In M_π, there are a' dyadic blocks among $D_{l-i,0}, \ldots, D_{l-i,2^i-1}$ that contain a (b, x, y)-error. Also, all of these (b, x, y)-errors are clearly disjoint. If we let a be the largest power of 2 that is at most a', we can simply combine a of these (b, x, y)-errors to get an (a, b, x, y)-error. Note that

$$\frac{abx}{y^2} \geq \frac{a'bx}{2y^2} \geq \frac{E}{400l^6}$$

so we have proved the claim. ■

In the statement of Lemma 18, we want to find an (a, b, x, y)-error of sufficient size where $y \leq b$. Note that so far we have shown how to find an (a, b, x, y)-error of sufficient size but have not dealt with the $y \leq b$ condition. The error that we found so far has a certain structure captured by the fact that we only needed to compare two consecutive permutations π_i, π_{i+1} in the dyadic decomposition of π. Call such an error an (a, b, x, y)-error *at level* i. Formally

Definition 19 *Say we have a matrix $M \in \mathsf{BISO}_{n \times n}$ and a permutation π on $[n]$. Let the dyadic decomposition of π be $(\pi_0, \ldots, \pi_l)$. We say the matrix M_π contains an (a, b, x, y)-error at level i if there are distinct indices $i_1, \ldots, i_a \in \{0, 1, \ldots, 2^i - 1\}$ such that for each $j \in [a]$, when comparing the matrices $R(M_{\pi_i}, D_{l-i,i_j})$ and $R(M_{\pi_{i+1}}, D_{l-i,i_j})$, the latter contains a (b, x, y) error (note that the rows of $R(M_{\pi_i}, D_{l-i,i_j})$ are sorted in increasing order).*

In order to show that we can find an (a, b, x, y)-error with $y \leq b$, we will argue that if when comparing π_i, π_{i+1}, the error we find does not satisfy $y \leq b$, then we can find an (a', b', x', y')-error of almost the same size at a significantly earlier level in the dyadic decomposition of π. Note that if we find an (a, b, x, y)-error at the first level of the dyadic decomposition with $\frac{abx}{y^2} \geq n$, we must have $a = 1$ and since clearly $x \leq n$, we conclude $y \leq b$.

Claim Assume M_π contains an (a, b, x, y)-error at level i with $\frac{abx}{y^2} \geq 4nf(n)$ and $y > b$. Then for $i' = \lceil i - \log f(n) \rceil$

$$||M_{\mathsf{id}} - M_{\pi_{i'}}||^2 \geq \frac{abx}{16y^2}$$

Proof If $i \leq \log f(n)$, then $a \leq 2^i \leq f(n)$. Clearly $x \leq n$ so if $y > b$ then $\frac{abx}{y^2} \leq 4nf(n)$. Thus, we only need to consider when $i \geq \log f(n)$.

Consider the dyadic blocks at level i', $D_{l-i',0}, \ldots, D_{l-i',2^{i'}-1}$.

$$||M_{\mathsf{id}} - M_{\pi_{i'}}||^2 = \sum_{j=0}^{2^{i'}-1} ||R(M_{\mathsf{id}}, D_{l-i',j}) - R(M_{\pi_{i'}}, D_{l-i',j})||^2$$

Now we will lower bound the terms on the right hand side. For each $0 \leq j \leq 2^{i'} - 1$, let M_j be the largest L^1 distance between two rows in $R(M_{\mathsf{id}}, D_{l-i',j})$. Note $M_0 + \cdots + M_{2^{i'}-1} \leq n$ since the rows of M_{id} are sorted in increasing order and the L^1 distance between any two rows is at most n.

Let T be the set of j such that $|M_j| \geq \frac{x}{2y}$. We have $|T| \leq \frac{2ny}{x}$. In the (a, b, x, y)-error at level i that we start with, we naturally have a disjoint (b, x, y)-errors. Call these *selected* errors. At most $\frac{2ny}{x}\frac{2^i}{2^{i'}} \leq \frac{2nyf(n)}{x}$ of the selected (b, x, y)-errors can be included in the dyadic blocks indexed by T. This is because there is at most one selected (b, x, y)-error within each dyadic block of size $\frac{n}{2^i}$ so there are at most $\frac{2^i}{2^{i'}}$ selected (b, x, y)-errors in each dyadic block of size $\frac{n}{2^{i'}}$. Thus, if we remove all dyadic blocks indexed by T, there must still be at least $a - \frac{2nyf(n)}{x}$ selected errors remaining. Combining $\frac{abx}{y^2} \geq 4nf(n)$ and $y > b$, we get $\frac{ax}{y} \geq 4nf(n)$. Thus $a - \frac{2nyf(n)}{x} \geq \frac{a}{2}$.

Next, consider a (b, x, y)-error in level i consisting of rows $(r_1, \ldots, r_b), (r'_1, \ldots, r'_b)$ contained in the dyadic block $D_{l-i,j}$. Say the dyadic block at level i' containing $D_{l-i,j}$ is $D_{l-i',j'}$ with $j' \notin T$. Let S be the set of indices indexing the locations of $r_1, \ldots, r_b, r'_1, \ldots, r'_b$ in the matrix $M_{\pi_{i'}}$. Note all elements of S are in $D_{l-i',j'}$ by definition. Now

$$||R(M_{\mathsf{id}}, S) - R(M_{\pi_{i'}}, S)||^2 \geq \frac{bx}{8y^2}$$

since the L^1 distance between the biggest and smallest rows in $R(M_{\mathsf{id}}, S)$ is at most $\frac{x}{2y}$ while there is a (x, y)-gap between all of $(r'_1, \ldots, r'_b)$ and all of $(r_1, \ldots, r_b)$. We can obtain similar inequalities for all (b, x, y)-errors that occur outside all of the dyadic blocks indexed by T and combine them to get

$$||M_{\mathsf{id}} - M_{\pi_{i'}}||^2 \geq \frac{abx}{16y^2}$$

■

Note we can iteratively apply Claim C.1 to find errors in earlier levels of the permutation decomposition of π. We will do this to complete the proof of Lemma 18. First note the following: Claim If M_π contains an (a_0, b_0, x_0, y_0)-error at level i_0 with $\frac{a_0 b_0 x_0}{y_0^2} \geq n l^{100 l^{0.5}}$ and $y_0 > b_0$ then M_π contains an (a_1, b_1, x_1, y_1)-error at some level $i_1 < i_0 - 100\sqrt{l}\log l$ with $\frac{a_1 b_1 x_1}{y_1^2} \geq \frac{a_0 b_0 x_0}{6400 y_0^2 l^6}$.
Proof Combining Claim C.1 and Claim C.1, we get that if M_π contains an (a_0, b_0, x_0, y_0)-error at level i_0 with $\frac{a_0 b_0 x_0}{y_0^2} \geq n l^{100 l^{0.5}}$ and $y_0 > b_0$ then $M_{\pi_{i_0'}}$ ($i_0' = i_0 - 100\sqrt{l}\log l$) contains a (a_1, b_1, x_1, y_1)-error at some level i_1 with $\frac{a_1 b_1 x_1}{y_1^2} \geq \frac{a_0 b_0 x_0}{6400 y_0^2 l^6}$. Note that we must have $i_1 < i_0'$ since the dyadic decomposition of $\pi_{i_0'}$ is constant at level i_0' and beyond. We conclude that M_π must contain the same (a_1, b_1, x_1, y_1)-error at level $i_1 < i_0 - 100\sqrt{l}\log l$. ■

Proof [Proof of Lemma 18] By Claim C.1, there must exist an (a_0, b_0, x_0, y_0)-error at some level i_0 with $\frac{a_0 b_0 x_0}{y_0^2} \geq \frac{E}{400 l^6}$. If $y_0 \leq b_0$ we are done. Otherwise, $y_0 > b_0$ and we can apply the above to find an (a_1, b_1, x_1, y_1)-error at level $i_1 < i_0 - 100\sqrt{l}\log l$. If $y_1 > b_1$ we can recurse again.

We can have at most $\frac{\sqrt{l}}{100}$ steps of recursion $i_0 \to i_1 \to \cdots \to i_j$ while maintaining that $i_j \geq 0$. Note it can easily be verified that the condition $\frac{a_j b_j x_j}{y_j^2} \geq n l^{100 l^{0.5}}$ is maintained at each step. Therefore, there must be some index $0 \leq g \leq \frac{\sqrt{l}}{100}$ such that $b_g \geq y_g$ and then M_π must contain a (a_g, b_g, x_g, y_g)-error satisfying

$$\frac{a_g b_g x_g}{y_g^2} \geq \frac{E}{(6400 l^6)^{\frac{\sqrt{l}}{10}}} \geq \frac{E}{l^{0.5}}$$

as desired. ■

Remark 20 *Note that the proof above actually allows us to slightly strengthen Lemma 18 to ensure that the (a, b, x, y)-error occurs at some level i in the dyadic decomposition.*

C.2. Finer Characterization of Difference Structures

Intuitively, Lemma 11 says that if two rows r and r' are well-sorted and $r > r'$, then the locations where the entry in r is larger than the corresponding entry in r' must "concentrate" together. The next result, Lemma 23, gives us a precise way to formulate this intuition when we consider a block of several rows simultaneously. This will be central to proving the correctness of our algorithms. First we make a few technical definitions.

Definition 21 *Let $M \in \mathsf{Perm}_{n\times n}$. We say a set of its rows $r_1 \ldots r_k$ contains a (c, x, y)-detectable difference if*

- *There exist two disjoint subsets of $\{i_1, \ldots, i_c\}, \{j_1, \ldots, j_c\} \subset \{1, 2, \ldots k\}$ such that $(r_{i_1}, \ldots, r_{i_c})$ and $(r_{j_1}, \ldots r_{j_c})$ form a (c, x, y)-difference*
- *There do not exist two disjoint subsets of $\{i_1', \ldots, i_c'\}, \{j_1', \ldots, j_c'\} \subset \{1, 2, \ldots k\}$ such that $(r_{i_1'}, \ldots, r_{i_c'})$ and $(r_{j_1'}, \ldots r_{j_c'})$ form a (c', x', y')-difference and $\frac{c' x'}{y'^2} \geq l^{0.5} \frac{cx}{y^2}$*
- $c \geq y$

- $\frac{cx}{y^2} \geq kl^{10^4 l^{0.5}}$
- $\frac{x}{y} \leq \sqrt{n} l^2$

Definition 22 *We define a (c, x, y)-strongly detectable difference in the same way as a (c, x, y)-detectable difference except we require that*

$$\frac{cx}{y^2} \geq kn^{\frac{1}{6}} l^{10^4 l^{0.5}}$$

Note that all strongly detectable differences are detectable.

Lemma 23 *Say we have a matrix $M_{\mathsf{id}} \in \mathsf{BISO}_{n \times n}$ and consider some subset of its rows $r_1, \ldots, r_k$ with $r_1 < \cdots < r_k$. Let c, x, y be powers of 2 such that $(r_{k-c+1}, \ldots, r_k)$ and $(r_1, \ldots r_c)$ form a (c, x, y)-detectable difference. For an integer I, consider dividing the columns into dyadic clusters of size 2^I,*

$$D_{I,0}, \ldots, D_{I,\frac{n}{2^I}-1}$$

There exists some integer I for which we can mark some subset of the dyadic clusters, say $D_{I,j_1}, \ldots, D_{I,j_e}$, such that the following properties hold.

- $\frac{x}{5yl^2} \leq 2^I \leq 200 l^{l^{0.5}} x$
- $e \leq 4y$
- *For each marked dyadic cluster, the mean of its two neighbors differ by at least $\frac{1}{10y}$. In other words for all $1 \leq i \leq e$, the following equation holds:*

$$|\mu(M, \{r_1, \ldots, r_k\}, D_{I,j_i-1}) - \mu(M, \{r_1, \ldots, r_k\}, D_{I,j_i+1})| \geq \frac{1}{10y}$$

  (We handle the case when the index $j_i \pm 1$ is out of bounds as usual. If $j_i - 1 < 0$, the mean is 0 and if $j_i + 1 \geq 2^I$, the mean is 1.)

- *Within each marked dyadic cluster $D_{I,j_b}, 1 \leq b \leq e$, highlight all of the columns such that the corresponding entries in r_c and r_{k-c+1} differ by at least $\frac{1}{y}$. The number of highlighted entries in each dyadic cluster is at least z where z is a power of 2 and satisfies $z \geq \frac{2^I}{800 l^{2+l^{0.5}}}$ and $ze \geq \frac{x}{600l^2}$.*
- *Let R be the matrix obtained by restricting M_{id} to the rows $r_{k-c+1}, \ldots, r_k, r_1, \ldots r_c$. Let j be an index corresponding to a highlighted column and say $j \in D_{I,j_b}$ where D_{I,j_b} is a marked dyadic cluster. Let j' be an index such that $j' \notin D_{I,j_b-1} \cup D_{I,j_b} \cup D_{I,j_b+1}$. Then the columns of R indexed by j and j' differ by $\frac{1}{2y}$ in at least $\frac{c}{2}$ different locations.*

Proof Define $S_d \subset [n]$ to be the set of locations where r_{k-c+1} is larger than r_c by at least d. Let $T_d \subset [n]$ be the set of locations where $r_{k-\lceil \frac{c}{2} \rceil + 1}$ is larger than $r_{\lceil \frac{c}{2} \rceil}$ by at least d. The set $T_{\frac{1}{4y}}$ can be broken down into a collection of "maximal" intervals. Say these are $[a_1, b_1), [a_2, b_2), \ldots, [a_m, b_m)$ (where $[a_i, b_i)$ denotes the set of locations $\{a_i, a_i + 1 \ldots, b_i - 1\}$) and $b_1 < a_2 < b_2 < a_3 < \cdots < b_{m-1} < a_m$.

For all i, let $s_i = b_i - a_i$ be length of the corresponding interval. Let $t_i = \left|S_{\frac{1}{y}} \cap [a_i, b_i)\right|$ be the number of elements of $S_{\frac{1}{y}}$ in the interval $[a_i, b_i)$. Note that maximality implies

$$|r_{k-c+1}^{(b_i)} - r_c^{(b_i)}| \leq |r_{k-\lceil \frac{c}{2} \rceil+1}^{(b_i)} - r_{\lceil \frac{c}{2} \rceil}^{(b_i)}| < \frac{1}{4y}$$

Also

$$x \leq t_1 + \cdots + t_m \leq s_1 + \cdots + s_m$$

For integers i, i', let $c_{i,i'}$ be the number of indices j such that $s_j \in [2^i, 2^{i+1})$ and $t_j \in [2^{i'}, 2^{i'+1})$. There must exist some I_0, I_0' such that

$$2^{I_0'} c_{I_0, I_0'} \geq \frac{x}{3l^2}$$

Let $[a_{i_1}, b_{i_1}), \ldots, [a_{i_g}, b_{i_g})$ with $g = c_{I_0, I_0'}$ be the intervals satisfying the condition $s_{i_j} \in [2^{I_0}, 2^{I_0+1})$ and $t_{i_j} \in [2^{I_0'}, 2^{I_0'+1})$. By Lemma 11, we must have $g \leq 4y$. This is because otherwise, we would be able to find $x_1 \in [a_{i_1}, b_{i_1}), \ldots, x_g \in [a_{i_g}, b_{i_g})$ such that $|r_{k-c+1}^{(x_i)} - r_c^{(x_i)}| \geq \frac{1}{y}$ and considering the subsequence $x_1, b_{i_1}, x_2, b_{i_2}, \ldots, x_g, b_{i_g}$ would give us a contradiction.

We will set $I = I_0 + 2$. Divide the columns into dyadic clusters of size 2^I. We will now perform the marking and highlighting process. Note that for each index $1 \leq j \leq g$, the interval $[a_{i_j}, b_{i_j})$ is either contained in a dyadic cluster of size 2^I or contained in the union of two consecutive dyadic clusters. If it is contained in a dyadic cluster, mark that cluster. If it is contained in the union of two consecutive dyadic clusters, mark the dyadic cluster (among the two) that contains more elements of the set $[a_{i_j}, b_{i_j}) \cap S_{\frac{1}{y}}$. It is fine if we mark a dyadic cluster multiple times. Now highlight the columns that are in a marked dyadic cluster and in $S_{\frac{1}{y}}$. We proceed to show that the desired conditions are satisfied.

- First, we bound the number of highlighted columns. Note that $\sum_{i=0}^{m} s_i \leq 33l^{l^{0.5}}x$ since otherwise we would have a $\lceil \frac{c}{2} \rceil, 33l^{l^{0.5}}x, 4y$ difference, contradicting the detectability of the (c, x, y)-difference.

  We also trivially must have $I_0 \geq I_0'$ so

  $$2^{I_0} \geq 2^{I_0'} \geq \frac{x}{3l^2 c_{I_0, I_0'}} \geq \frac{x}{3gl^2} \geq \frac{x}{20yl^2} \tag{2}$$

  Thus, $2^I \geq \frac{x}{5yl^2}$. Also note that none of the elements $s_1, \ldots, s_g$ can be larger than $33l^{l^{0.5}}x$ so $2^I \leq 200l^{l^{0.5}}x$. This completes the proof of the first property.

- The second property is clear since $g \leq 4y$ and we mark $e \leq g$ total dyadic clusters.

- Now we show that the means of the marked dyadic clusters differ from the means of their neighbors. Each marked dyadic cluster, say $D_{I,z}$, must correspond to some interval $[a_{i_j}, b_{i_j})$ in the

marking process. Note $D_{I,z}$ may correspond to multiple intervals in which case we let $[a_{i_j}, b_{i_j})$ be one of them.

Note that

$$r^{(a_{i_j}-1)}_{k-\lceil \frac{c}{2} \rceil+1} - r^{(a_{i_j}-1)}_{\lceil \frac{c}{2} \rceil} < \frac{1}{4y}$$
$$r^{(b_{i_j})}_{k-\lceil \frac{c}{2} \rceil+1} - r^{(b_{i_j})}_{\lceil \frac{c}{2} \rceil} < \frac{1}{4y}$$

and there must exist some integer $x_{i_j} \in [a_{i_j}, b_{i_j})$ such that

$$r^{(x_{i_j})}_{k-\lceil \frac{c}{2} \rceil+1} - r^{(x_{i_j})}_{\lceil \frac{c}{2} \rceil} \geq \frac{1}{y}$$

Since the rows have all of their entries in increasing order, we conclude

$$r^{(b_{i_j})}_{\lceil \frac{c}{2} \rceil} > \frac{1}{2y} + r^{(a_{i_j}-1)}_{k-\lceil \frac{c}{2} \rceil+1} \tag{3}$$

First we consider the case where $D_{I,z}$ contains the interval $[a_{i_j}, b_{i_j})$. We bound the difference between the means of $D_{I,z-1}$ and $D_{I,z+1}$. Let $N = R(M_{\mathsf{id}}, \{r_1, \ldots, r_k\})$ be the matrix formed by rows $r_1, \ldots, r_k$ and all n columns. Consider restricting N to each of blocks

$$\{r_1, \ldots, r_{\lceil \frac{c}{2} \rceil - 1}\}, \{r_{\lceil \frac{c}{2} \rceil}, \ldots, r_{k-\lceil \frac{c}{2} \rceil+1}\}, \{r_{k-\lceil \frac{c}{2} \rceil+2}, \ldots, r_k\}$$

and each of the dyadic clusters $D_{I,z-1}$ and $D_{I,z+1}$. Note that

$$\mu\left(N, \left(0, \lceil \frac{c}{2} \rceil - 1\right], D_{I,z-1}\right) \leq \mu\left(N, \left(0, \lceil \frac{c}{2} \rceil - 1\right], D_{I,z+1}\right)$$
$$\frac{1}{2y} + \mu\left(N, \left(\lceil \frac{c}{2} \rceil - 1, k - \lceil \frac{c}{2} \rceil + 1\right], D_{I,z-1}\right) \leq \mu\left(N, \left(\lceil \frac{c}{2} \rceil - 1, k - \lceil \frac{c}{2} \rceil + 1\right], D_{I,z+1}\right)$$
$$\mu\left(N, \left(k - \lceil \frac{c}{2} \rceil + 1, k\right], D_{I,z-1}\right) \leq \mu\left(N, \left(k - \lceil \frac{c}{2} \rceil + 1, k\right], D_{I,z+1}\right)$$

where the second inequality follows from (3) and the others are immediate. Also note that at least half of the entries of $R(N, (0, k], D_{I,z-1})$ and $R(N, (0, k], D_{I,z+1})$ are captured in the second inequality so we conclude

$$\mu(N, (0, k], D_{I,z+1}) - \mu(N, (0, k], D_{I,z-1}) \geq \frac{1}{4y}$$

Now consider the case where $D_{I,z}$ does not fully contain the interval $[a_{i_j}, b_{i_j})$. WLOG, the union of $D_{I,z}$ and $D_{I,z+1}$ contains $[a_{i_j}, b_{i_j})$. We can use essentially the same argument and note that since $b_{i_j} - a_{i_j} < 2^{I_0+1} = \frac{2^I}{2}$, then at least $\frac{1}{4}$ of the entries of $D_{I,z+1}$ are in the restriction $R\left(N, \left(\lceil \frac{c}{2} \rceil - 1, k - \lceil \frac{c}{2} \rceil + 1\right], \left(b_{i_j}, (z+1)\frac{n}{2^I}\right]\right)$ so

$$\frac{1}{4y} + \mu\left(N, \left(\lceil \frac{c}{2} \rceil - 1, k - \lceil \frac{c}{2} \rceil + 1\right], D_{I,z-1}\right) \leq \mu\left(N, \left(\lceil \frac{c}{2} \rceil - 1, k - \lceil \frac{c}{2} \rceil + 1\right], D_{I,z+1}\right)$$

In the end, we conclude

$$\mu(N, (0, k], D_{I,z+1}) - \mu(N, (0, k], D_{I,z-1}) \geq \frac{1}{8y}$$

This completes the proof of the third property.

- Note that
$$\frac{x}{3l^2} \leq 2^{I_0'} g \leq 2^{I_0} g \leq s_1 + \cdots + s_m \leq 33 l^{l^{0.5}} x$$
Thus,
$$2^{I_0'} \geq \frac{x}{3gl^2} \geq \frac{2^{I_0}}{100 l^{l^{0.5}+2}}$$
and clearly each marked dyadic cluster must contain at least $z = 2^{I_0'-1} \geq \frac{2^{I_0}}{200 l^{l^{0.5}+2}} \geq \frac{2^I}{800 l^{l^{0.5}+2}}$ highlighted columns since it must contain more than $\frac{t_{i_j}}{2}$ elements from some set of the form $\left|S_{\frac{1}{y}} \cap [a_{i_j}, b_{i_j})\right|$. Next, each marked dyadic cluster contains between z and $100z$ columns from the set
$$S' = S_{\frac{1}{y}} \cap \left([a_{i_1}, b_{i_1}) \cup \cdots \cup [a_{i_g}, b_{i_g})\right)$$
In total, all of the marked dyadic clusters must contain at least
$$\frac{t_{i_1} + \cdots + t_{i_g}}{2} \geq 2^{I_0'-1} g \geq \frac{x}{6l^2}$$
elements of S' so if e is the total number of marked dyadic clusters, we get
$$ze \geq \frac{x}{600 l^2}$$

- To prove the last property, we will start with the same observation as the third part. Each marked dyadic cluster, say $D_{I,z}$, must correspond to some interval $[a_{i_j}, b_{i_j})$ in the marking process. For any $x_{i_j} \in [a_{i_j}, b_{i_j})$ with x_{i_j} highlighted,
$$r^{(a_{i_j}-1)}_{k-\lceil \frac{c}{2} \rceil+1} - r^{(a_{i_j}-1)}_{\lceil \frac{c}{2} \rceil} < \frac{1}{4y}$$
$$r^{(x_{i_j})}_{k-c+1} - r^{(x_{i_j})}_{c} \geq \frac{1}{y}$$
$$r^{(b_{i_j})}_{k-\lceil \frac{c}{2} \rceil+1} - r^{(b_{i_j})}_{\lceil \frac{c}{2} \rceil} < \frac{1}{4y}$$
Since the entries are in increasing order in each row, the above implies that
$$r^{(b_{i_j})}_{\lceil \frac{c}{2} \rceil} - r^{(x_{i_j})}_{c} \geq \frac{3}{4y}$$
$$r^{(x_{i_j})}_{k-c+1} - r^{(a_{i_j}-1)}_{k-\lceil \frac{c}{2} \rceil+1} \geq \frac{3}{4y}$$
The union of $D_{I,z-1}, D_{I,z}, D_{I,z+1}$ contains the interval $[a_{i_j}, b_{i_j})$. Comparing the column indexed by x_{i_j} to any column with index smaller than $(z-1)2^I$, the entries in the rows $r_{k-c+1}, \ldots, r_{k-\lceil \frac{c}{2} \rceil+1}$ must differ by at least $\frac{3}{4y}$. Comparing the column indexed by x_{i_j} to any column with index larger than $(z+2)2^I$, the entries in rows $r_{\lceil \frac{c}{2} \rceil}, \ldots, r_c$ must differ by at least $\frac{3}{4y}$. This completes the proof of the final property.

■

Appendix D. 1D Multiscale Sort Analysis

We now analyze our algorithm in the case when the columns are perfectly sorted. Throughout this section, when we say pivoting algorithm, we are referring to the 1D PIVOTING ALGORITHM. We show that with high probability, the 1D MULTISCALE SORT algorithm successfully sorts the rows.

Theorem 24 *We have a base matrix $M_{id} \in BISO_{n\times n}$. Say the observed noisy matrix is $M'_{\eta,id}$. After running* 1D MULTISCALE SORT*, say the output is the matrix $M'_{\tau,id}$. Then with at least $1-\frac{1}{n^7}$ probability over the random noise*

$$||M_\tau - M_{id}||^2 \leq n^{1+o(1)}$$

Remark 25 *The precise bound we will show is*

$$||M_\tau - M_{id}||^2 \leq nl^{10^5 l^{0.5}}$$

The first important observation is that when running the 1D pivoting algorithm on a block that contains detectable differences, the total row variance among the remaining rows decreases by a non-negligible fraction at every step.

Lemma 26 *Say we have rows $r_1 < \cdots < r_k$ of the matrix M_{id} and let c, x, y be powers of 2 such that $(r_{k-c+1}, \ldots, r_k)$ and $(r_1, \ldots r_c)$ form a (c, x, y)-detectable difference. Recall that this means the following properties hold:*

- $c \geq y$
- $\frac{cx}{y^2} \geq kl^{10^4 l^{0.5}}$
- $\frac{x}{y} \leq \sqrt{n}l^2$

Let r'_i be rows obtained by adding noise to each entry of r_i. Let π be a permutation on $[k]$ and let A be a $k \times n$ matrix with rows $r'_{\pi(1)}, \ldots, r'_{\pi(k)}$ in order from bottom to top. Consider running the 1D pivoting algorithm on A with any index $0 \leq m \leq k$. Let $r'_{i_1}, \ldots r'_{i_m}$ be the set of rows that are not added to either the upper or lower set. Then with probability at least $1-\frac{1}{n^{10}}$ (over the random noise), we have

$$V(r_{i_1}, \ldots r_{i_m}) \leq \left(1 - \frac{1}{l^{10l^{0.5}}}\right) V(r_1, \ldots, r_k)$$

Let I be the integer that satisfies the properties of Lemma 23. Let $D_{I,j_1}, \ldots, D_{I,j_e}$ be the marked dyadic clusters according to Lemma 23. Throughout this proof we use B to denote the matrix with rows $r_{\pi(1)}, \ldots, r_{\pi(k)}$ in order from bottom to top (so B is A with the noise removed). We will actually show that running the pivoting algorithm with parameters $r = 2^I$ and $t = y$ will produce the desired result. Note that in the first step of the pivoting algorithm, we divide the columns into clusters of size r and form $k \times r$ matrices $A_1, \ldots, A_{\frac{n}{r}}$. Let $A_{h_1}, \ldots, A_{h_b}$ be the set of these matrices that contain (exactly) one of the marked dyadic clusters. Claim Let S be the preliminary set we construct when running the pivoting algorithm on A with parameters $r = 2^I$ and $t = y$. With at least $1 - \frac{1}{n^{20}}$ probability, $\{h_1, \ldots, h_b\} \subset S$ and $|S| \leq 1000y$.

Proof Let $B_1, \ldots, B_{\frac{n}{r}}$ be analogous to $A_1, \ldots, A_{\frac{n}{r}}$ except for the matrix B. By the third property of Lemma 23, the following holds for all $1 \leq i \leq b$: either $|\mu(B_{h_i}) - \mu(B_{h_i-1})| \geq \frac{1}{20y}$ or $|\mu(B_{h_i}) - \mu(B_{h_i+1})| \geq \frac{1}{20y}$ (since each of B_{h_i} contains one of the marked dyadic clusters). Next we claim that for any j, $|\mu(B_j) - \mu(A_j)| \leq \frac{1}{100y}$ with high probability. Note that the difference between the means of A_j and B_j is the mean of kr samples from the noise distribution. We have

$$kr \geq 2^I c \geq \frac{xc}{5yl^2} \geq \frac{1}{5} kyl^{100} \geq y^2 l^{10}$$

so this implies $|\mu(B_j) - \mu(A_j)| \leq \frac{1}{100y}$ with negligible failure probability. Thus with probability at least $1 - \frac{1}{n^{20}}$, either $|\mu(A_{h_i}) - \mu(A_{h_i-1})| \geq \frac{1}{30y}$ or $|\mu(A_{h_i}) - \mu(A_{h_i+1})| \geq \frac{1}{30y}$ for all $1 \leq i \leq b$.

The second clause in the claim follows from $|\mu(B_j) - \mu(A_j)| \leq \frac{1}{100y}$ and the assumption that the columns are perfectly sorted. ■

Claim Let v_{test} be the test vector we construct when running the 1D pivoting algorithm on A with parameters $r = 2^I$ and $t = y$. Let $s'_1, \ldots, s'_k$ be as defined in the pivoting algorithm. Let $s_1, \ldots, s_k$ be there denoised versions. With at least $1 - \frac{1}{n^{20}}$ probability

$$V(s_1 \cdot v_{\text{test}}, \ldots, s_k \cdot v_{\text{test}}) \geq 0.1bc \left(\frac{1}{1600yl^{2l^{0.5}}} \right)^2$$

Proof We analyze the step of the algorithm when we construct the columns c'_i. Let d_i, d'_i be the vectors analogous to c_i, c'_i except for the matrix B. First we analyze the vectors $d_{h_1}, \ldots, d_{h_b}, d'_{h_1}, \ldots, d'_{h_b}$. Note that by the fourth property in Lemma 23, for any $1 \leq i \leq b$, the entries $d_{h_1}^{(\pi^{-1}(1))}, \ldots, d_{h_1}^{(\pi^{-1}(c))}$ are all less than the entries $d_{h_1}^{(\pi^{-1}(k-c+1))}, \ldots, d_{h_1}^{(\pi^{-1}(k))}$ by at least

$$\frac{\frac{2^I}{800l^{0.5+2}} \frac{1}{y}}{r} \geq \frac{1}{800yl^{2l^{0.5}}}$$

This implies that

$$\left|\left| \frac{1}{\sqrt{b}} (d'_{h_1} + \cdots + d'_{h_b}) \right|\right|^2 \geq bc \left(\frac{1}{1600yl^{2l^{0.5}}} \right)^2$$

Note that $c_i = d_i + v$ where v is a vector whose entries are independently drawn from a sub-Gaussian distribution with mean 0 and sub-Gaussian parameter $\frac{1}{\sqrt{r}}$. Let C be the matrix with columns $c_i, i \in S$ and C' be the matrix with columns $c'_i, i \in S$. Define D, D' similarly. Note

$$C' = D' + V$$

where V is drawn from the following distribution

- Construct V_0, a $k \times |S|$ matrix with entries drawn independently and at random from the a sub-Gaussian distribution with mean 0 and sub-Gaussian parameter $\frac{1}{\sqrt{r}}$
- For each column of V_0, subtract the mean of the entries in that column from each entry.

With all but negligible probability (see Rudelson and Vershynin (2010)), the largest singular value of V_0 is at most

$$10l\sqrt{\frac{k}{r}}\left(1+\sqrt{\frac{|S|}{k}}\right) \leq 10l\sqrt{\frac{k}{r}}\left(1+\sqrt{\frac{1000y}{c}}\right) \leq 500l\sqrt{\frac{k}{r}}$$

Note that $(V - V_0)$ is a rank-1 matrix and it's largest singular value is at most the largest singular value of V_0. Therefore, with all but negligible probability, the largest singular value of V is at most $1000l\sqrt{\frac{k}{r}}$.

By Claim D, with all but negligible probability $h_1, \ldots, h_b \in S$. Also if we let u be the vector with $\frac{1}{\sqrt{b}}$ in entries indexed by $h_1, \ldots, h_b$ and 0 everywhere else we get that with all but negligible probability

$$||C'u||_2 \geq ||D'u||_2 - ||Vu||_2 \geq \sqrt{bc}\left(\frac{1}{1600yl^{2l^{0.5}}}\right) - 1000l\sqrt{\frac{k}{r}}$$

However, note that

$$\sqrt{bc}\left(\frac{1}{1600yl^{2l^{0.5}}}\right) \geq 10^8 l\sqrt{\frac{k}{r}} \tag{4}$$

since $r = 2^I$, $2^I b \geq ze \geq \frac{x}{600l^2}$ by the fourth clause of Lemma 23, and $\frac{cx}{y^2} \geq kl^{10^4 l^{0.5}}$ by assumption. Thus, we know that with all but negligible probability, the vector v that we obtain in the pivoting algorithm satisfies

$$||C'v||_2 \geq ||C'u||_2 \geq 0.9\sqrt{bc}\left(\frac{1}{1600yl^{2l^{0.5}}}\right)$$

Thus, with all but negligible probability

$$||D'v||_2 \geq ||C'v||_2 - ||Vv||_2 \geq 0.8\sqrt{bc}\left(\frac{1}{1600yl^{2l^{0.5}}}\right)$$

and therefore

$$||D'v_{\text{test}}||_2 \geq 0.4\sqrt{bc}\left(\frac{1}{1600yl^{2l^{0.5}}}\right)$$

Since the rows of D' are exactly $s_1, \ldots, s_k$ and the mean of the entries in each column is 0, the above immediately implies the desired conclusion ■

Proof [Proof of Lemma 26] Now we will complete the proof of Lemma 26. Note that regardless of the pivot, we have

$$|s'_{i_b} \cdot v_{\text{test}} - s'_{i_a} \cdot v_{\text{test}}| \leq \frac{20l}{\sqrt{r}}$$

for all $1 \leq a, b \leq m$.

Next, with at least $1 - \frac{1}{n^{20}}$ probability, for all $i \in \{1, 2, \ldots, k\}$,

$$|s'_i \cdot v_{\text{test}} - s_i \cdot v_{\text{test}}| \leq \frac{10l}{\sqrt{r}}$$

which implies

$$|s_{i_b} \cdot v_{\text{test}} - s_{i_a} \cdot v_{\text{test}}| \leq \frac{40l}{\sqrt{r}}$$

Thus

$$V(s_{i_1} \cdot v_{\text{test}}, \ldots, s_{i_m} \cdot v_{\text{test}}) \leq \frac{1600l^2 m}{r} \leq \frac{1600kl^2}{r} \tag{5}$$

Next note

$$\begin{aligned}
&V(s_1 \cdot v_{\text{test}}, \ldots, s_k \cdot v_{\text{test}}) - V(s_{i_1} \cdot v_{\text{test}}, \ldots, s_{i_m} \cdot v_{\text{test}}) \\
=&\sum_{j=1}^{k} \left(\left(s_j - \frac{s_1 + \cdots + s_k}{k} \right) \cdot v_{\text{test}} \right)^2 - \sum_{a=1}^{m} \left(\left(s_{i_a} - \frac{s_{i_1} + \cdots + s_{i_m}}{m} \right) \cdot v_{\text{test}} \right)^2 \\
=&\sum_{\substack{1 \leq j \leq k \\ j \notin \{i_1, \ldots, i_m\}}} \left(\left(s_j - \frac{s_1 + \cdots + s_k}{k} \right) \cdot v_{\text{test}} \right)^2 + m \left(\left(\frac{s_1 + \cdots + s_k}{k} - \frac{s_{i_1} + \cdots + s_{i_m}}{m} \right) \cdot v_{\text{test}} \right)^2 \\
\leq &\sum_{\substack{1 \leq j \leq k \\ j \notin \{i_1, \ldots, i_m\}}} \left\| s_j - \frac{s_1 + \cdots + s_k}{k} \right\|^2 + m \left\| \frac{s_1 + \cdots + s_k}{k} - \frac{s_{i_1} + \cdots + s_{i_m}}{m} \right\|^2 \\
=&V(s_1, \ldots, s_k) - V(s_{i_1}, \ldots, s_{i_m}) \leq \frac{1}{r} \left(V(r_1, \ldots, r_k) - V(r_{i_1}, \ldots, r_{i_m}) \right)
\end{aligned}$$

Next by Claim D and (5), we have with at least $1 - \frac{1}{n^{20}}$ probability

$$\begin{aligned}
&V(s_1 \cdot v_{\text{test}}, \ldots, s_k \cdot v_{\text{test}}) - V(s_{i_1} \cdot v_{\text{test}}, \ldots, s_{i_m} \cdot v_{\text{test}}) \\
\geq 0.1bc &\left(\frac{1}{1600yl^{2l^{0.5}}} \right)^2 - \frac{1600kl^2}{r} \geq 0.05bc \left(\frac{1}{1600yl^{2l^{0.5}}} \right)^2
\end{aligned}$$

where the last step follows from (4) in the proof of Claim D. Also, by Lemma 16 and the assumption that the (c, x, y)-difference is detectable we deduce

$$\frac{cx}{y^2} \geq \frac{V(r_1, \ldots, r_k)}{32l^{l^{0.5}+2}}$$

Combining everything, we have

$$V(r_1, \ldots, r_k) - V(r_{i_1}, \ldots, r_{i_m}) \geq rbc \frac{1}{10^8 y^2 l^{4l^{0.5}}} \geq \frac{xc}{600l^2} \frac{1}{10^8 y^2 l^{4l^{0.5}}} \geq \frac{V(r_1, \ldots, r_k)}{l^{10l^{0.5}}}$$

which immediately rearranges into the desired. ■

We need one more simple observation about the behavior of the pivoting algorithm before we can complete the analysis of the 1D MULTISCALE SORT algorithm. Claim Let $M \in \mathsf{Perm}_{n \times n}$ and let $r_1, \ldots, r_k$ be a subset of its rows. Let A be the matrix with rows $r'_1, \ldots, r'_k$, obtained by adding entrywise noise to $r_1, \ldots, r_k$. After running one iteration of the 1D pivoting algorithm on A, say the remaining rows are $r'_{j_1}, \ldots, r'_{j_d}$. With probability at least $1 - \frac{1}{n^{10}}$ over the random noise, for any indices $1 \leq a < b \leq d$,

$$||r_{j_a} - r_{j_b}||_1 \leq 100\sqrt{n}l$$

Proof Consider running the pivoting algorithm with parameters $r = n$ and $t = 1$. In this case we have $v = 1$ and the matrix N we construct for the pivoting step is simply a $k \times 1$ column vector containing the mean of the entries in each row. Regardless of the pivot row, say r_p, all rows r_i satisfying $|\mu(r_i') - \mu(r_p')| \geq \frac{10l}{\sqrt{n}}$ will be added to either the upper or lower set. This means for any indices $1 \leq a < b \leq d$,

$$|\mu(r_{j_a}') - \mu(r_{j_b}')| \leq \frac{20l}{\sqrt{n}}$$

Note $|\mu(r_i') - \mu(r_i)| \leq \frac{10l}{\sqrt{n}}$ for all i with at least $1 - \frac{1}{n^{10}}$ probability and $|\mu(r_i) - \mu(r_j)| = \frac{1}{n}||r_i - r_j||_1$ (since $M \in \mathsf{Perm}_{n\times n}$). Combining these with the above we get the desired inequality. ■

Now we are ready to complete the analysis of our algorithm when the columns are sorted.

Proof [Proof of Theorem 24] Let $E = nl^{10^5 l^{0.5}}$. Consider the dyadic decomposition of τ into $\tau_0, \tau_1, \ldots, \tau_l$. By Lemma 18, it suffices to show that our algorithm will not create any (a, b, x, y)-errors with $b \geq y$ and $\frac{abx}{y^2} \geq \frac{E}{l^{0.5}}$ at any level in the dyadic decomposition.

First we claim that the pivoting algorithm will not make any errors (i.e. any rows added to the lower set are among the d smallest rows and any rows added to the upper set are among the $k - d$ largest) as long as for any parameters r, t and rows s_i, s_i', the corresponding test vector v_{test} satisfies

$$|s_i' \cdot v_{\text{test}} - s_i \cdot v_{\text{test}}| < \frac{5l}{\sqrt{r}}$$

Assume that the block we are considering consists of the rows $r_1, \ldots, r_k$ and λ is a permutation on $\{1, 2, \ldots, k\}$ such that $s_{\lambda(1)}' \cdot v_{\text{test}} \leq \cdots \leq s_{\lambda(k)}' \cdot v_{\text{test}}$. Say that the pivot index is d. If some row r_i' is added to the lower set, we must have $s_i' \cdot v_{\text{test}} \leq s_{\lambda(d+1)}' \cdot v_{\text{test}} - \frac{10l}{\sqrt{r}}$. If this is an "error", then there must be some row r_j' among $r_{\lambda(d+1)}', \ldots, r_{\lambda(k)}'$ that should be in the lower set. However since v_{test} has all non-negative coordinates and $s_{\lambda(1)}' \cdot v_{\text{test}} \leq \cdots \leq s_{\lambda(k)}' \cdot v_{\text{test}}$, we must have

$$s_i' \cdot v_{\text{test}} \leq s_{\lambda(d+1)}' \cdot v_{\text{test}} - \frac{10l}{\sqrt{r}} \leq s_j' \cdot v_{\text{test}} - \frac{10l}{\sqrt{r}}$$
$$s_j \cdot v_{\text{test}} \leq s_i \cdot v_{\text{test}}$$

which cannot happen unless $|s_i' \cdot v_{\text{test}} - s_i \cdot v_{\text{test}}| \geq \frac{5l}{\sqrt{r}}$ or $|s_j' \cdot v_{\text{test}} - s_j \cdot v_{\text{test}}| \geq \frac{5l}{\sqrt{r}}$. We can use the same argument to deal with the case when some row is added to the upper set. However, for fixed v_{test} and index i, with all but negligible probability, $|s_i' \cdot v_{\text{test}} - s_i \cdot v_{\text{test}}| < \frac{5l}{\sqrt{r}}$. Union bounding, we conclude that the probability our pivoting algorithm makes an error is negligible. Note that here we use the fact that we resample the random noise *after* constructing v_{test}.

Now assume for the sake of contradiction that an (a, b, x, y)-error with $\frac{abx}{y^2} \geq \frac{E}{l^{0.5}}$ and $b \geq y$ occurs at level i in the dyadic decomposition with i minimal. This means that there exist dyadic blocks $D_{l-i,j_1}, \ldots, D_{l-i,j_a}$ such that for any $1 \leq c \leq a$, $R(M_{\tau_{i+1}}, D_{l-i,j_c})$ contains a (b, x, y) error. Now fix an index c with $1 \leq c \leq a$. Say the set of rows in $R(M_{\tau_i}, D_{l-i,j_c})$ is $u_1, \ldots, u_{\frac{n}{2^i}}$. After i phases of the full row sorting algorithm, say the matrix we are working with is M_{λ_i}'. The rows in $R(M_{\lambda_i}, D_{l-i,j_c})$ are exactly $u_1, \ldots, u_{\frac{n}{2^i}}$ up to some permutation. In the $i + 1^{\text{st}}$ phase of the full row sorting algorithm, we run the block sorting algorithm on the block $R(M_{\lambda_i}', D_{l-i,j_c})$.

The output of the block sorting algorithm determines the sets of rows in $R(M_{\tau_{i+1}}, D_{l-i-1,2j_c})$ and $R(M_{\tau_{i+1}}, D_{l-i-1,2j_c+1})$. Since we showed that with high probability, the pivoting algorithm will not make any errors, the only way for there to be a (b, x, y)-error is when we arbitrarily split the remaining rows after running $l^{l^{0.51}}$ iterations of the pivoting algorithm. Say the (b, x, y) error consists of the rows $s_1, \ldots, s_b, t_1, \ldots, t_b$. Let $s'_1, \ldots, s'_b, t'_1, \ldots, t'_b$ denote the respective rows with noise added. Note that $V(u_1, \ldots, u_{\frac{n}{2^i}}) \leq n^2$ so if the hypotheses of Lemma 26 are satisfied each time we run the pivoting algorithm, then with at least $1 - \frac{1}{n^9}$ probability

$$V(s_1, \ldots, s_b, t_1, \ldots, t_b) \leq \left(1 - \frac{1}{l^{10l^{0.5}}}\right)^{l^{l^{0.51}}} V(u_1, \ldots, u_{\frac{n}{2^i}}) < 1$$

Otherwise, there must be at least $l^{l^{0.5}}$ steps when we run the pivoting algorithm and the set of remaining rows does not contain a (c', x', y')-detectable difference for any c', x', y'. Note that $s_1, \ldots, s_b, t_1, \ldots, t_b$ form a (b, x, y)-difference and must be among the remaining rows. If k_t is the number of remaining rows after t iterations of the pivoting algorithm, $k_t \leq \frac{n}{2^i}$ for all t. Also $a \leq 2^i$ and $\frac{abx}{y^2} \geq \frac{E}{l^{0.5}}$ implies

$$\frac{bx}{y^2} \geq k_t \frac{E}{nl^{l^{0.5}}} \geq k_t l^{9 \cdot 10^4 l^{0.5}}$$

Claim D implies that after the first iteration of the pivoting algorithm, there cannot be any (c', x', y')-differences with $\frac{x'}{y'} > \sqrt{n}l^2$. Therefore, the only way for the hypotheses of Lemma 26 to fail is for there to be a (c', x', y')-difference with $c' < y'$ and $\frac{c'x'}{y'^2} \geq l^{l^{0.5}} \frac{bx}{y^2}$.

Note that there are at most l^3 possible triples (c', x', y'). Applying the above argument on each of the dyadic blocks $D_{i,j_1}, \ldots, D_{i,j_a}$, we get that for some (c', x', y') there are at least $\frac{a}{l^3}$ dyadic blocks $D_{i,j_c}, 1 \leq c \leq a$ such that $R(M_{\tau_i}, D_{i,j_c})$ contains a (c', x', y')-difference. We construct a permutation γ on $\{1, 2, \ldots n\}$ as follows: start with τ_i and then for each of the (c', x', y')-differences, "flip" the two sets of c' rows to create a (c', x', y') error. Note that the first i levels of the dyadic decomposition of γ agree with the dyadic decomposition of τ i.e. $(\gamma_0, \ldots, \gamma_i) = (\tau_0, \ldots, \tau_i)$. Also M_γ contains an $\left(\frac{a}{l^3}, c', x', y'\right)$-error at level i. Note

$$\frac{ac'x'}{l^3 y'^2} \geq \frac{abx}{y^2} l^{l^{0.5-3}}$$

Since $c' < y'$, we can iteratively apply Claim C.1, similar to the method at the end of Section C.1, to find an (a'', b'', x'', y'')-error with $\frac{a''b''x''}{y''^2} \geq \frac{E}{l^{0.5}}$ and $b'' \geq y''$ at some level $i' < i$, contradicting the minimality of i. ■

Appendix E. 2D Multiscale Sort Analysis

We now analyze our 2D MULTISCALE SORT algorithm when the columns may not be perfectly sorted. Throughout this section, when we say pivoting algorithm, we are referring to the 2D PIVOTING ALGORITHM. Our main theorem is stated below.

Theorem 27 *Assume we observe a matrix $M'_{\eta,\gamma} = M_{\eta,\gamma} + E$ where $M \in \mathsf{BISO}_{n\times n}$ and E is drawn from the error distribution. When we run the* 2D MULTISCALE SORT *on $M'_{\eta,\gamma}$, then with at least $1 - \frac{1}{n^{10}}$ probability, the final result will be a matrix $M'_{\pi,\sigma}$ such that*

$$||M_{\pi,\mathsf{id}} - M||_2^2 \leq n^{\frac{7}{6}+o(1)}$$
$$||M_{\mathsf{id},\sigma} - M||_2^2 \leq n^{\frac{7}{6}+o(1)}$$

Remark 28 *The precise bounds that we will show are*

$$||M_{\pi,\mathsf{id}} - M||_2^2 \leq n^{\frac{7}{6}+10^5\frac{\log\log n}{\sqrt{\log n}}}$$

and similar for $||M_{\mathsf{id},\sigma} - M||_2^2$.

At a high level, we will show that if when sorting the rows, our algorithm makes an (a, b, x, y)-error, then there must be an (a', b', x', y')-error in the column permutation that either has significantly larger size or has almost the same size and $x' >> x$. This will then allow us to conclude that at each iteration, our algorithm makes "progress" and eventually corrects all relevant (a, b, x, y)-errors.

Definition 29 *Say we run our 2D pivoting algorithm on a matrix containing the noisy rows $r'_1, \ldots, r'_{2c}$ such that their non-noisy versions satisfy $r_1 < \cdots < r_{2c}$ and $(r_1, \ldots, r_c)$ and $(r_{c+1}, \ldots, r_{2c})$ form a (c, x, y)-difference. We say that the algorithm resolves the difference if either all of $(r_1, \ldots, r_c)$ are added to the lower set or all of $(r_{c+1}, \ldots, r_{2c})$ are added to the upper set.*

The first key result is stated below. Claim We have a matrix $M_{\eta,\gamma}$ and its noisy version $M'_{\eta,\gamma}$. Consider running the block sorting algorithm on $M'_{\eta,\gamma}$ on all dyadic blocks of size $\frac{n}{2^i}$ for some i. Note that this involves many iterations of running the 2D pivoting algorithm on various blocks.

Assume that with non-negligible probability, there exist disjoint blocks $B_1, \ldots, B_a$ of sizes $k_1, \ldots, k_a$ respectively and parameters (c, x, y) such that for each block $B_1, \ldots, B_a$, the 2D pivoting algorithm failed to resolve some (c, x, y)-strongly detectable difference when run on it. Also assume $\frac{acx}{y^2} \geq n^{\frac{7}{6}} l^{10^4 l^{0.5}}$. Then one of the following must hold

- $||M_{\eta,\gamma} - M_{\eta,\mathsf{id}}||_2^2 \geq \frac{acx}{y^2} l^{5000 l^{0.5}}$
- There exists some i' such that in the column permutation γ, there exists an (a', b', x', y')-error at level i' with

$$\frac{a'b'x'}{y'} \geq \frac{1}{10^{20} l^{30}} \frac{acx}{y^2}$$
$$x' \geq l^{4900 l^{0.5}} x$$

Remark 30 *To simplify the explanation, throughout the proof we will assume that the blocks $B_1, \ldots, B_a$ and (c, x, y)-strongly detectable differences within each block are fixed ahead of time. Following the exact same method (and noting that we only run the pivoting algorithm polynomially many times), we can prove that the statement holds simultaneously for all blocks on which we run the 2D pivoting algorithm and all (c, x, y)-strongly detectable differences.*

The proof consists of several steps. First we will "regularize" the structure of the (c, x, y) differences in each block. We will then analyze the possible failure modes of the pivoting algorithm and show that if the differences are not resolved with high probability then the column permutation γ must have certain structural properties. Finally, we will use these structural properties to show that the conditions of Claim E are satisfied.

E.1. Regularization Step

First if for at least $\frac{a}{2}$ of the blocks, $B_1, \ldots, B_a$ we have

$$||R(M_{\eta,\gamma}, B_i) - R(M_{\eta,\mathsf{id}}, B_i)||_2^2 \geq \frac{cx}{y^2} l^{7000l^{0.5}}$$

then we immediately deduce

$$||M_{\eta,\gamma} - M_{\eta,\mathsf{id}}||_2^2 \geq \frac{a}{2} \cdot \frac{cx}{y^2} l^{7000l^{0.5}} \geq \frac{acx}{y^2} l^{6900l^{0.5}}$$

Otherwise, for at least $\frac{a}{2}$ of the blocks (WLOG $B_1, \ldots, B_{\frac{a}{2}}$)

$$||R(M_{\eta,\gamma}, B_i) - R(M_{\eta,\mathsf{id}}, B_i)||_2^2 < \frac{cx}{y^2} l^{7000l^{0.5}} \tag{6}$$

For each of these blocks, apply the marking and highlighting described in Claim 23 on the (c, x, y)-detectable difference. There are at most l^2 distinct possible values for the parameters (I, z) so at least $\frac{a}{2l^2}$ of the blocks (WLOG $B_1, \ldots, B_{\frac{a}{2l^2}}$), share the same parameters I, z. Set

$$r = \frac{2^I x}{y^2 n^{\frac{1}{6}} l^{2000l^{0.5}}}$$

Now we will first modify the marking and highlighting scheme. First, for each $i \in [\frac{a}{2l^2}]$ and each marked dyadic cluster $D_{I,j}$ in $R(M_{\eta,\mathsf{id}}, B_i)$, mark the corresponding submatrix $R(M_{\eta,\mathsf{id}}, B_i, D_{I,j})$ of $M_{\eta,\mathsf{id}}$. For each $0 \leq g \leq \frac{n}{r} - 1$ and index $i \in [\frac{a}{2l^2}]$, let $W_{g,i}$ be the number of marked submatrices in $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$.

Let W be a parameter that will be chosen later. For each index i, let $C_{i,W}$ be the number of indices $g \in \{0, 1, \ldots, \frac{n}{r} - 1\}$ such that $W_{g,i} \geq W$. For each g, i, if $W_{g,i} < W$, unmark all submatrices in $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$. Otherwise, arbitrarily choose W of the marked submatrices in $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ to keep marked and unmark the rest. Claim There exists a choice of W such that W is a power of 2 and for at least $\frac{a}{4l^3}$ of the indices $i \in [\frac{a}{2l^2}]$,

$$zWC_{i,W} \geq \frac{x}{10^4 l^3}$$

Proof For each index $i \in [\frac{a}{2l^2}]$ let e_i be the number of marked dyadic clusters in the *original* marking. Since

$$W_{0,i} + \cdots + W_{\frac{n}{r}-1,i} = e_i$$

we can use the standard argument about dividing into scales and find some parameter $W(i)$ that is a power of 2 such that $W(i)C_{i,W(i)} \geq \frac{e_i}{2l}$. Over all $i \in [\frac{a}{2l^2}]$, there must be some parameter W such that $W = W(i)$ for at least $\frac{a}{4l^3}$ distinct values of i. For these indices we have

$$zWC_{i,W} \geq \frac{ze_i}{2l} \geq \frac{x}{10^4 l^3}$$

■

WLOG for all $i \in \{1, 2, \ldots, \frac{a}{4l^3}\}$, the statement of the above claim is satisfied. Restrict i to this set from now on. We will now find a subset $G \subset \{0, 1, \ldots, \frac{n}{r} - 1\}$ such that for all $g \in G$, the interval $\mathsf{In}_{r,g}$ contains many marked submatrices. Claim There exists a subset $G \subset \{0, 1, \ldots, \frac{n}{r} - 1\}$ and a parameter h such that h is a power of 2 and the following properties hold

- $h|G|Wz \geq \frac{ax}{(10l)^8}$
- For each $g \in G$, the submatrix $R(M_{\eta,\mathsf{id}}, B_1 \cup \cdots \cup B_{\frac{a}{4l^3}}, \mathsf{In}_{r,g})$ must contain at least hW marked submatrices. Note each submatrix $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ either contains 0 or W marked submatrices.

Proof For each element $g \in \{0, 1, \ldots, \frac{n}{r} - 1\}$, let H_g be the number of marked submatrices in

$$R(M_{\eta,\mathsf{id}}, B_1 \cup \cdots \cup B_{\frac{a}{4l^3}}, \mathsf{In}_{r,g})$$

Let D_H be the number of indices $g \in \{0, 1, \ldots, \frac{n}{r} - 1\}$ such that $H_g \geq H$. Using a standard argument, there exists some parameter H that is a power of 2 such that

$$H \cdot D_H \geq \frac{H_0 + \cdots + H_{\frac{n}{r}-1}}{2l} = \frac{W(C_{1,W} + \cdots + C_{\frac{a}{4l^3},W})}{2l}$$

Let $h = \frac{H}{W}$ and G be the set of all $g \in \{0, 1, \ldots, \frac{n}{r} - 1\}$ such that $H_g \geq H$.

$$h|G|Wz = hD_HWz = HD_Hz \geq \frac{zW(C_{1,W} + \cdots + C_{\frac{a}{4l^3},W})}{2l} \geq \frac{1}{2l} \cdot \frac{a}{4l^3} \cdot \frac{x}{10^4 l^3} \geq \frac{ax}{(10l)^8}$$

The second condition is clear by the way we defined h and G. ■

Now for some $i \in [\frac{a}{4l^3}], g \in G$, we can unmark all submatrices in $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ so that for each $g \in G$, the submatrix $R(M_{\eta,\mathsf{id}}, B_1 \cup \cdots \cup B_{\frac{a}{4l^3}}, \mathsf{In}_{r,g})$ contains exactly hW marked submatrices. To summarize the properties of the final marking scheme

- Each marked submatrix $R(M_{\eta,\mathsf{id}}, B_i, D_{I,j})$ contains exactly z highlighted columns with respect to the (c, x, y)-difference in $R(M_{\eta,\mathsf{id}}, B_i)$ (note we simply unhighlight some columns within each submatrix to ensure that exactly z columns are highlighted).
- For each $i \in [\frac{a}{4l^3}], g \in \{0, 1, \ldots, \frac{n}{r} - 1\}$, the submatrix $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ contains either 0 or W marked submatrices
- For each $g \in G$, the submatrix $R(M_{\eta,\mathsf{id}}, B_1 \cup \cdots \cup B_{\frac{a}{4l^3}}, \mathsf{In}_{r,g})$ contains exactly hW marked submatrices
- $h|G|Wz \geq \frac{ax}{(10l)^8}$

E.2. 2D Pivoting Algorithm Analysis

For $i \in [\frac{a}{4l^3}]$ let Z_i be the number of marked submatrices in $R(M_{\eta,\mathsf{id}}, B_i, \bigcup_{g\in G} \mathsf{ln}_{r,g})$. Let Y be the set of i such that $Z_i \geq \frac{l^3}{a} h|G|W$.

Now we consider the behavior of the pivoting algorithm when run on a block $R(M'_{\eta,\gamma}, B_i)$ for each $i \in Y$. To simplify notation, we will fix $i = 1$ (assume $1 \in Y$ as clearly Y is nonempty) and let $k = k_1$ be the size of the block B_1. Say $R(M_{\eta,\mathsf{id}}, B_1)$ consists of rows $r_1, \ldots, r_k$ such that $r_1 < \cdots < r_k$ and $(r_1, \ldots, r_c)$ and $(r_{k-c+1}, \ldots, r_k)$ form a (c, x, y)-strongly detectable difference. Recall the following conditions for a (c, x, y) difference to be strongly detectable.

1. The matrix $R(M_{\eta,\mathsf{id}}, B_1)$ does not contain a (c', x', y')-difference with $\frac{c'x'}{y'^2} \geq l^{l^{0.5}} \frac{cx}{y^2}$
2. $c \geq y$
3. $\frac{cx}{y^2} \geq kn^{\frac{1}{6}} l^{10^4 l^{0.5}}$
4. $\frac{x}{y} \leq \sqrt{n} l^2$

Let $h_1, \ldots, h_b \in G$ be the indices such that $R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,h_i})$ contains (exactly) W marked submatrices. Note

$$bW = Z_i \geq \frac{l^3}{a} h|G|W$$

We will analyze the possible failure modes of the pivoting algorithm. Let S_0 and S be the starter set and preliminary set that we construct when running the pivoting algorithm on $R(M'_{\eta,\gamma}, B_1)$ with parameters $r = \frac{2^I x}{y^2 n^{\frac{1}{6}} l^{2000 l^{0.5}}}$ and $t = y$. Claim The following conditions must hold with all but negligible probability

- $|S_0| \leq 300t$
- Let Q be the set of indices $1 \leq f \leq b$ such that $h_f \in S_0$. Then $|Q| \geq 0.99b$.

Proof First note that as g ranges over $\{0, \ldots, \frac{n}{r}\}$, we have

$$\left|\mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,g}\right) - \mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,g-1}\right)\right| \geq \frac{1}{50y}$$

for at most $50y$ values of g. If $|S_0| \geq 300t = 300y$, then there must be at least $100y$ values of g such that the following two conditions hold

$$\left|\mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,g}\right) - \mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,g-1}\right)\right| \leq \frac{1}{50y}$$
$$\left|\mu\left(R(M'_{\eta,\gamma}, B_1, \mathsf{ln}_{r,g}]\right) - \mu\left(R(M'_{\eta,\gamma}, B_1, \mathsf{ln}_{r,g-1}\right)\right| \geq \frac{1}{30y}$$

Since $k \geq c \geq y$ and

$$r = \frac{2^I x}{y^2 n^{1/6} l^{2000 l^{0.5}}} \geq \frac{x}{5yl^2} \frac{x}{y^2 n^{1/6} l^{2000 l^{0.5}}} \geq \frac{x}{5yl^2} \frac{k l^{7900 l^{0.5}}}{c} > y l^{7000 l^{0.5}}$$

we deduce that with all but negligible probability,

$$\left|\mu\left(R(M'_{\eta,\gamma}, B_1, \mathsf{ln}_{r,g}\right) - \mu\left(R(M_{\eta,\gamma}, B_1, \mathsf{ln}_{r,g}\right)\right| \leq \frac{1}{10000y} \tag{7}$$

for all $g \in \{0, \ldots, \frac{n}{r} - 1\}$. Thus, for at least $100y$ distinct values of g

$$\left|\mu\left(R(M_{\eta,\gamma}, B_1, \mathsf{ln}_{r,g}\right) - \mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,g}\right)\right| \geq \frac{1}{1000y}$$

This implies

$$||R(M_{\eta,\gamma}, B_1) - R(M_{\eta,\mathsf{id}}, B_1)||_2^2 \geq 100y\frac{kr}{(1000y)^2} \geq \frac{k}{10^4 y}\frac{x}{5yl^2}\frac{x}{y^2 n^{1/6} l^{2000l^{0.5}}} \geq \frac{kx}{y^2} l^{7000l^{0.5}} \geq \frac{cx}{y^2} l^{7000l^{0.5}}$$

which contradicts our assumption (6) about the block B_1.
Now we consider the second condition. Using the third clause of Lemma 23 and the properties of our marking scheme, we deduce that one of the following two conditions holds for all $1 \leq f \leq b$

$$\left|\mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,h_f}\right) - \mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,h_f+1}\right)\right| \geq \frac{W}{20y}$$
$$\left|\mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,h_f}\right) - \mu\left(R(M_{\eta,\mathsf{id}}, B_1, \mathsf{ln}_{r,h_f-1}\right)\right| \geq \frac{W}{20y}$$

Note we can construct a set $Q' \subset [b]\backslash Q$ such that $|Q'| \geq \frac{b-|Q|}{3}$ and for any $f_1, f_2 \in Q'$, $|h_{f_1} - h_{f_2}| \geq 3$. By the definition of our pivoting algorithm, for any $f \in Q'$

$$\left|\mu\left(R(M'_{\eta,\gamma}, B_1, \mathsf{ln}_{r,h_f}\right) - \mu\left(R(M'_{\eta,\gamma}, B_1, \mathsf{ln}_{r,h_f+1}\right)\right| \leq \frac{1}{30y}$$
$$\left|\mu\left(R(M'_{\eta,\gamma}, B_1, \mathsf{ln}_{r,h_f}\right) - \mu\left(R(M'_{\eta,\gamma}, B_1, \mathsf{ln}_{r,h_f-1}\right)\right| \leq \frac{1}{30y}$$

If $|Q| \leq 0.99b$ then using the above and (7), we deduce that

$$||R(M_{\eta,\gamma}, B_1) - R(M_{\eta,\mathsf{id}}, B_1)||_2^2 \geq kr|Q'|\frac{1}{4}\left(\frac{W}{20y} - \left(\frac{1}{30y} + \frac{1}{10000y}\right)\right)^2$$
$$\geq kr|Q'|\left(\frac{W}{10^5 y}\right)^2 \geq \frac{krbW}{10^{20}y^2} \geq \frac{1}{10^{20}} \cdot \frac{k2^I x}{y^4 n^{\frac{1}{6}} l^{2000l^{0.5}}} \cdot \frac{l^3 h|G|W}{a}$$
$$\geq \frac{kx}{10^{20} l^{2000l^{0.5}} n^{\frac{1}{6}} y^4} \cdot \frac{zW|G|hl^3}{a} \geq \frac{kx^2}{10^{20} l^{2100l^{0.5}} n^{\frac{1}{6}} y^4} \geq \frac{x}{10^{20} n^{\frac{1}{6}} l^{2100l^{0.5}} y^2} \cdot \frac{cx}{y^2} \geq \frac{cx}{y^2} l^{7000l^{0.5}}$$

which again is a contradiction. ■

Note that we highlight some subset of columns of $M_{\eta,\mathsf{id}}$, When we apply the permutation γ on the columns, the highlighted columns move to new locations. Naturally this gives us a way to highlight some subset of the columns of $M_{\eta,\gamma}$.

Definition 31 *We say a permutation γ moves a column c_i by at least r distance if when $\{1, 2, \ldots n\}$ is divided into intervals of length r, $\gamma^{-1}(i)$ and i are in non-neighboring intervals.*

Claim Assume that the starter set S_0 satisfies $|S_0| < 300t$ and $|Q| \geq 0.99b$ where Q is the set of indices $1 \leq f \leq b$ such that $h_f \in S_0$. One of the following must be true

- When applying the permutation γ to the columns of $R(M_{\eta,\mathsf{id}}, B_1)$, for at least $0.98b$ indices $f \in Q$, the following holds:
  - For at least $0.99W$ of the marked submatrices within $R(M_{\eta,\mathsf{id}}, B_1, \mathsf{In}_{r,h_f})$, the permutation γ moves at least $0.99z$ highlighted columns by a distance of at least r.
- There exists a parameter w such that when running the 2D pivoting algorithm with parameter w, with all but negligible probability
  - $|R| \leq \frac{x^2}{y^2 l^{410}}$
  - When considering the matrix $R(M_{\eta,\gamma}, B_1)$, the set of columns indexed by elements of R contains at least $\frac{x}{l^6}$ highlighted columns.

Proof We will follow the same outline as the proof of Claim D. The intuition is that if the permutation γ does not move the highlighted columns too far from their original locations, then the test set we construct is essentially the same as the one we would have constructed if the columns were fully sorted and will contain sufficient signal to separate the rows. We break into two cases depending on how large r is compared to x, y.

Case 1: $r \leq \frac{x^2}{10^{20} l^{420} y^3}$

Here we claim that if the first condition is not satisfied then any $w \geq 1000y$ will suffice. Note that $|S| < 1000y$ so $T = S$. If the first condition does not hold, the number of highlighted columns among $R\left(M_{\eta,\mathsf{id}}, B_1, \bigcup_{a \in S_0} \mathsf{In}_{r,a}\right)$ that are moved by a distance less than r is at least

$$\frac{zWb}{10^{10}} \geq \frac{l^3 zh|G|W}{10^{10}a} \geq \frac{x}{l^6}$$

However, these highlighted columns will all be in $R = \bigcup_{a \in S} \mathsf{In}_{r,a}$. Also note $|R| \leq 1000yr \leq \frac{x^2}{y^2 l^{410}}$.

Case 2: $r > \frac{x^2}{10^{20} l^{420} y^3}$

We first set up notation. Most variables are defined similarly to in the proof of Claim D.

- Let $A_1, \ldots, A_{\frac{n}{r}}$ be the $k \times r$ matrices that we construct in the first step when running the 2D pivoting algorithm on $R\left(M'_{\eta,\gamma}, B_1\right)$
- Let c_i, c'_i be the columns that we construct in Step 2 of the pivoting algorithm
- Define d_i, d'_i analogous to c_i, c'_i for the de-noised matrix $R\left(M_{\eta,\gamma}, B_1\right)$
- Define $v_i = c_i - d_i$, $v'_i = c'_i - d'_i$

If there are at least $\frac{2^I}{10^6 l^{0.5+2}}$ highlighted columns in A_f for some $1 \leq f \leq \frac{n}{r}$ then

$$||d'_f|| \geq 10^{-20}\sqrt{2c}\left(\frac{yn^{\frac{1}{6}} l^{1800 l^{0.5}}}{x}\right)$$

Next note that for all i, $||v_i|| \leq 10l\sqrt{\frac{k}{r}}$ with all but negligible probability and also $||v_i'|| \leq ||v_i||$. Therefore with all but negliglble probability

$$||d_i'|| - 10l\sqrt{\frac{k}{r}} \leq ||c_i'|| \leq ||d_i'|| + 10l\sqrt{\frac{k}{r}} \tag{8}$$

Now we claim,

$$10^{-40} \cdot 2c\left(\frac{yn^{\frac{1}{6}}l^{1800l^{0.5}}}{x}\right)^2 \geq 10^{10}\left(10l\sqrt{\frac{k}{r}}\right)^2 \tag{9}$$

The above rearranges into

$$\frac{2rcy^2n^{\frac{1}{3}}l^{3600l^{0.5}}}{x^2} \geq 10^{52}l^2k$$

However note that

$$\frac{2rcy^2n^{\frac{1}{3}}l^{3600l^{0.5}}}{x^2} \geq \frac{cn^{\frac{1}{3}}l^{3600l^{0.5}}}{10^{20}l^{420}y} \geq \frac{cx}{y^2}\frac{y}{x} \cdot n^{\frac{1}{3}}l^{l^{0.5}} \geq 10^{52}l^2k$$

where we used the assumption that the (c, x, y)-difference is strongly detectable in the last step.

Let C be the number of indices f such that

$$||c_f'|| \geq 0.9 \cdot 10^{-20} \cdot \sqrt{2c}\left(\frac{yn^{\frac{1}{6}}l^{1800l^{0.5}}}{x}\right)$$

We show that setting w to be the unique power of 2 between C and $2C$ suffices. By (8) and (9), to bound C, it suffices to upper bound the number of indices $1 \leq f \leq \frac{n}{r}$ such that

$$||d_f'|| \geq 0.8 \cdot 10^{-20} \cdot \sqrt{2c}\left(\frac{yn^{\frac{1}{6}}l^{1800l^{0.5}}}{x}\right)$$

Note

$$||d_1'||^2 + \cdots + ||d_{\frac{n}{r}}'||^2 \leq \frac{1}{r}V(\{r_1, \ldots, r_k\}) \leq 32l^{2l^{0.5}}\frac{cx}{ry^2}$$

where the second inequality follows from the second clause in the definition of (c, x, y)-detectable and Lemma 16. Therefore, with all but negligible probability

$$C \leq \frac{32l^{2l^{0.5}}\frac{cx}{ry^2}}{\left(0.8 \cdot 10^{-20} \cdot \sqrt{2c}\left(\frac{yn^{\frac{1}{6}}l^{1800l^{0.5}}}{x}\right)\right)^2} \leq \frac{10^{100}x^3}{y^4rn^{\frac{1}{3}}l^{3000l^{0.5}}}$$

Since

$$\frac{200x^2}{y^2n^{\frac{1}{6}}l^{1900l^{0.5}}} \geq r = \frac{2^Ix}{y^2n^{\frac{1}{6}}l^{2000l^{0.5}}} > \frac{x^2}{10^{20}l^{420}y^3}$$

we have

$$\frac{x}{y^2} \leq \sqrt{n}l^2\frac{1}{y} \leq \sqrt{n}l^2\left(\frac{10^{23}}{l^{1400l^{0.5}}n^{\frac{1}{6}}}\right) = \frac{10^{23}n^{\frac{1}{3}}}{l^{1400l^{0.5}}}$$

and thus

$$C \leq \frac{x^2}{ry^2} \cdot \frac{10^{100}x}{y^2 n^{\frac{1}{3}} l^{3000l^{0.5}}} \leq \frac{x^2}{ry^2 l^{4000l^{0.5}}}$$

Since, $|R| = rw \leq 2rC$ we deduce that $|R|$ is within the desired range. Now we analyze whether the set of columns indexed by R contains a sufficient number of highlighted columns.

If the first condition fails, then for at least $0.01b$ values of $f \in Q$, one of the submatrices

$$R(M_{\eta,\gamma}, B_1, \mathsf{In}_{r,h_f-1}), R(M_{\eta,\gamma}, B_1, \mathsf{In}_{r,h_f}), R(M_{\eta,\gamma}, B_1, \mathsf{In}_{r,h_f+1})$$

must contain at least $10^{-3}zW$ highlighted columns. Note,

$$10^{-3}zW \geq \frac{z}{10^3} \geq \frac{2^I}{10^6 l^{l^{0.5}+2}}$$

so with all but negligible probability, the index $i \in \{h_f - 1, h_f, h_f + 1\}$ such that $R(M_{\eta,\gamma}, B_1, \mathsf{In}_{r,i})$ contains at least $10^{-3}zW$ highlighted columns will be added to T. The total number of highlighted columns in R will be at least

$$\frac{1}{3} \cdot 10^{-3} \cdot zW(0.01)b \geq \frac{zWb}{10^5} \geq \frac{l^3 zh|G|W}{10^5 a} \geq \frac{x}{l^6}$$

(note the factor of $\frac{1}{3}$ is because we might count some index i up to 3 times) ■

E.3. Completing the Proof of Claim E

We will now combine the previous two claims and the assumption that the (c, x, y)-detectable difference in B_1 is not resolved by the pivoting algorithm with all but negligible probability. First, we make a simple observation that with all but negligible probability, the pivoting algorithm never incorrectly adds rows to the upper or lower set. Claim Say we run the pivoting algorithm with index $1 \leq d \leq k$ on a block consisting of rows $r'_1, \ldots, r'_k$, where their non-noisy versions, $r_1, \ldots, r_k$ satisfy $r_1 < \cdots < r_k$. Then with all but negligible probability, only rows among $r'_1, \ldots, r'_d$ are added to the lower set and only rows among $r'_{d+1}, \ldots, r'_k$ are added to the upper set.

Proof For each set of parameters r, t, w, let $R_{r,t,w}$ be the test set that is constructed when running the pivoting algorithm with parameters r, t, w. With all but negligible probability

$$|\sigma(r'_i, R_{r,t,w}) - \sigma(r_i, R_{r,t,w})| \leq 5l \tag{10}$$

for all parameters r, t, w and $1 \leq i \leq k$ (note here we use the fact that the we re-sample the noise *after* constructing the test set R). If the above is satisfied then we claim the pivoting algorithm will not make any errors. Fix r, t, w. Let λ be a permutation on $\{1, 2, \ldots, k\}$ such that

$$\sigma(r'_{\lambda(1)}, R_{r,t,w}) \leq \cdots \leq \sigma(r'_{\lambda(k)}, R_{r,t,w})$$

Now assume for the sake of contradiction that some row r'_i is incorrectly added to the lower set. Then there must be some row r'_j among $r'_{\lambda(d+1)}, \ldots, r'_{\lambda(k)}$ that should actually be in the lower set. Note since $r_j < r_i$ we have

$$\sigma(r'_i, R_{r,t,w}) \leq \sigma(r'_{\lambda(d+1)}, R_{r,t,w}) - 10l \leq \sigma(r'_j, R_{r,t,w}) - 10l$$
$$\sigma(r_j, R_{r,t,w}) \leq \sigma(r_i, R_{r,t,w})$$

However this contradicts (10). We conclude that with all but negligible probability, the pivoting algorithm does not incorrectly add any rows to the upper or lower set. ■

Claim For each $i \in Y$, let $b_i = \frac{Z_i}{W}$ be the number of indices $g \in G$ such that $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ contains W marked submatrices. When applying the permutation γ on the columns of $R(M_{\eta,\mathsf{id}}, B_i)$, for at least $0.98b_i$ indices $1 \leq f \leq b_i$, for at least $0.99W$ of the marked submatrices within $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,h_f})$, the permutation γ moves at least $0.99z$ highlighted columns by a distance of at least r.

Proof Let the rows in $R(M_{\eta,\mathsf{id}}, B_i)$ be $r_1, \ldots, r_k$ such that $(r_{i_1}, \ldots, r_{i_c})$ and $(r_{j_1}, \ldots, r_{j_c})$ form a (c, x, y)-detectable difference. Let $\gamma(r_i)$ denote the row obtained by applying the permutaion γ to the entries of r_i. By Claim E.2 and E.2, if the desired condition does not hold then for some parameters r, t, w, when we run the pivoting algorithm on $R(M'_{\eta,\gamma}, B_i)$ we construct a test-set R such that for $i \in \{i_1, \ldots, i_c\}, j \in \{j_1, \ldots, j_c\}$

$$\sigma(\gamma(r_i), R) - \sigma(\gamma(r_j), R) \geq \frac{x}{yl^6\sqrt{|R|}} \geq 1000l$$

This implies that with all but negligible probability

$$\sigma(\gamma(r'_i), R) - \sigma(\gamma(r'_j), R) \geq 900l$$

where $\gamma(r'_i)$ is the noisy version of $\gamma(r_i)$. In particular this implies that with all but negligible probability, either all of the rows $\gamma(r_{i_1}), \ldots, \gamma(r_{i_c})$ are added to the upper set or all of $\gamma(r_{j_1}), \ldots, \gamma(r_{j_c})$ are added to the lower set and the (c, x, y)-detectable difference is resolved. However, this contradicts the assumption in Claim E. ■

We will now essentially aggregate the above result over all $B_i, i \in Y$.

Definition 32 *For $i \in Y$, call an index $g \in G$ B_i-important if the following conditions hold.*

- *There are W marked submatrices in $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$*
- *For at least $0.99W$ of the marked submatrices within $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$, the permutation γ moves at least $0.99z$ highlighted columns by a distance of at least r.*

Note Claim E.3 can be rephrased as follows. For a block B_i, let $b_i = \frac{Z_i}{W}$ be the number of indices $g \in G$ such that $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ contains W marked submatrices. Then for each $i \in Y$, there are at least $0.98b_i$ indices $g' \in G$ that are B_i-important. Claim Consider applying the permutation γ to the columns $c_1, \ldots, c_n$ of $M_{\eta,\mathsf{id}}$. Consider the dyadic decomposition of $\gamma_0 = \mathsf{id}, \gamma_1, \ldots, \gamma_l = \gamma$ and let $\lambda = \lfloor \log_2 \frac{n}{r} \rfloor$. Either there are at least $0.01W|G|z$ indices $j \in [n]$ such that $c_{\gamma_\lambda(j)}$ is $(\frac{ch}{100}, \frac{100y}{W})$-above c_j or there are $0.01W|G|z$ indices $j \in [n]$ such that c_j is $(\frac{ch}{100}, \frac{100y}{W})$-above $c_{\gamma_\lambda(j)}$.

Proof For an index $g \in G$ let $\omega(g)$ be the number of indices $i \in Y$ for which g is B_i-important. Call an index g heavy if $\omega(g) \geq \frac{h}{2}$. Let $G_h \subset G$ denote the set of heavy indices.

Note that the total number of marked submatrices among $R(M_{\eta,\mathsf{id}}, \bigcup_{i \in [\frac{a}{4l^3}] \setminus Y} B_i, \bigcup_{g \in G} \mathsf{In}_{r,g})$ is at most $\frac{h|G|W}{4}$. On the other hand the total number of marked submatrices among $R(M_{\eta,\mathsf{id}}, \bigcup_{i \in [\frac{a}{4l^3}]} B_i, \bigcup_{g \in G} \mathsf{In}_{r,g})$

is exactly $h|G|W$. Therefore, there are at least $\frac{3h|G|W}{4}$ marked submatrices among $R(M_{\eta,\mathsf{id}}, \bigcup_{i\in Y} B_i, \bigcup_{g\in G} \mathsf{In}_{r,g})$. This implies $\sum_{i\in Y} b_i \geq \frac{3h|G|}{4}$. By Claim E.3,

$$\sum_{g\in G} \omega(g) \geq \sum_{i\in Y} 0.98 b_i \geq \frac{2.9h|G|}{4}$$

Note since $R(M_{\eta,\mathsf{id}}, \bigcup_{i\in[\frac{a}{4l^3}]} B_i, \mathsf{In}_{r,g})$ contains exactly hW marked submatrices for all g

$$\sum_{g\in G} \omega(g) = \sum_{g\in G_h} \omega(g) + \sum_{g\in G\setminus G_h} \omega(g) \leq \frac{h}{2}(|G| + |G_h|)$$

In particular, we deduce $|G_h| \geq \frac{|G|}{10}$.

Now fix an element $g \in G_h$ and focus on the submatrix $R(M_{\eta,\mathsf{id}}, \bigcup_{i\in Y} B_i, \mathsf{In}_{r,g})$. For each $i \in Y$, $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ can be viewed as $\frac{r}{2^I}$ disjoint $k_i \times 2^I$ submatrices arranged horizontally.

There are at least $\frac{h}{2}$ indices, say $i_1, \ldots, i_{\frac{h}{2}} \in Y$ such that g is B_i-important. This means that for $i \in \{i_1, \ldots, i_{\frac{h}{2}}\}$, the matrix $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ contains W marked submatrices and for at least $0.99W$ of these submatrices, the permutation γ moves at least $0.99z$ highlighted columns by a distance of at least r. For a given $i \in \{i_1, \ldots, i_{\frac{h}{2}}\}$, let $\mathsf{Mov}_i \subset \mathsf{In}_{r,g}$ be the set of indices of the highlighted columns in $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ that are moved by a distance of at least r. Note $|\mathsf{Mov}_i| \geq 0.98Wz$. For a set S of integers and an integer $1 \leq j \leq |S|$, let $S^{\mathsf{rank}(j)}$ be the j^{th} smallest element of S.

First we claim there must exist some $i_{\mathsf{mid}} \in \{i_1, \ldots, i_{\frac{h}{2}}\}$ such that the following properties hold

- Let $\mathsf{low}_i = \mathsf{Mov}_i^{\mathsf{rank}(\lceil 0.1Wz\rceil)}$ and $\mathsf{high}_i = \mathsf{Mov}_i^{\mathsf{rank}(\lceil |\mathsf{Mov}_i| - 0.1Wz\rceil)}$
- For at least $\frac{h}{10}$ distinct values of $i \in \{i_1, \ldots, i_{\frac{h}{2}}\}$, we have

$$\mathsf{low}_i \leq \mathsf{low}_{i_{\mathsf{mid}}}$$
$$\mathsf{high}_i \geq \mathsf{high}_{i_{\mathsf{mid}}}$$

This is because there are at most $\frac{h}{10}$ values i such that low_i is among the $\frac{h}{10}$-smallest elements of $\{\mathsf{low}_{i_1}, \ldots, \mathsf{low}_{i_{\frac{h}{2}}}\}$ and at most $\frac{h}{10}$ values i such that high_i is among the $\frac{h}{10}$-largest elements of $\{\mathsf{high}_{i_1}, \ldots, \mathsf{high}_{i_{\frac{h}{2}}}\}$ (ties broken in an arbitrary but consistent manner).

Now consider the highlighted columns indexed between $\mathsf{low}_{i_{\mathsf{mid}}}$ and $\mathsf{high}_{i_{\mathsf{mid}}}$ (inclusive) in $R(M_{\eta,\mathsf{id}}, B_{i_{\mathsf{mid}}}, \mathsf{In}_{r,g})$. Let c_j be such a column and since γ moves c_j by at least r distance, WLOG $\gamma^{-1}(j) \leq r(g-1)$. Note this also implies $\gamma_\lambda^{-1}(j) \leq r(g-1)$.

Note that for each index $i \in \{i_1, \ldots, i_{\frac{h}{2}}\}$ such that $\mathsf{low}_i \leq \mathsf{low}_{i_{\mathsf{mid}}}$ and $\mathsf{high}_i \geq \mathsf{high}_{i_{\mathsf{mid}}}$, there are at least $0.1Wz$ highlighted columns in $R(M_{\eta,\mathsf{id}}, B_i, \mathsf{In}_{r,g})$ to the left and right of c_j. This means

there are at least $0.1W$ marked submatrices to the left and right of c_j (both inclusive). This is because each marked submatrix has exactly z highlighted columns.

Using the last property in Claim 23, we claim that c_j, when restricted to the entries in B_i, is at least $(\frac{c}{10}, \frac{100y}{W})$-above $c_{\gamma_\lambda^{-1}(j)}$. To see this restrict $c_j - c_{\gamma_\lambda^{-1}(j)}$ to the rows of B_i that form a (c, x, y) difference and call this restriction $v \in \mathbb{R}^{2c}$. v is lower bounded entry-wise by a sum of the form $v_1 + \cdots + v_e$ where $e \geq 0.05W$ and each v_i is a vector with nonnegative entries and at least $\frac{c}{2}$ entries larger than $\frac{1}{2y}$. This is because the permutation γ moves c_j "through" $0.1W$ distinct marked submatrices. Thus, c_j is $(\frac{ch}{100}, \frac{100y}{W})$-above $c_{\gamma_\lambda^{-1}(j)}$.

Next note that there are at least $0.7Wz$ columns c_j for which we can apply the above argument. Also there are at least $|G_h| \geq \frac{|G|}{10}$ elements $g \in G$ for which we can apply the same argument. Thus either there are $0.01W|G|z$ columns $\gamma_\lambda^{-1}(j)$ such that c_j is $(\frac{ch}{100}, \frac{100y}{W})$-above $\gamma_\lambda^{-1}(j)$ or there are $0.01W|G|z$ columns $c_{\gamma_\lambda^{-1}(j)}$ such that $c_{\gamma_\lambda^{-1}(j)}$ is $(\frac{ch}{100}, \frac{100y}{W})$-above c_j. From this, we immediately get the desired condition. ■

Lemma 33 *Say we have a matrix $M \in \mathsf{Perm}_{n\times n}$ with rows $r_1 < \cdots < r_n$ in order. Let π be a permutation on $[n]$ and say its dyadic decomposition is $\pi_0 = \mathsf{id}, \pi_1, \ldots, \pi_l = \pi$. Let $1 \leq d_0 \leq l$ be some index and assume that there are at least t distinct values of $j \in [n]$ such that $r_{\pi_{d_0}(j)}$ is (x_0, y_0)-above r_j. Then M_π must contain an (a, b, x, y)-error at some level $d < d_0$ with a, b, x, y all powers of 2 such that*

- $x \geq \frac{x_0}{2l^2}$
- $y \leq 2y_0 l$
- $\frac{abx}{y^2} \geq \frac{1}{10^4 l^{20}} \frac{tx_0}{y_0^2}$

Proof Let $S \subset [n]$ be the set of indices $j \in [n]$ such that $r_{\pi_{d_0}(j)}$ is (x_0, y_0)-above r_j. Consider the sequence $r_j, r_{\pi_1(j)}, \ldots, r_{\pi_{d_0}(j)}$. First we claim there must be an index $1 \leq i \leq d_0$ such that $r_{\pi_i(j)}$ is $\left(\frac{x_0}{d_0}, d_0 y_0\right)$-above $r_{\pi_{i-1}(j)}$. This is because for each entry where $r_{\pi_{d_0}(j)} - r_j$ is at least $\frac{1}{y_0}$, there must be some index $1 \leq i \leq d_0$ such that the corresponding entry of $r_{\pi_i(j)} - r_{\pi_{i-1}(j)}$ is at least $\frac{1}{d_0 y_0}$. Therefore for some $1 \leq i \leq d_0$, $r_{\pi_i(j)} - r_{\pi_{i-1}(j)}$ has at least $\frac{x_0}{d_0}$ entries that are at least $\frac{1}{d_0 y_0}$.

We now deduce that for some $1 \leq i \leq d_0$, there is a set $S' \subset [n]$ with $|S'| \geq \frac{t}{d_0}$ such that for all $j \in S'$, $r_{\pi_i(j)}$ is $\left(\frac{x_0}{d_0}, d_0 y_0\right)$-above $r_{\pi_{i-1}(j)}$. For each of the 2^{i-1} dyadic clusters $R(M, D_{l-i+1,j})$ for $j \in \{0, 1, \ldots 2^{i-1} - 1\}$, let $S_j = S' \cap D_{l-i+1,j}$.

Let $k = \frac{n}{2^i}$ and consider $j = 0$. Let the elements of S_0 be $j_1 \leq \cdots \leq j_p$. $R(M_{\pi_{i-1}}, D_{l-i+1,0})$ consists of the $2k$ rows $s_1 < \cdots < s_{2k}$ in order. We will slightly abuse notation and view π_i as a permutation on $\{1, 2, \ldots, 2k\}$. We use $s_{\pi_i(j)}$ to denote the row among $\{s_1, \ldots, s_{2k}\}$ that is moved to location j in $R(M_{\pi_i}, D_{l-i+1,0})$.

Say that in π_i, exactly m rows are swapped between $(s_1, \ldots, s_k)$ and $(s_{k+1}, \ldots, s_{2k})$. Note that $s_{\pi_i(k-m+1)}, \ldots, s_{\pi_i(k)}$ are the rows from among $(s_{k+1}, \ldots, s_{2k})$ that are moved to the first half and

$s_{\pi_i(k+1)}, \ldots, s_{\pi_i(k+m)}$ are the rows from among $(s_1, \ldots, s_k)$ that are moved to the second half.

Consider the following chain for each $1 \leq f \leq m$

$$C_f = \{\pi_i(k-f+1), k-f+1, \pi_i^{-1}(k-f+1), \pi_i^{-2}(k-f+1)\ldots\}$$

where we continue the chain until the first time that $\pi_i^{-c}(k-f+1) > k$. Note that since we assumed that $s_{\pi_i(k-m+1)}, \ldots, s_{\pi_i(k)}$ are the rows from among $(s_{k+1}, \ldots, s_{2k})$ that are moved to the first k positions, the chain eventually terminates. Say the chain continues until $\pi_i^{-c_f}(k-f+1)$ so we have $\pi_i^{-(c_f+1)}(k-f+1) > k$. Note that as f ranges over $1, 2, \ldots m$, the values $\pi_i^{-(c_f+1)}(k-f+1)$ must exactly be the set $k+1, \ldots, k+m$. Thus, there are exactly m distinct chains. Also note that for every index $1 \leq j \leq k$, either $\pi_i(j) = j$ or $\pi_i(j), j$ occur as two consecutive terms in one of the chains.

Consider the set of pairs $P = \{(\pi_i(j_1), j_1), \ldots, (\pi_i(j_p), j_p)\}$. Note we must actually have $1 \leq j_1, \ldots, j_p \leq k$. For each chain C_f, let α_f be the number of pairs from P that occur as consecutive elements in the chain. For each integer g, let β_g be the number of indices $1 \leq f \leq m$ such that $\alpha_f \geq g$. Using a standard argument, there must exist a g such that

$$g\beta_g \geq \frac{\alpha_1 + \cdots + \alpha_m}{2l}$$

If the chain C_f contains at least g pairs of consecutive elements from the set P then Claim C gives us that $s_{\pi_i(k-f+1)}$ is (x', y')-above $s_{\pi_i^{-c_f}(k-f+1)}$ for some x', y' that are both powers of 2 and satisfy

$$x' \geq \frac{x_1}{2l}$$
$$\frac{y_1}{n} \leq y' \leq 2y_1$$
$$\frac{x'}{y'^2} \geq \frac{gx_1}{8ly_1^2}$$

where $x_1 = \frac{x_0}{d_0}, y_1 = y_0 d_0$.

There are only $2l^2$ possibilities for x', y' so over all β_g chains containing at least g pairs from the set P, at least $\lceil \frac{\beta_g}{2l^2} \rceil$ of them must share the same parameters x', y'. This implies that the block $R(M_{\pi_i}, D_{l-i+1,0})$ contains a (c_0, x', y') error for some c_0 such that c_0 is a power of 2 and $c_0 \geq \frac{\beta_g}{4l^2}$. We have

$$\frac{c_0 x'}{y'^2} \geq \frac{g\beta_g x_1}{32l^3 y_1^2} \geq \frac{(\alpha_1 + \cdots + \alpha_m)x_1}{64l^4 y_1^2} = \frac{|S_0|x_1}{64l^4 y_1^2}$$

We can repeat the above argument for all $0 \leq j \leq 2^i - 1$. Using a standard argument, must exist some integer p such that for at least A indices $0 \leq j \leq 2^i - 1$, $|S_j| \geq p$ and

$$Ap \geq \frac{|S_0| + \cdots + |S_{2^i-1}|}{2l} = \frac{|S'|}{2l}$$

For each of these A indices say $j_1, \ldots, j_A$, there exists some (c_0, x', y') error in $R(M_{\pi_i}, D_{l-i+1,j})$ and since there are at most $4l^3$ possibilities for the triple (c_0, x', y') we can choose some (c_0, x', y') that is common for at least $\frac{A}{4l^3}$ of the blocks $R(M_{\pi_i}, D_{l-i+1,j})$. Let a be the largest power of 2 at most $\frac{A}{4l^3}$. Setting $b = c_0, x = x', y = y'$, we deduce that M_π must contain some (a, b, x, y)-error with

$$x \geq \frac{x_1}{2l} \geq \frac{x_0}{2l^2}$$
$$y \leq 2y_1 \leq 2y_0 l$$
$$\frac{abx}{y^2} \geq \frac{A}{8l^3}\frac{c_0 x'}{y'^2} \geq \frac{Apx_1}{2 \cdot 10^3 l^7 y_1^2} \geq \frac{|S'|}{4 \cdot 10^3 l^8} \cdot \frac{x_0}{y_0^2 l^3} \geq \frac{1}{10^4 l^{20}} \cdot \frac{tx_0}{y_0^2}$$

■

Now we are ready to complete the proof of Claim E.

Proof [Proof of Claim E] Combining Lemma 33 and Claim E.3, we get that there must exist some index $1 \leq d_0 \leq l$ such that

- $2^{d_0} \leq \frac{n}{r}$
- γ contains an (a', b', x', y')-error at level d_0 in its dyadic decomposition such that
  - $x' \geq \frac{ch}{200l^2}$
  - $y' \leq \frac{200yl}{W}$
  - $\frac{a'b'x'}{y'^2} \geq \frac{1}{10^4 l^{20}} \cdot \frac{chW^3|G|z}{10^{10}y^2}$

Note that

$$\frac{a'b'x'}{y'^2} \geq \frac{1}{10^4 l^{20}} \cdot \frac{chW^3|G|z}{10^{10}y^2} \geq \frac{1}{10^{20}l^{30}}\frac{acx}{y^2}W^2$$

so if $W \geq l^{3000l^{0.5}}$ then $\frac{a'b'x'}{y'^2} \geq \frac{acx}{y^2}l^{5900l^{0.5}}$. However, this implies that γ contains an (a', b', x', y')-error and we get

$$||M_{\eta,\gamma} - M_{\eta,\mathsf{id}}||_2^2 \geq \frac{acx}{y^2}l^{5000l^{0.5}}$$

Otherwise note $|G|z \leq 2^I \frac{n}{r}$. We then have

- $x' \geq \frac{ch}{200l^2} \geq \frac{acx}{(10l)^8 W|G|z} \geq \frac{acx}{2^I \frac{n}{r} l^{3100l^{0.5}}} \geq \frac{acx}{y^2}\frac{1}{n^{\frac{7}{6}}l^{5100l^{0.5}}} \cdot x \geq l^{4900l^{0.5}}x$
- $\frac{a'b'x'}{y'^2} \geq \frac{1}{10^{20}l^{30}}\frac{acx}{y^2}$

■

E.4. Completing the Analysis of the Full Algorithm

We need a few more technical tools. First we prove a slightly more precise version of Claim C.1.
Claim Let $M \in \mathsf{BISO}_{n\times n}$ be a matrix and let π be a permutation on $[n]$. Assume we have an (a,b,x,y)-difference at level i in the dyadic decomposition of M_π such that $\frac{abx}{y^2} \geq 4nf(n)$ and $y > b$. Then there must exist some $i' \leq \lceil i - \log_2 f(n)\rceil$ such that there exists an (a',b',x',y')-error at level i' in the dyadic decomposition of M_π such that

- $x' \geq \frac{x}{16l^2}$
- $y' \leq 16yl$
- $\frac{a'b'x'}{y'^2} \geq \frac{1}{10^8 l^{20}}\frac{abx}{y^2}$

Proof Note it suffices to consider $i \geq \log_2 f(n)$ since $a \leq 2^i$. Consider the dyadic blocks at level $i_0 = \lceil i - \log_2 f(n)\rceil$, say $D_{l-i_0,0},\ldots,D_{l-i_0,2^{i_0}-1}$. Let X_j be the largest L^1 distance between two rows of $R(M, D_{l-i_0,j})$. Note $X_0 + \cdots + X_{2^{i_0}-1} \leq n$. Let $T \subset \{0,1,\ldots,2^{i_0}-1\}$ be the set of indices j such that $|X_j| \geq \frac{x}{2y}$. We have $|T| \leq \frac{2ny}{x}$.

Also, the (a,b,x,y)-difference we start with can naturally be viewed as the union of a disjoint (b,x,y)-differences. Call these *selected* differences. At most

$$\frac{2ny}{x}\frac{2^i}{2^{i_0}} \leq \frac{2nyf(n)}{x}$$

of the selected (b,x,y)-differences can be contained in dyadic blocks $R(M_\pi, D_{l-i_0,j})$ for $j \in T$. Thus, for $j \in \{0,1,\ldots,2^{i_0}-1\}\backslash T$, there are at least

$$a - \frac{2nyf(n)}{x} \geq \frac{a}{2}$$

selected differences remaining.
Consider such a selected difference consisting of rows $\left(r_{\pi(j_1)},\ldots,r_{\pi(j_b)}\right)$ and $\left(r_{\pi(j'_1)},\ldots r_{\pi(j'_b)}\right)$ contained in $R(M_\pi, D_{l-i_0,j})$ where $j \notin T$.

By the definition of dyadic decomposition, for all $1 \leq f \leq b$ we have $\pi_{i_0}^{-1}(\pi(j_f)), \pi_{i_0}^{-1}(\pi(j'_f)) \in D_{l-i_0,j}$. Also for any $1 \leq f,g \leq b$, the L^1 distance between $r_{\pi_{i_0}^{-1}(\pi(j_f))}$ and $r_{\pi_{i_0}^{-1}(\pi(j'_g))}$ is at most $\frac{x}{2y}$. We claim that this implies, either for all $1 \leq f \leq b$, $r_{\pi(j_f)}$ is $\left(\frac{x}{8}, 8y\right)$-above $r_{\pi_{i_0}^{-1}(\pi(j_f))}$ or for all $1 \leq f \leq b$, $r_{\pi_{i_0}^{-1}(\pi(j'_f))}$ is $\left(\frac{x}{8}, 8y\right)$-above $r_{\pi(j'_f)}$.

Now we can apply Lemma 33 with $t = \frac{ab}{4}, x_0 = \frac{x}{8}, y_0 = 8y$ to deduce that for some $i' \leq i_0$ M_π contains an (a',b',x',y')-error at level i' such that

- $x' \geq \frac{x}{16l^2}$
- $y' \leq 16yl$
- $\frac{a'b'x'}{y'^2} \geq \frac{1}{10^8 l^{20}}\frac{abx}{y^2}$

■

We are now ready to complete the proof of the main result, Theorem 27.

Proof [Proof of Theorem 27] Throughout this proof we will let π_t, σ_t be our algorithm's estimated row and column permutations after t rounds of alternately sorting (running the full row sorting algorithm) the rows and columns. Let $T = l^{10}$ be the number of total rounds. Note $\pi_T = \pi, \sigma_T = \sigma$.

First note that using the same argument as Claim D, we can ensure that with all but negligible probability, we never create any (a, b, x, y)-errors in the row or column permutation such that $\frac{x}{y} \geq 100\sqrt{n}l$. Now assume that the desired claim is not true for $M_{\pi,\mathsf{id}}$ (the exact same argument will work for $M_{\mathsf{id},\sigma}$). Let $E = n^{\frac{7}{6}+10^5\frac{\log\log n}{\sqrt{\log n}}}$. By Lemma 18, there must exist some integer $1 \leq i \leq l$ and some (a, b, x, y)-error at level i such that

- $b \geq y$
- $\frac{abx}{y^2} \geq \frac{E}{l^{0.5}} \geq n^{\frac{7}{6}} l^{9000l^{0.5}}$

Also note that we can assume $\frac{x}{y} \leq 100\sqrt{n}l$. Note the (a, b, x, y)-error can be viewed as a union of a disjoint (b, x, y)-errors. Consider the blocks $R(M_{\pi,\mathsf{id}}, D_{l-i,j})$ with $0 \leq j \leq 2^i - 1$ that contain (b, x, y)-errors. First we consider the case where for at least $\frac{a}{2}$ of the blocks, there exists a (b', x', y')-difference such that

$$y' > b' \tag{11}$$

$$\frac{b'x'}{y'^2} \geq l^{l^{0.5}} \frac{bx}{y^2} \tag{12}$$

Note there are at most l^3 distinct triples (b', x', y') so there must be some triple that occurs at least $\frac{a}{2l^3}$ times and this implies that $M_{\pi,\mathsf{id}}$ contains an $\left(\frac{a}{2l^3}, b', x', y'\right)$-difference at level i in its dyadic decomposition. We can now iteratively apply Claim E.4 (since we cannot decrease the level i indefinitely) to deduce that $M_{\pi,\mathsf{id}}$ must contain, at some level in its dyadic decomposition, an (a'', b'', x'', y'')-error with $b'' \geq y''$ and

$$\frac{a''b''x''}{y''^2} \geq \frac{abx}{y^2} l^{0.5l^{0.5}}$$

Otherwise, we for at least $\frac{a}{2}$ of the blocks that contain (b, x, y)-errors, there are no (b', x', y')-differences for b', x', y' satisfying (11, 12). Fix such a block $R(M_{\pi,\mathsf{id}}, D_{l-i,j})$. Consider running the full row sorting algorithm on $M'_{\pi_{T-1},\gamma_{T-1}}$. Consider the i^{th} round when we run the block sorting algorithm on blocks of size $\frac{n}{2^i}$. The blocks before the i^{th} round of sorting are the same as $R(M'_{\pi,\gamma_{T-1}}, D_{l-i,j})$ for $j = \{0, 1, \ldots, 2^i - 1\}$, except the rows may be permuted within each block. For a given block $R(M'_{\pi,\gamma_{T-1}}, D_{l-i,j})$, let $B^t_{l-i,j}$ be the set of rows remaining after running t rounds of the pivoting algorithm. Note that in the t^{th} iteration of running the pivoting algorithm, we run the pivoting algorithm on $R(M'_{\pi,\gamma_{T-1}}, B^{t-1}_{l-i,j})$ (technically $B^t_{l-i,j}$ is an unordered set, but the order does not matter when we run the pivoting algorithm).

Let $T_{\text{pivot}} = l^{l^{0.51}}$ be the number of times we run the pivoting algorithm. Since there are no (b', x', y')-differences for b', x', y' satisfying (11, 12) and $T_{\text{pivot}} = l^{l^{0.51}}$ is sufficiently large, there must exist some $1 \leq t \leq T_{\text{pivot}}$ such that $R(M'_{\pi,\gamma_{T-1}}, B^{t-1}_{l-i,j})$ contains a (b', x', y')-difference that satisfies

- Either $(b', x', y') = (b, x, y)$ or $\frac{b'x'}{y'^2} \geq l^{l^{0.5}} \frac{bx}{y^2}$
- The (b', x', y')-difference is detectable in $R(M'_{\pi,\gamma_{T-1}}, B^{t-1}_{l-i,j})$ but is not resolved by the t^{th} iteration of the pivoting algorithm

Since there are only l^3 possibilities for (b', x', y'), among the $\frac{a}{2}$ blocks in consideration, there is some triple (b', x', y') such that the above condition holds for at least $\frac{a}{2l^3}$ blocks. We now apply Claim E to deduce that there is some "larger" error in γ_{T-1} from the previous round of sorting the columns. First if $(b', x', y') = (b, x, y)$ then either

$$||M_{\mathsf{id},\gamma_{T-1}} - M_{\mathsf{id},\mathsf{id}}||_2^2 \geq \frac{abx}{2l^3y^2} l^{5000l^{0.5}}$$

or there exists some i' such that in the column permutation γ_{T-1}, there is an (a'', b'', x'', y'')-error at level i' with

$$\frac{a''b''x''}{y''^2} \geq \frac{1}{10^{20}l^{30}} \frac{abx}{2l^3y^2}$$
$$x' \geq l^{4900l^{0.5}} x$$

In the first case we can apply Lemma 18 to conclude that there exists an (a'', b'', x'', y'')-error at some level i' in the dyadic decomposition of γ_{T-1} such that

$$\frac{a''b''x''}{y''^2} \geq l^{4000l^{0.5}} \frac{abx}{y^2}$$

If $(b', x', y') \neq (b, x, y)$ then combining Claim E.4 and Lemma 18, there must exist an (a'', b'', x'', y'')-error at some level i' in the column permutation γ_{T-1} with

$$\frac{a''b''x''}{y''^2} \geq l^{0.9l^{0.5}} \frac{abx}{y^2}$$

Overall, we have shown that if π_T contains an (a, b, x, y)-error with $b \geq y$ and $\frac{abx}{y^2} \geq n^{\frac{7}{6}} l^{9000l^{0.5}}$ at some level i, there must exist one of the following

- An (a'', b'', x'', y'')-error in π_T at some level i' such that $b'' \geq y''$ and $\frac{a''b''x''}{y''^2} \geq \frac{abx}{y^2} l^{0.5l^{0.5}}$
- An (a'', b'', x'', y'')-error in γ_{T-1} at some level i' such that $b'' \geq y''$ and $\frac{a''b''x''}{y''^2} \geq \frac{abx}{y^2} l^{0.9l^{0.5}}$
- An (a'', b'', x'', y'')-error in γ_{T-1} at some level i' such that $b'' \geq y''$ and both of the following hold
  - $\frac{a''b''x''}{y''^2} \geq \frac{1}{l^{200}} \frac{abx}{y^2}$
  - $x'' \geq l^{4800l^{0.5}} x$

We consider replacing a quadruple (a, b, x, y) with a quadruple (a'', b'', x'', y'') using one of the above rules a "step" of type 1, type 2 or type 3. Note there can be at most $\frac{l^{0.5}}{\log l}$ steps of type 3 in a row since for any error with $x \geq 0.1n$ and $\frac{abx}{y^2} \geq n^{\frac{7}{6}}$, we must have $y \leq n^{\frac{5}{12}}$ and thus $\frac{x}{y} \geq 0.1n^{\frac{7}{12}}$. However, as noted at the beginning of the algorithm, such errors occur with negligible probability. Thus if π_T contains an (a, b, x, y)-error at some level i with $b \geq y$ and $\frac{abx}{y^2} \geq n^{\frac{7}{6}} l^{10^4 l^{0.5}}$, there must exist one of the following

- An (a''', b''', x''', y''') error in π_T at some level i' such that $b''' \geq y'''$ and $\frac{a'''b'''x'''}{y'''^2} \geq \frac{abx}{y^2} l^{0.5l^{0.5}}$
- An (a''', b''', x''', y''') error in $\pi_{T'}$ or $\gamma_{T'}$ for $T' \geq T - \left(\frac{l^{0.5}}{\log l} + 1\right)$ rounds earlier such that $b''' \geq y'''$ and

$$\frac{a'''b'''x'''}{y'''^2} \geq \left(\frac{1}{l^{200}}\right)^{\frac{l^{0.5}}{\log l}} l^{0.5l^{0.5}} \frac{abx}{y^2} \geq l^{0.4l^{0.5}} \frac{abx}{y^2}$$

However we run a total of $O(l^{10})$ rounds of alternately sorting the rows and columns so the above immediately gives a contradiction since there cannot be any (a, b, x, y) errors with $\frac{abx}{y^2} \geq n^2$. Thus, our initial assumption about the existence of an a, b, x, y error in the final output was false and we are done.

■