Submitted to the Annals of Statistics
arXiv: arXiv:0000.0000

COMPUTATIONAL AND STATISTICAL BOUNDARIES FOR SUBMATRIX LOCALIZATION IN A LARGE NOISY MATRIX

By T. Tony Cai[*,‡], Tengyuan Liang[‡] and Alexander Rakhlin[†,‡]

University of Pennsylvania[‡]

We study in this paper computational and statistical boundaries for submatrix *localization*. Given one observation of (one or multiple non-overlapping) signal submatrix (of magnitude λ and size $k_m \times k_n$) embedded in a large noise matrix (of size $m \times n$), the goal is to optimal identify the support of the signal submatrix computationally and statistically.

Two transition thresholds for the signal to noise ratio λ/σ are established in terms of m, n, k_m and k_n. The first threshold, $\mathsf{SNR_c}$, corresponds to the computational boundary. We introduce a new linear time spectral algorithm that identifies the submatrix with high probability when the signal strength is above the threshold $\mathsf{SNR_c}$. Below this threshold, it is shown that no polynomial time algorithm can succeed in identifying the submatrix, under the *hidden clique hypothesis*. The second threshold, $\mathsf{SNR_s}$, captures the statistical boundary, below which no method can succeed in localization with probability going to one in the minimax sense. The exhaustive search method successfully finds the submatrix above this threshold. In marked contrast to submatrix detection and sparse PCA, the results show an interesting phenomenon that $\mathsf{SNR_c}$ is *always* significantly larger than $\mathsf{SNR_s}$, which implies an essential gap between statistical optimality and computational efficiency for submatrix localization.

1. Introduction.

The "signal + noise" model

$$X = M + Z, \tag{1.1}$$

where M is the signal of interest and Z is noise, is ubiquitous in statistics and is used in a wide range of applications. Such a "signal + noise" model

[*] The research of Tony Cai was supported in part by NSF Grants DMS-1208982 and DMS-1403708, and NIH Grant R01 CA127334.

[†] Alexander Rakhlin gratefully acknowledges the support of NSF under grant CAREER DMS-0954737.

MSC 2010 subject classifications: Primary 62C20; secondary 90C27

Keywords and phrases: computational boundary, computational complexity, detection, planted clique, lower bounds, minimax, signal to noise ratio, statistical boundary, submatrix localization

has been well studied in statistics in a number of settings, including nonparametric regression where M is a function, and the Gaussian sequence model where M is a finite or an infinite dimensional vector. See, for example, [35, 27] and the references therein. In nonparametric regression, the structural knowledge on M is typically characterized by smoothness, and in the sequence model the structural knowledge on M is often described by sparsity. Fundamental statistical properties such as the minimax estimation rates and the signal detection boundaries have been established under these structural assumptions.

For a range of contemporary applications in statistical learning and signal processing, M and Z in the "signal + noise" model (1.1) are high-dimensional matrices [37, 22, 19, 15, 12]. In this setting, many new interesting problems arise under a variety of structural assumptions on M and the distribution of Z. Examples include sparse principal component analysis (PCA) [41, 6, 7, 10, 11], low-rank matrix de-noising [19], matrix factorization and decomposition [15, 12, 1], non-negative PCA [44, 32], submatrix detection and localization [8, 9], synchronization and planted partition [26, 17], among many others. In the conventional statistical framework, the goal is developing optimal statistical procedures (for estimation, testing, etc), where optimality is understood with respect to the sample size and parameter space.

When the dimensionality of the data becomes large as in many contemporary applications, the computational concerns associated with the statistical procedures come to the forefront. After all, statistical methods are useful in practice only if they can be computed within a reasonable amount of time. A fundamental question is: Is there a price to pay for statistical performance if one only considers computable (polynomial-time) procedures? This question is particularly relevant for non-convex problems with combinatorial structures. These problems pose a significant computational challenge because naive methods based on exhaustive search are typically not computationally efficient. Trade-off between computational efficiency and statistical accuracy in high-dimensional inference has drawn increasing attention in the literature. In particular, [14] and [42] considered a general class of linear inverse problems, with different emphasis on geometry of convex relaxation and decomposition of statistical and computational errors. [13] studied an approach for trading off computational demands with statistical accuracy via relaxation hierarchies. [5, 30, 45] focused on computational difficulties for various statistical problems, such as detection and regression.

In the present paper, we study the interplay between computational efficiency and statistical accuracy in submatrix localization based on a noisy observation of a large matrix. The problem considered in this paper is formalized as follows.

1.1. *Problem Formulation.* Consider the matrix X of the form

$$X = M + Z, \quad \text{where} \quad M = \lambda \cdot 1_{R_m} 1_{C_n}^T \tag{1.2}$$

and $1_{R_m} \in \mathbb{R}^m$ denotes a binary vector with 1 on the index set R_m and zero otherwise. Here, the entries Z_{ij} of the noise matrix are i.i.d. zero-mean sub-Gaussian random variables with parameter σ (defined formally in Equation (1.5)). Given the parameters $m, n, k_m, k_n, \lambda/\sigma$, the set of all distributions described above – for all possible choices of R_m and C_n – forms the submatrix model $\mathcal{M}(m, n, k_m, k_n, \lambda/\sigma)$.

This model can be further extended to multiple submatrices where

$$M = \sum_{s=1}^{r} \lambda_s \cdot 1_{R_s} 1_{C_s}^T \tag{1.3}$$

where $|R_s| = k_s^{(m)}$ and $|C_s| = k_s^{(n)}$ denote the support set of the s-th submatrix. For simplicity, we first focus on the single submatrix and then extend the analysis to the model (1.3) in Section 2.5.

There are two fundamental questions associated with the submatrix model (1.2). One is the *detection* problem: given one observation of the X matrix, decide whether it is generated from a distribution in the submatrix model or from the pure noise model. Precisely, the detection problem considers testing of the hypotheses

$$H_0 : M = \mathbf{0} \quad \text{v.s.} \quad H_\alpha : M \in \mathcal{M}(m, n, k_m, k_n, \lambda/\sigma).$$

The other is the *localization* problem, where the goal is to exactly recover the signal index sets R_m and C_n (the support of the mean matrix M). It is clear that the localization problem is at least as hard (both computationally and statistically) as the detection problem. The focus of the current paper is on the *localization* problem. As we will show in this paper, the localization problem requires larger signal to noise ratio, as well as novel algorithm and analysis to exploit the submatrix structure.

1.2. *Main Results.* To state our main results, let us first define a hierarchy of algorithms in terms of their worst-case running time on instances of the submatrix localization problem:

$$\mathsf{LinAlg} \subset \mathsf{PolyAlg} \subset \mathsf{ExpoAlg} \subset \mathsf{AllAlg}.$$

The set LinAlg contains algorithms $\mathcal{A}$ that produce an answer (in our case, the localization subset $\hat{R}_m^{\mathcal{A}}, \hat{C}_n^{\mathcal{A}}$) in time linear in $m \times n$ (the minimal computation required to read the matrix). The classes $\mathsf{PolyAlg}$ and $\mathsf{ExpoAlg}$ of algorithms, respectively, terminate in polynomial and exponential time, while AllAlg has no restriction.

Combining Theorem 3 and 4 in Section 2 and Theorem 5 in Section 3, the statistical and computational boundaries for submatrix localization can be summarized as follows. The notations $\gtrsim, \lesssim, \asymp$ are formally defined in Section 1.5.

Theorem 1 (Computational and Statistical Boundaries). *Consider the submatrix localization problem under the model (1.2). The computational boundary* $\mathsf{SNR_c}$ *for the dense case when* $\min\{k_m, k_n\} \gtrsim \max\{m^{1/2}, n^{1/2}\}$ *is*

$$\mathsf{SNR_c} \asymp \sqrt{\frac{m \vee n}{k_m k_n}} + \sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}},$$

in the sense that

$$\overline{\lim_{m,n,k_m,k_n \to \infty}} \inf_{\mathcal{A} \in \mathsf{LinAlg}} \sup_{M \in \mathcal{M}} \mathbb{P}\left(\hat{R}_m^{\mathcal{A}} \neq R_m \text{ or } \hat{C}_n^{\mathcal{A}} \neq C_n\right) = 0, \quad \text{if } \frac{\lambda}{\sigma} \gtrsim \mathsf{SNR_c}$$

(1.4)

$$\underline{\lim_{m,n,k_m,k_n \to \infty}} \inf_{\mathcal{A} \in \mathsf{PolyAlg}} \sup_{M \in \mathcal{M}} \mathbb{P}\left(\hat{R}_m^{\mathcal{A}} \neq R_m \text{ or } \hat{C}_n^{\mathcal{A}} \neq C_n\right) > 0, \quad \text{if } \frac{\lambda}{\sigma} \lesssim \mathsf{SNR_c}$$

where (1.4) *holds under the Hidden Clique hypothesis* $\mathsf{HC_l}$ *(see Section 2.1). For the sparse case when* $\max\{k_m, k_n\} \lesssim \min\{m^{1/2}, n^{1/2}\}$*, the computational boundary is* $\mathsf{SNR_c} = \Theta^*(1)$*, more precisely*

$$1 \lesssim \mathsf{SNR_c} \lesssim \sqrt{\log \frac{m \vee n}{k_m k_n}}.$$

The statistical boundary $\mathsf{SNR_s}$ *is*

$$\mathsf{SNR_s} \asymp \sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}},$$

in the sense that

$$\overline{\lim_{m,n,k_m,k_n \to \infty}} \inf_{\mathcal{A} \in \mathsf{ExpoAlg}} \sup_{M \in \mathcal{M}} \mathbb{P}\left(\hat{R}_m^{\mathcal{A}} \neq R_m \text{ or } \hat{C}_n^{\mathcal{A}} \neq C_n\right) = 0, \quad \text{if } \frac{\lambda}{\sigma} \gtrsim \mathsf{SNR_s}$$

$$\underline{\lim_{m,n,k_m,k_n \to \infty}} \inf_{\mathcal{A} \in \mathsf{AllAlg}} \sup_{M \in \mathcal{M}} \mathbb{P}\left(\hat{R}_m^{\mathcal{A}} \neq R_m \text{ or } \hat{C}_n^{\mathcal{A}} \neq C_n\right) > 0, \quad \text{if } \frac{\lambda}{\sigma} \lesssim \mathsf{SNR_s}$$

under the minimal assumption $\max\{k_m, k_n\} \lesssim \min\{m, n\}$*.*

If we parametrize the submatrix model as $m = n, k_m \asymp k_n \asymp k = \Theta^*(n^\alpha), \lambda/\sigma = \Theta^*(n^{-\beta})$, for some $0 < \alpha, \beta < 1$, we can summarize the results of Theorem 1 in a phase diagram, as illustrated in Figure 1.

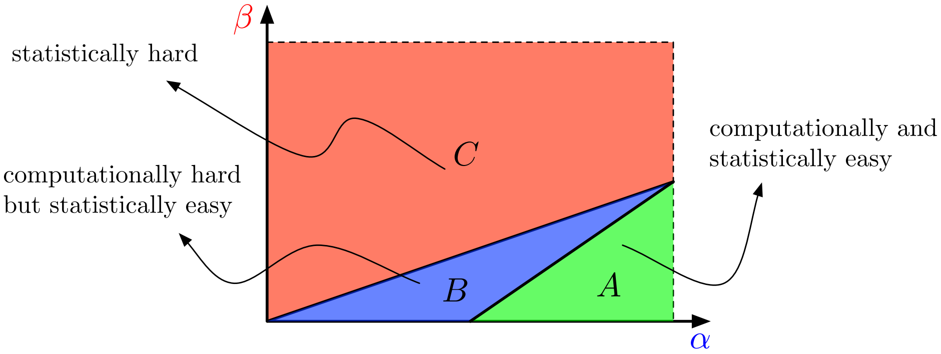

Fig 1. *Phase diagram for submatrix localization. Red region (C): statistically impossible, where even without computational budget, the problem is hard. Blue region (B): statistically possible but computationally expensive (under the hidden clique hypothesis), where the problem is hard to all polynomial time algorithm but easy with exponential time algorithm. Green region (A): statistically possible and computationally easy, where a fast polynomial time algorithm will solve the problem.*

To explain the diagram, consider the following cases. First, the statistical boundary is

$$\sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}},$$

which gives the line separating the red and the blue regions. For the dense regime $\alpha \geq 1/2$, the computational boundary given by Theorem 1 is

$$\sqrt{\frac{m \vee n}{k_m k_n}} + \sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}},$$

which corresponds to the line separating the blue and the green regions. For the sparse regime $\alpha < 1/2$, the computational boundary is $\Theta(1) \precsim \mathsf{SNR_c} \precsim \Theta(\sqrt{\log \frac{m \vee n}{k_m k_n}})$, which is the horizontal line connecting $(\alpha = 0, \beta = 0)$ to $(\alpha = 1/2, \beta = 0)$.

As a key part of Theorem 1, we provide linear time spectral algorithm that will succeed in localizing the submatrix with high probability in the regime above the computational threshold. Furthermore, the method is data-driven and adaptive: it does not require the prior knowledge on the size of the submatrix. This should be contrasted with the method of [16] which requires the prior knowledge of k_m, k_n; furthermore, the running time of their SDP-based method is superlinear in nm. Under the hidden clique hypothesis, we prove that below the computational threshold there is no polynomial time algorithm that can succeed in localizing the submatrix. We remark that the computational lower bound for *localization* requires distinct new techniques compared to the lower bound for *detection*; the latter has been resolved in [30].

Beyond localization of one single submatrix, we generalize both the computational and statistical story to a growing number of submatrices in Section 2.5. As mentioned earlier, the statistical boundary for one single submatrix localization has been investigated by [9] in the Gaussian case. Our result focuses on the computational intrinsic difficulty of localization for a growing number of submatrices, at the expense of not providing the exact constants for the thresholds.

The phase transition diagram in Figure 1 for localization should be contrasted with the corresponding result for detection, as shown in [8, 30]. For a large enough submatrix size (as quantified by $\alpha > 2/3$), the computationally-intractable-but-statistically-possible region collapses for the detection problem, but not for localization. In plain words, detecting the presence of a large submatrix becomes both computationally and statistically easy beyond a certain size, while for localization there is always a gap between statistically possible and computationally feasible regions. This phenomenon also appears to be distinct to that of other problems like estimation of sparse principal components [10], where computational and statistical easiness coincide with each other over a large region of the parameter spaces.

1.3. *Prior Work.* There is a growing body of work in statistical literature on submatrix problems. [2] studied the detection problem for a cluster inside a large matrix. [8, 9] formulated the submatrix detection and localization problems under Gaussian noise and determined sharp statistical transition boundaries. For the detection problem, [30] provided a computational lower bound result under the assumption that hidden clique detection is computationally difficult.

[34] provided a fast iterative maximization algorithm to heuristically solve the submatrix localization problem. [3, 28] focused on statistical and computational trade-offs for the submatrix localization problem. Under the sparse regime $k_m \precsim m^{1/2}$ and $k_n \precsim n^{1/2}$, the entry-wise thresholding turns out to be the "near optimal" polynomial-time algorithm (which we will show to be a de-noised spectral algorithm that perform slightly better in Section 2.4). However, for the dense regime when $k_m \succsim m^{1/2}$ and $k_n \succsim n^{1/2}$, the algo-

rithms provided in [28] are not optimal in the sense that there are other polynomial-time algorithm that can succeed in finding the submatrix with smaller SNR. Concurrently with our work, [16] provided a convex relaxation algorithm that improves the SNR boundary of [28] in the dense regime. On the computational downside, the implementation of the method requires a full SVD on each iteration, and therefore does not scale well with the dimensionality of the problem. Furthermore, there is no computational lower bound in the literature to guarantee the optimality of the SNR boundary achieved in [16]. A problem similar to submatrix localization is that of clique finding in random graph. [18] presented an iterative approximate message passing algorithm to solve the latter problem with sharp boundaries on SNR.

We would like to emphasize on the differences between the localization and the detection problems. In terms of the theoretical results, unlike detection, there is always a gap between statistically optimal and computationally feasible regions for localization. This non-vanishing computational-to-statistical-gap phenomenon also appears in the community detection literature with growing number of communities [17]. In terms of the methodology, for detection, combining the results in [20, 30], there is no loss in treating M in model (1.2) as a vector and applying the higher criticism method [20] to the vectorized matrix for the problem of submatrix detection, in the computationally efficient region. In fact, the procedure achieves sharper constants in the Gaussian setting. However, in contrast, we will show that for localization, it is crucial to utilize the matrix structure, even in the computationally efficient region.

1.4. *Organization of the Paper.* The paper is organized as follows. Section 2 establishes the computational boundary, with the computational lower bounds given in Section 2.1 and upper bound results in Sections 2.2-2.4. An extension to the case of multiple submatrices is presented in Section 2.5. The upper and lower bounds for statistical boundary for multiple submatrices are discussed in Section 3. A short discussion is given in Section 4. Technical proofs are deferred to Section 5. Additional proofs are deferred to Appendix A. In addition to the spectral method, Appendix B contains a new analysis of a known method that is based on a convex relaxation [16]. Comparison of computational lower bounds for localization and detection is included in Appendix C.

1.5. *Notation.* Let $[m]$ denote the index set $\{1, 2, \ldots, m\}$. For a matrix $X \in \mathbb{R}^{m\times n}$, $X_{i\cdot} \in \mathbb{R}^n$ denotes its i-th row and $X_{\cdot j} \in \mathbb{R}^m$ denotes its j-th column. For any $I \subseteq [m]$, $J \subseteq [n]$, X_{IJ} denotes the submatrix corresponding to the index set $I \times J$. For a vector $v \in \mathbb{R}^n$, $\|v\|_{\ell_p} = (\sum_{i\in[n]} |v_i|^p)^{1/p}$ and for a matrix $M \in \mathbb{R}^{m\times n}$, $\|M\|_{\ell_p} = \sup_{v\neq \mathbf{0}} \|Mv\|_{\ell_p}/\|v\|_{\ell_p}$. When $p=2$, the latter is the usual spectral norm, abbreviated as $\|M\|_2$. The nuclear norm of a matrix M is convex surrogate for the rank, with the notation to be $\|M\|_*$. The Frobenius norm of a matrix M is defined as $\|M\|_F = \sqrt{\sum_{i,j} M_{ij}^2}$. The inner product associated with the Frobenius norm is defined as $\langle A, B\rangle = \mathsf{tr}(A^T B)$.

Denote the asymptotic notation $a(n) = \Theta(b(n))$ if there exist two universal constants c_l, c_u such that $c_l \leq \underline{\lim}_{n\to\infty} a(n)/b(n) \leq \overline{\lim}_{n\to\infty} a(n)/b(n) \leq c_u$. Θ^* is asymptotic equivalence hiding logarithmic factors in the following sense: $a(n) = \Theta^*(b(n))$ iff there exists $c>0$ such that $a(n) = \Theta(b(n)\log^c n)$. Additionally, we use the notation $a(n) \asymp b(n)$ as equivalent to $a(n) = \Theta(b(n))$, $a(n) \succsim b(n)$ iff $\lim_{n\to\infty} a(n)/b(n) = \infty$ and $a(n) \precsim b(n)$ iff $\lim_{n\to\infty} a(n)/b(n) = 0$.

We define the zero-mean sub-Gaussian random variable $\mathbf{z}$ with sub-Gaussian parameter σ in terms of its Laplacian

$$\mathbb{E}e^{\lambda \mathbf{z}} \leq \exp(\sigma^2\lambda^2/2), \quad \text{for all} \quad \lambda > 0, \tag{1.5}$$

then we have

$$\mathbb{P}(|\mathbf{z}| > \sigma t) \leq 2\cdot \exp(-t^2/2).$$

We call a random vector $Z \in \mathbb{R}^n$ isotropic with parameter σ if

$$\mathbb{E}(v^T Z)^2 = \sigma^2 \|v\|_{\ell_2}^2, \quad \text{for all} \quad v \in \mathbb{R}^n.$$

Clearly, Gaussian and Bernoulli measures, and more general product measures of zero-mean sub-Gaussian random variables satisfy this isotropic definition up to a constant scalar factor.

2. Computational Boundary.

We characterize in this section the computational boundaries for the submatrix localization problem. Sections 2.1 and 2.2 consider respectively the computational lower bound and upper bound. The computational lower bound given in Theorem 2 is based on the hidden clique hypothesis.

2.1. *Algorithmic Reduction and Computational Lower Bound.*

Theoretical Computer Science identifies a range of problems which are believed to be "hard," in the sense that in the worst-case the required computation grows exponentially with the size of the problem. Faced with a new computational problem, one might try to reduce any of the "hard" problems to the new problem, and therefore claim that the new problem is as hard as the rest in this family. Since statistical procedures typically deal with a random (rather than worst-case) input, it is natural to seek token problems that are believed to be computationally difficult on average with respect to some distribution on instances. The hidden clique problem is one such example (for recent results on this problem, see [23, 18]). While there exists a quasi-polynomial algorithm, no polynomial-time method (for the appropriate regime, described below) is known. Following several other works on reductions for statistical problems, we work under the hypothesis that no polynomial-time method exists.

Let us make the discussion more precise. Consider the hidden clique model $\mathcal{G}(N, \kappa)$ where N is the total number of nodes and κ is the number of clique nodes. In the hidden clique model, a random graph instance is generated in the following way. Choose κ clique nodes uniformly at random from all the possible choices, and connect all the edges within the clique. For all the other edges, connect with probability $1/2$.

Hidden Clique Hypothesis for Localization ($\mathsf{HC_l}$*).* Consider the random instance of hidden clique model $\mathcal{G}(N, \kappa)$. For any sequence $\kappa(N)$ such that $\kappa(N) \le N^{\beta}$ for some $0 < \beta < 1/2$, there is no randomized polynomial time algorithm that can find the planted clique with probability tending to 1 as $N \to \infty$. Mathematically, define the randomized polynomial time algorithm class $\mathsf{PolyAlg}$ as the class of algorithms $\mathcal{A}$ that satisfies

$$\overline{\lim_{N,\kappa(N)\to\infty}} \sup_{\mathcal{A}\in\mathsf{PolyAlg}} \mathbb{E}_{\mathsf{Clique}}\mathbb{P}_{\mathcal{G}(N,\kappa)|\mathsf{Clique}}\left(\text{runtime of } \mathcal{A} \text{ not polynomial in } N\right) = 0.$$

Then

$$\underline{\lim_{N,\kappa(N)\to\infty}} \inf_{\mathcal{A}\in\mathsf{PolyAlg}} \mathbb{E}_{\mathsf{Clique}}\mathbb{P}_{\mathcal{G}(N,\kappa)|\mathsf{Clique}}\left(\text{clique set returned by } \mathcal{A} \text{ not correct}\right) > 0,$$

where $\mathbb{P}_{\mathcal{G}(N,\kappa)|\mathsf{Clique}}$ is the (possibly more detailed due to randomness of algorithm) σ-field conditioned on the clique location and $\mathbb{E}_{\mathsf{Clique}}$ is with respect to uniform distribution over all possible clique locations.

Hidden Clique Hypothesis for Detection ($\mathsf{HC_d}$*).* Consider the hidden clique model $\mathcal{G}(N, \kappa)$. For any sequence of $\kappa(N)$ such that $\kappa(N) \le N^{\beta}$ for some $0 < \beta < 1/2$, there is no randomized polynomial time algorithm that can distinguish between

$$\mathsf{H}_0: \ \mathcal{P}_{\mathsf{ER}} \quad \text{v.s.} \quad \mathsf{H}_\alpha: \ \mathcal{P}_{\mathsf{HC}}$$

with probability going to 1 as $N \to \infty$. Here $\mathcal{P}_{\mathsf{ER}}$ is the Erdős-Rényi model, while $\mathcal{P}_{\mathsf{HC}}$ is the hidden clique model with uniform distribution on all the possible locations of the clique. More precisely,

$$\varliminf_{N,\kappa(N)\to\infty} \inf_{\mathcal{A}\in\mathsf{PolyAlg}} \mathbb{E}_{\mathsf{Clique}}\mathbb{P}_{\mathcal{G}(N,\kappa)|\mathsf{Clique}}\,(\text{detection decision returned by } \mathcal{A} \text{ wrong}) > 0,$$

where $\mathbb{P}_{\mathcal{G}(N,\kappa)|\mathsf{Clique}}$ and $\mathbb{E}_{\mathsf{Clique}}$ are the same as defined in HC_l.

The hidden clique hypothesis has been used recently by several authors to claim computational intractability of certain statistical problems. In particular, [5, 30] assumed the hypothesis HC_d and [43] used HC_l. Localization is harder than detection, in the sense that if an algorithm $\mathcal{A}$ solves the localization problem with high probability, it also correctly solves the detection problem. Assuming that no polynomial time algorithm can solve the detection problem implies impossibility results in localization as well. In plain language, HC_l is a milder hypothesis than HC_d.

We will provide computational lower bound result for localization in Theorem 2. In Appendix C, we contrast the difference of lower bound constructions between localization and detection. The detection computational lower bound was proved in [30]. For the localization computational lower bound, to the best of our knowledge, there is no proof in the literature. Theorem 2 ensures the upper bound in Lemma 1 being sharp.

Theorem 2 (Computational Lower Bound for Localization). *Consider the submatrix model (1.2) with parameter tuple* $(m = n, k_m \asymp k_n \asymp n^\alpha, \lambda/\sigma = n^{-\beta})$, *where* $\frac{1}{2} < \alpha < 1,\ \beta > 0$. *Under the computational assumption* HC_l, *if*

$$\frac{\lambda}{\sigma} \precsim \sqrt{\frac{m+n}{k_m k_n}} \quad \Rightarrow \quad \beta > \alpha - \frac{1}{2},$$

it is not possible to localize the true support of the submatrix with probability going to 1 *within polynomial time.*

Our algorithmic reduction for localization relies on a *bootstrapping* idea based on the matrix structure and a cleaning-up procedure introduced in Lemma 12 given in Section 5. These two key ideas offer new insights in addition to the usual computational lower bound arguments. Bootstrapping introduces an additional randomness on top of the randomness in the hidden clique. Careful examination of these two σ-fields allows us to write the resulting object into mixture of submatrix models. For submatrix localization we need to transform back the submatrix support to the original hidden clique support exactly, with high probability. In plain language, even though we lose track of the exact location of the support when reducing the hidden clique to submatrix model, we can still recover the exact location of the hidden clique with high probability. For technical details of the proof, please refer to Section 5.

2.2. *Adaptive Spectral Algorithm and Upper Bound.* In this section, we introduce linear time algorithm that solves the submatrix localization problem above the computational boundary $\mathsf{SNR_c}$. Our proposed localization Algorithms 1 and 2 are motivated by the spectral algorithm in random graphs [31, 33].

```
Algorithm 1: Vanilla Spectral Projection Algorithm for Dense Regime
Input: $X \in \mathbb{R}^{m\times n}$ the data matrix.
Output: A subset of the row indexes $\hat{R}_m$ and a subset of column
        indexes $\hat{C}_n$ as the localization sets of the submatrix.
1. Compute top left and top right singular vectors $U_{\cdot 1}$ and $V_{\cdot 1}$,
respectively (these correspond to the SVD $X = U\Sigma V^T$);
2. To compute $\hat{C}_n$, calculate the inner products $U_{\cdot 1}^T X_{\cdot j} \in \mathbb{R}, 1 \leq j \leq n$.
These values form two data-driven clusters, and a cut at the largest
gap between consecutive values returns the subsets $\hat{C}_n$ and $[n]\backslash\hat{C}_n$.
Similarly, for the $\hat{R}_m$, calculate $X_{i\cdot} V_{\cdot 1} \in \mathbb{R}, 1 \leq i \leq m$ and obtain two
separated clusters.
```

The proposed algorithm has several advantages over the localization algorithms that appeared in literature. First, it is a linear time algorithm (that is, $\Theta(mn)$ time complexity). The top singular vectors can be evaluated using fast iterative power methods, which is efficient both in terms of space and time. Secondly, this algorithm does not require the prior knowledge of k_m and k_n and automatically adapts to the true submatrix size.

Lemma 1 below justifies the effectiveness of the spectral algorithm.

LEMMA 1 (Guarantee for Spectral Algorithm). *Consider the submatrix model* (1.2)*, Algorithm 1 and assume* $\min\{k_m, k_n\} \gtrsim \max\{m^{1/2}, n^{1/2}\}$. *There exist a universal* $C > 0$ *such that when*

$$\frac{\lambda}{\sigma} \geq C \cdot \left(\sqrt{\frac{m \vee n}{k_m k_n}} + \sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}} \right),$$

the spectral method succeeds in the sense that $\hat{R}_m = R_m, \hat{C}_n = C_n$ *with probability at least* $1 - m^{-c} - n^{-c} - 2\exp\left(-c(m+n)\right)$.

Remark 2.1. The theory and algorithm remain the same if the signal matrix M is more general in the following sense: M has rank one, its left and right singular vectors are sparse, and the nonzero entries of the singular vectors are of the same order. Mathematically, $M = \lambda\sqrt{k_m k_n}\cdot uv^T$, where u, v are unit singular vectors with k_m, k_n non-zero entries, and $|u|_{\max}/|u|_{\min} \leq c$ and $|v|_{\max}/|v|_{\min} \leq c$ for some constant $c \geq 1$. Here for a vector w, $|w|_{\max}$ and $|w|_{\min}$ denote respectively the largest and smallest magnitudes among the nonzero coordinates. When $c = 1$, the algorithm is fully data-driven and does not require the knowledge of $\lambda, \sigma, k_m, k_n$. When c is large but finite, one may require in addition the knowledge of k_m and k_n to perform the final cut to obtain $\hat{C}_n$ and $\hat{R}_m$.

2.3. *Dense Regime.* We are now ready to state the SNR boundary for polynomial-time algorithms (under an appropriate computational assumption), thus excluding the exhaustive search procedure. The results hold under the dense regime when $k \gtrsim n^{1/2}$.

Theorem 3 (Computational Boundary for Dense Regime). *Consider the submatrix model (1.2) and assume* $\min\{k_m, k_n\} \gtrsim \max\{m^{1/2}, n^{1/2}\}$. *There exists a critical rate*

$$\mathsf{SNR_c} \asymp \sqrt{\frac{m \vee n}{k_m k_n}} + \sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}}$$

for the signal to noise ratio $\mathsf{SNR_c}$ *such that for* $\lambda/\sigma \gtrsim \mathsf{SNR_c}$, *both the adaptive linear time Algorithm 1 and the robust polynomial time Algorithm 1 will succeed in submatrix localization, i.e.,* $\hat{R}_m = R_m, \hat{C}_n = C_n$, *with high probability. For* $\lambda/\sigma \lesssim \mathsf{SNR_c}$, *there is no polynomial time algorithm that will work under the hidden clique hypothesis* $\mathsf{HC_l}$.

The proof of the above theorem is based on the theoretical justification of the spectral Algorithm 1 and convex relaxation Algorithm 1, and the new computational lower bound result for localization in Theorem 2. We remark that the analyses can be extended to multiple, even growing number of submatrices case. We postpone a proof of this fact to Section 2.5 for simplicity and focus on the case of a single submatrix.

2.4. *Sparse Regime.* Under the sparse regime when $k \lesssim n^{1/2}$, a naive plug-in of Lemma 1 requires the $\mathsf{SNR_c}$ to be larger than $\Theta(n^{1/2}/k) \gtrsim \sqrt{\log n}$, which implies the vanilla spectral Algorithm 1 is outperformed by simple entrywise thresholding. However, a modified version with entrywise soft-thresholding as a preprocessing de-noising step turns out to provide near optimal performance in the sparse regime. Before we introduce the formal algorithm, let us define the soft-thresholding function at level t to be

$$\eta_t(y) = \operatorname{sign}(y)(|y| - t)_+. \tag{2.1}$$

Soft-thresholding as a de-noising step achieving optimal bias-and-variance trade-off has been widely understood in the wavelet literature, for example, see Donoho and Johnstone [21].

Now we are ready to state the following de-noised spectral Algorithm 2 to localize the submatrix under the sparse regime when $k \precsim n^{1/2}$.

Algorithm 2: De-noised Spectral Algorithm for Sparse Regime

```
Input: $X \in \mathbb{R}^{m \times n}$ the data matrix, a thresholding level
        $t = \Theta(\sigma\sqrt{\log \frac{m \vee n}{k_m k_n}})$.
Output: A subset of the row indexes $\hat{R}_m$ and a subset of column
        indexes $\hat{C}_n$ as the localization sets of the submatrix.
1. Soft-threshold each entry of the matrix $X$ at level $t$, denote the
resulting matrix as $\eta_t(X)$ ;
2. Compute top left and top right singular vectors $U_{\cdot 1}$ and $V_{\cdot 1}$ of matrix
$\eta_t(X)$, respectively (these correspond to the SVD $\eta_t(X) = U \Sigma V^T$);
3. To compute $\hat{C}_n$, calculate the inner products $U_{\cdot 1}^T \cdot \eta_t(X_{\cdot j}), 1 \le j \le n$.
These values form two clusters. Similarly, for the $\hat{R}_m$, calculate
$\eta_t(X_{i \cdot}) \cdot V_{\cdot 1}, 1 \le i \le m$ and obtain two separated clusters. A simple
thresholding procedure returns the subsets $\hat{C}_n$ and $\hat{R}_m$.
```

Lemma 2 below provides the theoretical guarantee for the above algorithm when $k \precsim n^{1/2}$.

Lemma 2 (Guarantee for De-noised Spectral Algorithm). *Consider the submatrix model* (1.2)*, soft-thresholded spectral Algorithm 2 with thresholded level* σt*, and assume* $\min\{k_m, k_n\} \precsim \max\{m^{1/2}, n^{1/2}\}$*. There exist a universal* $C > 0$ *such that when*

$$\frac{\lambda}{\sigma} \ge C \cdot \left(\left[\sqrt{\frac{m \vee n}{k_m k_n}} + \sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}} \right] \cdot e^{-t^2/2} + t \right),$$

the spectral method succeeds in the sense that $\hat{R}_m = R_m, \hat{C}_n = C_n$ *with probability at least* $1 - m^{-c} - n^{-c} - 2\exp(-c(m+n))$*. Further if we choose* $\Theta(\sigma\sqrt{\log \frac{m \vee n}{k_m k_n}})$ *as the optimal thresholding level, we have de-noised spectral* *algorithm works when*

$$\frac{\lambda}{\sigma} \gtrsim \sqrt{\log \frac{m \vee n}{k_m k_n}}.$$

Combining the hidden clique hypothesis $\mathsf{HC_l}$ together with Lemma 2, the following theorem holds under the sparse regime when $k \lesssim n^{1/2}$.

THEOREM 4 (Computational Boundary for Sparse Regime). *Consider the submatrix model (1.2) and assume* $\max\{k_m, k_n\} \lesssim \min\{m^{1/2}, n^{1/2}\}$. *There exists a critical rate for the signal to noise ratio* $\mathsf{SNR_c}$ *between*

$$1 \lesssim \mathsf{SNR_c} \lesssim \sqrt{\log \frac{m \vee n}{k_m k_n}}$$

such that for $\lambda/\sigma \gtrsim \sqrt{\log \frac{m \vee n}{k_m k_n}}$, *the linear time Algorithm 2 will succeed in submatrix localization, i.e.,* $\hat{R}_m = R_m, \hat{C}_n = C_n$, *with high probability. For* $\lambda/\sigma \lesssim 1$, *there is no polynomial time algorithm that will work under the hidden clique hypothesis* $\mathsf{HC_l}$.

REMARK 4.1. The upper bound achieved by the de-noised spectral Algorithm 2 is optimal in the two boundary cases: $k = 1$ and $k \asymp n^{1/2}$. When $k = 1$, both the information theoretic and computational boundary meet at $\sqrt{\log n}$. When $k \asymp n^{1/2}$, the computational lower bound and upper bound match in Theorem 4, thus suggesting the near optimality of Algorithm 2 within the polynomial time algorithm class. The potential logarithmic gap is due to the crudeness of the hidden clique hypothesis. Precisely, for $k = 2$, hidden clique is not only hard for $G(n, p)$ with $p = 1/2$, but also hard for $G(n, p)$ with $p = 1/\log n$. Similarly for $k = n^{\alpha}, \alpha < 1/2$, hidden clique is not only hard for $G(n, p)$ with $p = 1/2$, but also for some $0 < p < 1/2$.

2.5. *Extension to Growing Number of Submatrices.* The computational boundaries established in the previous sections for a single submatrix can be extended to non-overlapping multiple submatrices model (1.3). The non-overlapping assumption corresponds to that for any $1 \leq s \neq t \leq r$, $R_s \cap R_t = \emptyset$ and $C_s \cap C_t = \emptyset$. The Algorithm 3 below is an extension of the spectral projection Algorithm 1 to address the multiple submatrices localization problem.

Algorithm 3: Spectral Algorithm for Multiple Submatrices

```
Input: $X \in \mathbb{R}^{m\times n}$ the data matrix. A pre-specified number of
    submatrices $r$.
Output: A subset of the row indexes $\{\hat{R}_m^s, 1 \leq s \leq r\}$ and a subset of
    column indexes $\{\hat{C}_n^s, 1 \leq s \leq r\}$ as the localization of the
    submatrices.
1. Calculate top $r$ left and right singular vectors in the SVD
   $X = U\Sigma V^T$. Denote these vectors as $U_r \in \mathbb{R}^{m\times r}$ and $V_r \in \mathbb{R}^{n\times r}$,
   respectively;
2. For the $\hat{C}_n^s, 1 \leq s \leq r$, calculate the projection
   $U_r(U_r^T U_r)^{-1} U_r^T X_{\cdot j}, 1 \leq j \leq n$, run $k$-means clustering algorithm (with
   $k = r+1$) for these $n$ vectors in $\mathbb{R}^m$. For the $\hat{R}_m^s, 1 \leq s \leq r$, calculate
   $V_r(V_r^T V_r)^{-1} V_r^T X_{i\cdot}^T, 1 \leq i \leq m$, run $k$-means clustering algorithm (with
   $k = r+1$) for these $m$ vectors in $\mathbb{R}^n$ (while the effective dimension is
   $\mathbb{R}^r$).
```

We emphasize that the following Proposition 3 holds even when the number of submatrices r grows with m, n.

Lemma 3 (Spectral Algorithm for Non-overlapping Submatrices Case). *Consider the non-overlapping multiple submatrices model* (1.3) *and Algorithm 3. Assume*

$$k_s^{(m)} \asymp k_m, k_s^{(n)} \asymp k_n, \lambda_s \asymp \lambda$$

for all $1 \leq s \leq r$ *and* $\min\{k_m, k_n\} \gtrsim \max\{m^{1/2}, n^{1/2}\}$. *There exist a universal* $C > 0$ *such that when*

$$\frac{\lambda}{\sigma} \geq C \cdot \left(\sqrt{\frac{r}{k_m \wedge k_n}} + \sqrt{\frac{\log n}{k_m}} \vee \sqrt{\frac{\log m}{k_n}} + \sqrt{\frac{m \vee n}{k_m k_n}} \right), \tag{2.2}$$

the spectral method succeeds in the sense that $\hat{R}_m^{(s)} = R_m^{(s)}, \hat{C}_n^{(s)} = C_n^{(s)}, 1 \leq s \leq r$ *with probability at least* $1 - m^{-c} - n^{-c} - 2\exp\left(-c(m+n)\right)$.

Remark 4.2. Under the non-overlapping assumption, $rk_m \lesssim m$, $rk_n \lesssim n$ hold in most cases. Thus the first term in Equation (2.2) is dominated by the latter two terms. Thus a growing number r does not affect the bound in Equation (2.2) as long as the non-overlapping assumption holds.

3. Statistical Boundary.

In this section we study the statistical boundary. As mentioned in the introduction, in the Gaussian noise setting, the statistical boundary for a single submatrix localization has been established in

[9]. In this section, we generalize to localization of a growing number of submatrices, as well as sub-Gaussian noise, at the expense of having non-exact constants for the threshold.

3.1. *Information Theoretic Bound.* We begin with the information theoretic lower bound for the localization accuracy.

Lemma 4 (Information Theoretic Lower Bound). *Consider the submatrix model* (1.2) *with Gaussian noise* $Z_{ij} \sim \mathcal{N}(0, \sigma^2)$*. For any fixed* $0 < \alpha < 1$*, there exist a universal constant* C_α *such that if*

$$\frac{\lambda}{\sigma} \leq C_\alpha \cdot \sqrt{\frac{\log(m/k_m)}{k_n} + \frac{\log(n/k_n)}{k_m}}, \tag{3.1}$$

any algorithm $\mathcal{A}$ *will fail to localize the submatrix with probability at least* $1 - \alpha - \frac{\log 2}{k_m \log(m/k_m) + k_n \log(n/k_n)}$ *in the following minimax sense:*

$$\inf_{\mathcal{A} \in \mathsf{AllAlg}} \sup_{M \in \mathcal{M}} \mathbb{P}\left(\hat{R}_m^{\mathcal{A}} \neq R_m \quad or \quad \hat{C}_n^{\mathcal{A}} \neq C_n\right) > 1 - \alpha - \frac{\log 2}{k_m \log(m/k_m) + k_n \log(n/k_n)}.$$

3.2. *Combinatorial Search for Growing Number of Submatrices.* Combinatorial search over all submatrices of size $k_m \times k_n$ finds the location with the strongest aggregate signal and is statistically optimal [9, 8]. Unfortunately, it requires computational complexity $\Theta\left(\binom{m}{k_m} + \binom{n}{k_n}\right)$, which is exponential in k_m, k_n. The search Algorithm 4 was introduced and analyzed under the Gaussian setting for a single submatrix in [8], which can be used iteratively to solve multiple submatrices localization.

```
Algorithm 4: Combinatorial Search Algorithm
  Input: $X \in \mathbb{R}^{m \times n}$ the data matrix.
  Output: A subset of the row indexes $\hat{R}_m$ and a subset of column
          indexes $\hat{C}_n$ as the localization of the submatrix.
  For all index subsets $I \times J$ with $|I| = k_m$ and $|J| = k_n$, calculate the
  sum of the entries in the submatrix $X_{IJ}$. Report the index subset
  $\hat{R}_m \times \hat{C}_n$ with the largest sum.
```

For the case of multiple submatrices, the submatrices can be extracted with the largest sum in a greedy fashion.

Lemma 5 below provides a theoretical guarantee for Algorithm 4 to achieve the information theoretic lower bound.

LEMMA 5 (Guarantee for Search Algorithm). *Consider the non-overlapping multiple submatrices model* (1.3) *and iterative application of Algorithm 4 in a greedy fashion for r times. Assume*

$$k_s^{(m)} \asymp k_m, k_s^{(n)} \asymp k_n, \lambda_s \asymp \lambda$$

for all $1 \leq s \leq r$ and $\max\{k_m, k_n\} \precsim \min\{m, n\}$. There exists a universal constant $C > 0$ such that if

$$\frac{\lambda}{\sigma} \geq C \cdot \sqrt{\frac{\log(em/k_m)}{k_n} + \frac{\log(en/k_n)}{k_m}},$$

then Algorithm 4 will succeed in returning the correct location of the submatrix with probability at least $1 - \frac{2k_m k_n}{mn}$.

To complete Theorem 1, we include the following Theorem 5 capturing the statistical boundary. It is proved by exhibiting the information-theoretic lower bound Lemma 4 and analyzing Algorithm 4.

THEOREM 5 (Statistical Boundary). *Consider the submatrix model* (1.2). *There exists a critical rate*

$$\mathsf{SNR}_s \asymp \sqrt{\frac{\log n}{k_m} \vee \frac{\log m}{k_n}}$$

for the signal to noise ratio, such that for any problem with $\lambda/\sigma \succsim \mathsf{SNR}_\mathsf{s}$, the statistical search Algorithm 4 will succeed in submatrix localization, i.e., $\hat{R}_m = R_m, \hat{C}_n = C_n$, with high probability. On the other hand, if $\lambda/\sigma \precsim \mathsf{SNR}_\mathsf{s}$, no algorithm will work (in the minimax sense) with probability tending to 1.

4. Discussion.

Submatrix Localization v.s. Detection. As pointed out in Section 1.2, for any $k = n^\alpha, 0 < \alpha < 1$, there is an intrinsic SNR gap between computational and statistical boundaries for submatrix localization. Unlike the submatrix detection problem where for the regime $2/3 < \alpha < 1$, there is no gap between what is computationally possible and what is statistical possible, the inevitable gap in submatrix localization is due to the combinatorial structure of the problem. This phenomenon is also seen in some network related problems, for instance, stochastic block models with a growing number of communities [17]. Compared to the submatrix detection problem, the algorithm to solve the localization problem is more complicated and the techniques required for the analysis are much more involved.

Detection for Growing Number of Submatrices. The current paper solves localization of a growing number of submatrices. In comparison, for detection, the only known results are for the case of a single submatrix as considered in [8] for the statistical boundary and in [30] for the computational boundary. The detection problem in the setting of a growing number of submatrices is of significant interest. In particular, it is interesting to understand the computational and statistical trade-offs in such a setting. This will need further investigation.

Estimation of the Noise Level σ. Although Algorithms 1 and 3 do not require the noise level σ as an input, Algorithm 2 does require the knowledge of σ. The noise level σ can be estimated robustly. In the Gaussian case, a simple robust estimator of σ is the following median absolute deviation (MAD) estimator due to the fact that M is sparse $k^2/m^2 \ll 0.25$:

$$\begin{aligned}\hat{\sigma} &= \mathrm{median}_{ij}|X_{ij} - \mathrm{median}_{ij}(X_{ij})|/\Phi^{-1}(0.75)\\ &\approx 1.4826 \times \mathrm{median}_{ij}|X_{ij} - \mathrm{median}_{ij}(X_{ij})|.\end{aligned}$$

5. Proofs.

We prove in this section the main results given in the paper. We first collect and prove a few important technical lemmas that will be used in the proofs of the main results.

5.1. *Prerequisite Lemmas.*

We start with the following version of the Wedin's Theorem.

LEMMA 6 (Davis-Kahan-Wedin's Type Perturbation Bound). *It holds that*

$$\sqrt{\|\sin\Phi\|_F^2 + \|\sin\Theta\|_F^2} \le \frac{\sqrt{2}\|E\|_F}{\delta}$$

and also the following holds for 2*-norm (or any unitary invariant norm)*

$$\max\{\|\sin\Phi\|_2, \|\sin\Theta\|_2\} \le \frac{\|E\|_2}{\delta}.$$

We will then introduce some concentration inequalities. Lemmas 7 and 8 are concentration of measure results from random matrix theory.

LEMMA 7 ([38], Theorem 39). *Let $Z \in \mathbb{R}^{m\times n}$ be a matrix whose rows $Z_{i\cdot}$ are independent sub-Gaussian isotropic random vectors in $\mathbb{R}^n$ with parameter σ. Then for every $t \ge 0$, with probability at least $1 - 2\exp(-ct^2)$ one has*

$$\|Z\|_2 \le \sigma(\sqrt{m} + C\sqrt{n} + t)$$

where $C, c > 0$ are some universal constants.

LEMMA 8 ([25], Projection Lemma). *Assume $Z \in \mathbb{R}^n$ is an isotropic sub-Gaussian vector with i.i.d. entries and parameter σ. $\mathcal{P}$ is a projection operator to a subspace of dimension r, then we have the following concentration inequality*

$$\mathbb{P}(\|\mathcal{P}Z\|_{\ell_2}^2 \geq \sigma^2(r + 2\sqrt{rt} + 2t)) \leq \exp(-ct),$$

where $c > 0$ is a universal constant.

The proof of this lemma is a simple application of Theorem 2.1 in[25] for the case that $\mathcal{P}$ is a rank r positive semidefinite projection matrix.

The following two are standard Chernoff-type bounds for bounded random variables.

LEMMA 9 ([24], Hoeffding's Inequality). *Let $X_i, 1 \leq i \leq n$ be independent random variables. Assume $a_i \leq X_i \leq b_i, 1 \leq i \leq n$. Then for $S_n = \sum_{i=1}^n X_i$*

$$\mathbb{P}\left(|S_n - \mathbb{E}S_n| > u\right) \leq 2\exp\left(-\frac{2u^2}{\sum_{i=1}^n (b_i - a_i)^2}\right). \tag{5.1}$$

LEMMA 10 ([4], Bernstein's Inequality). *Let $X_i, 1 \leq i \leq n$ be independent zero-mean random variables. Suppose $|X_i| \leq M, 1 \leq i \leq n$. Then*

$$\mathbb{P}\left(\sum_{i=1}^n X_i > u\right) \leq \exp\left(-\frac{u^2/2}{\sum_{i=1}^n \mathbb{E}X_i^2 + Mu/3}\right). \tag{5.2}$$

We will end this section stating the Fano's information inequality, which plays a key role in many information theoretic lower bounds.

LEMMA 11 ([35] Corollary 2.6). *Let $\mathcal{P}_0, \mathcal{P}_1, \ldots, \mathcal{P}_M$ be probability measures on the same probability space $(\Theta, \mathcal{F})$, $M \geq 2$. If for some $0 < \alpha < 1$*

$$\frac{1}{M+1}\sum_{i=0}^{M} d_{\mathrm{KL}}(\mathcal{P}_i||\bar{\mathcal{P}}) \leq \alpha \cdot \log M \tag{5.3}$$

where

$$\bar{\mathcal{P}} = \frac{1}{M+1}\sum_{i=0}^{M} \mathcal{P}_i.$$

Then

$$p_{e,M} \geq \bar{p}_{e,M} \geq \frac{\log(M+1) - \log 2}{\log M} - \alpha \tag{5.4}$$

where $p_{e,M}$ is the minimax error for the multiple testing problem.

5.2. *Main Proofs.*

Proof of Lemma 1. Recall the matrix form of the submatrix model, with the SVD decomposition of the mean signal matrix M

$$X = \lambda\sqrt{k_m k_n} UV^T + Z.$$

The largest singular value of λUV^T is $\lambda\sqrt{k_m k_n}$, and all the other singular values are 0s. Davis-Kahan-Wedin's perturbation bound tells us how close the singular space of X is to the singular space of M. Let us apply the derived Lemma 6 to $X = \lambda\sqrt{k_m k_n} UV^T + Z$. Denote the top left and right singular vector of X as $\tilde{U}$ and $\tilde{V}$. One can see that $\mathbb{E}\|Z\|_2 \asymp \sigma(\sqrt{m}+\sqrt{n})$ under very mild finite fourth moment conditions through a result in [29]. Lemma 7 provides a more explicit probabilisitic bound for the concentration of the largest singular value of i.i.d sub-Gaussian random matrix. Because the rows $Z_{i\cdot}$ are sampled from product measure of mean zero sub-Gaussians, they naturally satisfy the isotropic condition. Hence, with probability at least $1-2\exp\left(-c(m+n)\right)$, via Lemma 7, we reach

$$\|Z\|_2 \le C\cdot\sigma(\sqrt{m}+\sqrt{n}). \tag{5.5}$$

Using Weyl's interlacing inequality, we have

$$|\sigma_i(X)-\sigma_i(M)| \le \|Z\|_2$$

and thus

$$\sigma_1(X) \ge \lambda\sqrt{k_m k_n} - \|Z\|_2$$
$$\sigma_2(X) \le \|Z\|_2.$$

Applying Lemma 6, we have

$$\max\left\{|\sin\angle(U,\tilde{U})|, |\sin\angle(V,\tilde{V})|\right\} \le \frac{C\sigma(\sqrt{m}+\sqrt{n})}{\lambda\sqrt{k_m k_n} - C\sigma(\sqrt{m}+\sqrt{n})} \asymp \frac{\sigma(\sqrt{m}+\sqrt{n})}{\lambda\sqrt{k_m k_n}}.$$

In addition

$$\|U-\tilde{U}\|_{\ell_2} = \sqrt{2-2\cos\angle(U,\tilde{U})} = 2|\sin\frac{1}{2}\angle(U,\tilde{U})|,$$

which means

$$\max\left\{\|U-\tilde{U}\|_{\ell_2}, \|V-\tilde{V}\|_{\ell_2}\right\} \le C\cdot\frac{\sigma(\sqrt{m}+\sqrt{n})}{\lambda\sqrt{k_m k_n}}.$$

And according to the definition of the canonical angles, we have

$$\max\left\{\|UU^T - \tilde{U}\tilde{U}^T\|_2, \|VV^T - \tilde{V}\tilde{V}^T\|_2\right\} \leq C \cdot \frac{\sigma(\sqrt{m}+\sqrt{n})}{\lambda\sqrt{k_m k_n}}.$$

Now let us assume we have two observations of X. We use the first observation $\tilde{X}$ to solve for the singular vectors $\tilde{U}, \tilde{V}$, we use the second observation X to project to the singular vectors $\tilde{U}, \tilde{V}$. We can use Tsybakov's sample cloning argument ([36], Lemma 2.1) to create two independent observations of X when noise is Gaussian as follows. Create a pure Gaussian matrix Z' and define $X_1 = X + Z' = M + (Z + Z')$ and $X_2 = X - Z' = M + (Z - Z')$, making X_1, X_2 independent with the variance being doubled. This step is not essential because we can perform random subsampling as in [40]; having two observations instead of one does not change the picture statistically or computationally. Recall $X = M + Z = \lambda\sqrt{k_m k_n} UV^T + Z$.

Define the projection operator to be $\mathcal{P}$, we start the analysis by decomposing

$$\|\mathcal{P}_{\tilde{U}} X_{\cdot j} - M_{\cdot j}\|_{\ell_2} \leq \|\mathcal{P}_{\tilde{U}}(X_{\cdot j} - M_{\cdot j})\|_{\ell_2} + \|(\mathcal{P}_{\tilde{U}} - I)M_{\cdot j}\|_{\ell_2} \tag{5.6}$$

for $1 \leq j \leq n$.

For the first term of (5.6), note that $X_{\cdot j} - M_{\cdot j} = Z_{\cdot j} \in \mathbb{R}^m$ is an i.i.d. isotropic sub-Gaussian vector, and thus we have through Lemma 8, for $t = (1 + 1/c)\log n$, $Z_{\cdot j} \in \mathbb{R}^m, 1 \leq j \leq n$ and $r = 1$

(5.7)

$$\mathbb{P}\left(\|\mathcal{P}_{\tilde{U}}(X_{\cdot j} - M_{\cdot j})\|_{\ell_2} \geq \sigma\sqrt{r}\sqrt{1 + 2\sqrt{1+1/c}\cdot\sqrt{\frac{\log n}{r}} + 2(1+1/c)\cdot\frac{\log n}{r}}\right) \leq n^{-c-1}.$$

We invoke the union bound for all $1 \leq j \leq n$ to obtain

$$\max_{1\leq j\leq n} \|\mathcal{P}_{\tilde{U}}(X_{\cdot j} - M_{\cdot j})\|_{\ell_2} \leq \sigma\sqrt{r} + \sqrt{2(1+1/c)}\cdot\sigma\sqrt{\log n} \tag{5.8}$$

$$\leq \sigma + C\cdot\sigma\sqrt{\log n} \tag{5.9}$$

with probability at least $1 - n^{-c}$.

For the second term $M_{\cdot j} = \tilde{X}_{\cdot j} - \tilde{Z}_{\cdot j}$ of (5.6), there are two ways of upper bounding it. The first approach is to split

$$\|(\mathcal{P}_{\tilde{U}} - I)M\|_2 \leq \|(\mathcal{P}_{\tilde{U}} - I)\tilde{X}\|_2 + \|(\mathcal{P}_{\tilde{U}} - I)\tilde{Z}\|_2 \leq 2\|\tilde{Z}\|_2. \tag{5.10}$$

The first term of (5.10) is $\sigma_2(\tilde{X}) \leq \sigma_2(M) + \|\tilde{Z}\|_2$ through Weyl's interlacing inequality, while the second term is bounded by $\|\tilde{Z}\|_2$. We also know that

$\|\tilde{Z}\|_2 \leq C_3 \cdot \sigma(\sqrt{m} + \sqrt{n})$. Recall the definition of the induced ℓ_2 norm of a matrix $(\mathcal{P}_{\tilde{U}} - I)M$:

$$\|(\mathcal{P}_{\tilde{U}} - I)M\|_2 \geq \frac{\|(\mathcal{P}_{\tilde{U}} - I)MV\|_{\ell_2}}{\|V\|_{\ell_2}} = \|(\mathcal{P}_{\tilde{U}} - I)\lambda\sqrt{k_m k_n}U\|_{\ell_2} \geq \sqrt{k_n}\|(\mathcal{P}_{\tilde{U}} - I)M_{\cdot j}\|_{\ell_2}.$$

In the second approach, the second term of (5.6) can be handled through perturbation Sin Theta Theorem 6:

$$\|(\mathcal{P}_{\tilde{U}} - I)M_{\cdot j}\|_{\ell_2} = \|(\mathcal{P}_{\tilde{U}} - \mathcal{P}_U)M_{\cdot j}\|_{\ell_2} \leq \|\tilde{U}\tilde{U}^T - UU^T\|_2 \cdot \|M_{\cdot j}\|_{\ell_2} \leq C\frac{\sigma\sqrt{m+n}}{\lambda\sqrt{k_m k_n}}\lambda\sqrt{k_m}.$$

This second approach will be used in the multiple submatrices analysis.

Combining all the above, we have with probability at least $1 - n^{-c} - m^{-c}$, for all $1 \leq j \leq n$

$$\|\mathcal{P}_{\tilde{U}} X_{\cdot j} - M_{\cdot j}\|_{\ell_2} \leq C \cdot \left(\sigma\sqrt{\log n} + \sigma\sqrt{\frac{m \vee n}{k_n}}\right). \tag{5.11}$$

Similarly we have for all $1 \leq i \leq m$,

$$\|\mathcal{P}_{\tilde{V}} X_{i\cdot}^T - M_{i\cdot}^T\|_{\ell_2} \leq C \cdot \left(\sigma\sqrt{\log m} + \sigma\sqrt{\frac{m \vee n}{k_m}}\right). \tag{5.12}$$

Clearly we know that for $i \in R_m$ and $i' \in [m]\backslash R_m$

$$\|M_{i\cdot}^T - M_{i'\cdot}^T\|_{\ell_2} = \lambda\sqrt{k_n}$$

and for $j \in C_n$ and $j' \in [n]\backslash C_n$

$$\|M_{\cdot j} - M_{\cdot j'}\|_{\ell_2} = \lambda\sqrt{k_m}$$

Thus if

$$\lambda\sqrt{k_m} \geq 6C \cdot \left(\sigma\sqrt{\log n} + \sigma\sqrt{\frac{m \vee n}{k_n}}\right) \tag{5.13}$$

$$\lambda\sqrt{k_n} \geq 6C \cdot \left(\sigma\sqrt{\log m} + \sigma\sqrt{\frac{m \vee n}{k_m}}\right) \tag{5.14}$$

hold, we have

$$2\max_{i,i' \in R_m} \|\mathcal{P}_{\tilde{V}} X_{i\cdot}^T - \mathcal{P}_{\tilde{V}} X_{i'\cdot}^T\| \leq \min_{i \in R_m, i' \in [m]\backslash R_m} \|\mathcal{P}_{\tilde{V}} X_{i\cdot}^T - \mathcal{P}_{\tilde{V}} X_{i'\cdot}^T\|$$

Therefore we have got $d_i = X_{i\cdot}\tilde{V} \in \mathbb{R}$ (a one dimensional line along direction $\tilde{V}$) such that on this line, data forms two data-driven clusters in the sense that

$$2 \max_{i,i' \in R_m} |d_i - d_{i'}| \leq \min_{i \in R_m, i' \in [m] \backslash R_m} |d_i - d_{i'}|.$$

In this case, the largest adjacent gap in $d_i, i \in [m]$ (data-driven) suggests the cut-off (without requiring the knowledge of λ, σ, k_m). And the simple cut-off clustering recovers the nodes exactly.

In summary, if

$$\lambda \geq C \cdot \sigma \left(\sqrt{\frac{\log n}{k_m}} + \sqrt{\frac{\log m}{k_n}} + \sqrt{\frac{m+n}{k_m k_n}} \right),$$

the spectral algorithm succeeds with probability at least

$$1 - m^{-c} - n^{-c} - 2\exp\left(-c(m+n)\right).$$

□

Proof of Theorem 2. Computational lower bound for localization (support recovery) is of different nature than the computational lower bound for detection (two point testing). The idea is to design a randomized polynomial time algorithmic reduction to relate an instance of *hidden clique* problem to our submatrix localization problem. The proof proceeds in the following way: we will construct a randomized polynomial time transformation $\mathcal{T}$ to map a random instance of $\mathcal{G}(N,\kappa)$ to a random instance of our submatrix $\mathcal{M}(m = n, k_m \asymp k_n \asymp k, \lambda/\sigma)$ (abbreviated as $\mathcal{M}(n, k, \lambda/\sigma)$). Then we will provide a quantitative computational lower bound by showing that if there is a polynomial time algorithm that pushes below the hypothesized computational boundary for localization in the submatrix model, there will be a polynomial time algorithm that solves hidden clique localization with high probability (a contradiction to $\mathsf{HC_l}$).

Denote the randomized polynomial time transformation as

$$\mathcal{T} : \mathcal{G}(N, \kappa(N)) \to M(n, k = n^{\alpha}, \lambda/\sigma = n^{-\beta}).$$

There are several stages for the construction of the algorithmic reduction. First we define a graph $\mathcal{G}^e(N, \kappa(N))$ that is stochastically equivalent to the hidden clique graph $\mathcal{G}(N, \kappa(N))$, but is easier for theoretical analysis. $\mathcal{G}^e$ has the property: each node independently has the probability $\kappa(N)/N$ to be a clique node, and with the remaining probability a non-clique node.

Using Bernstein's inequality and the inequality (5.20) proved below, with probability at least $1-2N^{-1}$ the number of clique nodes κ^e in $\mathcal{G}^e$

$$\kappa\left(1-\sqrt{\frac{4\log N}{\kappa}}\right)\leq\kappa^e\leq\kappa\left(1+\sqrt{\frac{4\log N}{\kappa}}\right)\Rightarrow\kappa^e\asymp\kappa \tag{5.15}$$

as long as $\kappa\gtrsim\log N$.

Consider a hidden clique graph $\mathcal{G}^e(2N,2\kappa(N))$ with $N=n$ and $\kappa(N)=\kappa$. Denote the set of clique nodes for $\mathcal{G}^e(2N,2\kappa(N))$ to be $C_{N,\kappa}$. Represent the hidden clique graph using the symmetric adjacency matrix $G\in\{-1,1\}^{2N\times 2N}$, where $G_{ij}=1$ if $i,j\in C_{N,\kappa}$, otherwise with equal probability to be either -1 or 1. As remarked before, with probability at least $1-2N^{-1}$, we have planted $2\kappa(1\pm o(1))$ clique nodes in graph $\mathcal{G}^e$ with $2N$ nodes. Take out the upper-right submatrix of G, denote as G_{UR} where U is the index set $1\leq i\leq N$ and R is the index set $N+1\leq j\leq 2N$. Now G_{UR} has independent entries.

The construction of $\mathcal{T}$ employs the *Bootstrapping* idea. Generate l^2 (with $l\asymp n^\beta, 0<\beta<1/2$) matrices through bootstrap subsampling as follows. Generate $l-1$ independent index vectors $\psi^{(s)}\in\mathbb{R}^n, 1\leq s<l$, where each element $\psi^{(s)}(i), 1\leq i\leq n$ is a random draw with replacement from the row indices $[n]$. Denote vector $\psi^{(0)}(i)=i, 1\leq i\leq n$ as the original index set. Similarly, we can define independently the column index vectors $\phi^{(t)}, 1\leq t<l$. We remark that these bootstrap samples can be generated in polynomial time $\Omega(l^2n^2)$. The transformation is a weighted average of l^2 matrices of size $n\times n$ generated based on the original adjacency matrix G_{UR}.

$$\mathcal{T}:\quad M_{ij}=\frac{1}{l}\sum_{0\leq s,t<l}(G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j)},\quad 1\leq i,j\leq n. \tag{5.16}$$

Recall that $C_{N,\kappa}$ stands for the clique set of the hidden clique graph. We define the row candidate set $R_l:=\{i\in[n]:\exists\ 0\leq s<l,\psi^{(s)}(i)\in C_{N,\kappa}\}$ and column candidate set $C_l:=\{j\in[n]:\exists\ 0\leq t<l,\phi^{(t)}(j)\in C_{N,\kappa}\}$. Observe that $R_l\times C_l$ are the indices where the matrix M contains signal.

There are two cases for M_{ij}, given the candidate set $R_l\times C_l$. If $i\in R_l$ and $j\in C_l$, namely when (i,j) is a clique edge in at least one of the l^2 matrices, then $\mathbb{E}[M_{ij}|\mathcal{G}^e]\geq l^{-1}$ where the expectation is taken over the bootstrap σ-field conditioned on the candidate set $R_l\times C_l$ and the original σ-field of $\mathcal{G}^e$. Otherwise $\mathbb{E}[M_{ij}|\mathcal{G}^e]=l(\frac{|E|}{N^2-\kappa^2}-\frac{1}{2})$ for $(i,j)\notin R_l\times C_l$, where $|E|$ is a Binomial$(N^2-\kappa^2,1/2)$. With high probability, $\mathbb{E}[M_{ij}|\mathcal{G}^e]\asymp\frac{l}{\sqrt{N^2-\kappa^2}}\asymp\frac{l}{n}=$

$o(\frac{1}{l})$. Thus the mean separation between the signal position and non-signal position is $\frac{1}{\ell} - \frac{l}{n} \asymp \frac{1}{\ell}$. Note in the submatrix model, it does not matter if the noise has mean zero or not (since we can subtract the mean)– only the signal separation matters.

Now let us discuss the independence issue in M through our Bootstrapping construction. Clearly due to sampling with replacement and bootstrapping, condition on $\mathcal{G}^e$, we have independence among samples for the same location (i, j)

$$(G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j)} \perp (G_{UR})_{\psi^{(s')}(i)\phi^{(t')}(j)}.$$

For the independence among entries in one Bootstrapped matrix, clearly

$$(G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j)} \perp (G_{UR})_{\psi^{(s)}(i')\phi^{(t)}(j')}.$$

The only case where there might be a weak dependence is between

$$(G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j)}, (G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j')}$$

and $(G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j)}, (G_{UR})_{\psi^{(s)}(i)\phi^{(t')}(j)}$. The way to eliminate the weak dependence is through [39]'s result on universality of random discrete graphs. [39] showed that random regular graph $\mathcal{G}(n, n/2)$ shares many similarities with Erdős-Rényi random graph $\mathcal{G}(n, 1/2)$: for instance, top and second eigenvalues ($n/2$ and $\sqrt{n}$ respectively), limiting spectral distribution, sandwich conjecture, determinant, etc. Let us consider the case where the upper-right of the adjacency matrix G consists of random bi-regular graph with a planted clique. We assume that the hidden clique hypothesis for $k \precsim \sqrt{n}$ is still valid for the following random graph: for a $n \times n$ adjacency matrix G, first find a clique/principal submatrix of size k uniformly randomly and connect density, for the remaining part of the matrix, sample a random regular graph of $G(n-k, \frac{n-k}{2})$ and a random bi-regular graph of size $k \times (n-k)$ with left regular degree $n/2 - k$ and right regular degree $k/2$ (here degree test will not work in this graph and spectral barrier still suggests $k \precsim \sqrt{n}$ is hard due to universality result of random discrete graphs). In the bootstrapping step, conditionally on the row $\psi^{(s)}(i)$ being not a clique, $(G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j)} \perp (G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j')} | \psi^{(s)}(i)$, and each one is a Rademacher random variable (regardless of the choice of $\psi^{(s)}(i)$), which implies $(G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j)} \perp (G_{UR})_{\psi^{(s)}(i)\phi^{(t)}(j')}$ holds unconditionally. Thus in the bootstrapping procedure, we have independence among entries within the matrix unconditionally.

Let us move to verify the sub-Gaussianity of M matrix. Note that for the index i, j that is not a clique for any of the matrices, M_{ij} is sub-Gaussian, due to Hoeffding's inequality

$$\mathbb{P}\left(|M_{ij} - \mathbb{E}M_{ij}| \geq u\right) \leq 2\exp(-u^2/2). \tag{5.17}$$

For the index i, j being a clique in at least one of the matrices, we claim the number of matrices has (i, j) being clique is $O^*(1)$. Due to Bernstein's inequality, we have $\max_i |\{0 \leq s < l : \psi^{(s)}(i) \in C_{N,\kappa}\}| \leq \frac{\kappa l}{n} + \frac{8}{3}\log n$ with probability at least $1 - n^{-1}$. This further implies there are at least $l^2 - (\frac{\kappa l}{n} + \frac{8}{3}\log n)^2$ many independent Rademacher random variables in each i, j position, thus

$$\mathbb{P}\left(|M_{ij} - \mathbb{E}M_{ij}| \geq u\right) \leq 2\exp\left(-(1 - C \cdot (\kappa n^{-1} + l^{-1}\log n)^2)u^2/2\right). \tag{5.18}$$

Up to now we have proved that when i, j is a signal node for M, then $O^*(1)l^{-1} \geq \mathbb{E}M_{ij} \geq l^{-1}$. Thus the sub-Gaussian parameter is $\sigma = 1 - o(1)$ because $\kappa n^{-1}, l^{-1}\log n$ are both $o(1)$. The constructed $M(n, k, \lambda/\sigma)$ matrix satisfies the submatrix model with $\lambda/\sigma \asymp l^{-1}$ and sub-Gaussian parameter $\sigma = 1 - o(1)$.

Let us estimate the corresponding k in the submatrix model. We need to bound the order of the cardinality of R_l, denoted as $|R_l|$. The total number of positions with signal (at least one clique node inside) is

$$\mathbb{E}|R_l| = \mathbb{E}|\{1 \leq i \leq n : i \in R_l\}| = n\left[1 - (1 - \kappa/n)^l\right].$$

Thus we have the two sided bound

$$\kappa l\left(1 - \frac{\kappa l}{2n}\right) \leq \mathbb{E}|R_l| \leq \kappa l$$

which is of the order $k := \kappa l$. Let us provide a high probability bound on $|R_l|$. By Bernstein's inequality

$$\mathbb{P}\left(||R_l| - \mathbb{E}|R_l|| > u\right) \leq 2\exp\left(-\frac{u^2/2}{\kappa l + u/3}\right). \tag{5.19}$$

Thus if we take $u = \sqrt{4\kappa l \log n}$, as long as $\log n = o(\kappa l)$,

$$\mathbb{P}\left(||R_l| - \mathbb{E}|R_l|| > \sqrt{4\kappa l \log n}\right) \leq 2n^{-1}. \tag{5.20}$$

So with probability at least $1 - 2n^{-1}$, the number of positions that contain signal nodes is bounded as

$$\kappa l\left(1 - \frac{\kappa l}{n}\right)\left(1 - \sqrt{\frac{4\log n}{\kappa l}}\right) < |R_l| < \kappa l\left(1 + \sqrt{\frac{4\log n}{\kappa l}}\right) \Rightarrow |R_l| \asymp \kappa l. \tag{5.21}$$

Equation (5.21) implies that with high probability

$$\kappa l(1-o(1)) \leq |R_l| \leq \kappa l(1+o(1)),$$
$$\kappa l(1-o(1)) \leq |C_l| \leq \kappa l(1+o(1)).$$

The above means, in the submatrix parametrization, $k_m \asymp k_n \asymp \kappa l \asymp n^\alpha$, $\lambda/\sigma \asymp l^{-1} \asymp n^{-\beta}$, which implies $\kappa \asymp n^{\alpha-\beta}$.

Suppose there exists a polynomial time algorithm $\mathcal{A}_M$ that pushes below the computational boundary. In other words,

$$n^{-\beta} \asymp \frac{\lambda}{\sigma} \precsim \sqrt{\frac{m+n}{k_m k_n}} \asymp n^{(1-2\alpha)/2} \Rightarrow \beta > \alpha - \frac{1}{2}$$

with the last inequality having a slack $\epsilon > 0$. More precisely, $\mathcal{A}_M$ returns two estimated index sets $\hat{R}_n$ and $\hat{C}_n$ corresponding to the location of the submatrix (and correct with probability going to 1) under the regime $\beta = \alpha - 1/2 + \epsilon$. Suppose under some conditions, this algorithm $\mathcal{A}_M$ can be modified to a randomized polynomial time algorithm $\mathcal{A}_\mathcal{G}$ that correctly identifies the hidden clique nodes with high probability. It means in the corresponding hidden clique graph $\mathcal{G}(2N, 2\kappa)$, $\mathcal{A}_\mathcal{G}$ also pushes below the computational boundary of hidden clique by the amount ϵ:

$$\kappa(N) = 2\kappa \asymp (2n)^{\alpha-\beta} \asymp n^{1/2-\epsilon} \precsim n^{1/2} \asymp N^{\frac{1}{2}}.$$

In summary, the quantitative computational lower bound implies that if the computational boundary for submatrix localization is pushed below by an amount ϵ in the power, the *hidden clique* boundary is correspondingly improved by ϵ.

Now let us show that any algorithm $\mathcal{A}_M$ that localizes the submatrix introduces a randomized algorithm that finds the hidden clique nodes with probability tending to 1. The algorithm relies on the following simple lemma.

LEMMA 12. *For the hidden clique model $\mathcal{G}(N, \kappa)$, suppose an algorithm provides a candidate set S of size k that contains the true clique subset. If*

$$\kappa \geq C\sqrt{k \log N}$$

then by looking at the adjacency matrix restricted to S we can recover the clique subset exactly with high probability.

The proof of Lemma 12 is immediate. If i is a clique node, then $\min_i \sum_{j\in C} G_{ij} \geq \kappa - C/2 \cdot \sqrt{k \log N}$. If i is not a clique node, then $\max_i \sum_{j\in C} G_{ij} \leq C/2 \cdot \sqrt{k \log N}$. The proof is completed.

Algorithm $\mathcal{A}_M$ provides candidate sets R_l, C_l of size k, inside which κ are correct clique nodes, and thus exact recovery can be completed through Lemma 12 since $\kappa \gtrsim (k \log N)^{1/2}$ (since $\kappa \asymp n^{1/2-\epsilon} \gtrsim k^{1/2} \asymp n^{\alpha/2}$ when ϵ is small). The algorithm $\mathcal{A}_M$ induces another randomized polynomial time algorithm $\mathcal{A}_{\mathcal{G}}$ that solves the hidden clique problem $\mathcal{G}(2N, 2\kappa)$ with $\kappa \precsim N^{1/2}$. The algorithm $\mathcal{A}_{\mathcal{G}}$ returns the support $\hat{C}_{N,\kappa}$ that coincides with the true support $C_{N,\kappa}$ with probability going to 1 (a contradiction to the hidden clique hypothesis $\mathsf{HC_l}$). We conclude that, under the hypothesis, there is no polynomial time algorithm $\mathcal{A}_M$ that can push below the computational boundary $\lambda \precsim \sqrt{\frac{m+n}{k_m k_n}}$.

□

Proof of Theorem 3 is a direct result of Lemma 1 and Theorem 2. Proof of Theorem 4 is obvious based on Lemma 2 and the hidden clique hypothesis $\mathsf{HC_l}$. Proof of Theorem 5 combines the result of Lemma 5 and Lemma 4.

SUPPLEMENTARY MATERIAL

Supplement to: "COMPUTATIONAL AND STATISTICAL BOUNDARIES FOR SUBMATRIX LOCALIZATION IN A LARGE NOISY MATRIX"
(doi: COMPLETED BY THE TYPESETTER; Paper-Supplement.pdf). Due to space constraints, we have relegated remaining proofs to the Supplement.

References.

[1] Agarwal, A., Negahban, S., Wainwright, M. J., et al. (2012). Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions. *The Annals of Statistics*, 40(2):1171–1197.

[2] Arias-Castro, E., Candès, E. J., Durand, A., et al. (2011). Detection of an anomalous cluster in a network. *The Annals of Statistics*, 39(1):278–304.

[3] Balakrishnan, S., Kolar, M., Rinaldo, A., Singh, A., and Wasserman, L. (2011). Statistical and computational tradeoffs in biclustering. In *NIPS 2011 Workshop on Computational Trade-offs in Statistical Learning*.

[4] Bennett, G. (1962). Probability inequalities for the sum of independent random variables. *Journal of the American Statistical Association*, 57(297):33–45.

[5] Berthet, Q. and Rigollet, P. (2013a). Computational lower bounds for sparse pca. *arXiv preprint arXiv:1304.0828*.

[6] Berthet, Q. and Rigollet, P. (2013b). Optimal detection of sparse principal components in high dimension. *The Annals of Statistics*, 41(4):1780–1815.

[7] Birnbaum, A., Johnstone, I. M., Nadler, B., and Paul, D. (2013). Minimax bounds for sparse pca with noisy high-dimensional data. *Annals of statistics*, 41(3):1055.

[8] Butucea, C. and Ingster, Y. I. (2013). Detection of a sparse submatrix of a high-dimensional noisy matrix. *Bernoulli*, 19(5B):2652–2688.

[9] Butucea, C., Ingster, Y. I., and Suslina, I. (2013). Sharp variable selection of a sparse submatrix in a high-dimensional noisy matrix. *arXiv preprint arXiv:1303.5647*.

[10] Cai, T. T., Ma, Z., and Wu, Y. (2013). Sparse pca: Optimal rates and adaptive estimation. *The Annals of Statistics*, 41(6):3074–3110.

[11] Cai, T. T., Ma, Z., and Wu, Y. (2015). Optimal estimation and rank detection for sparse spiked covariance matrices. *Probability Theory and Related Fields*, page to appear.

[12] Candès, E. J., Li, X., Ma, Y., and Wright, J. (2011). Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11.

[13] Chandrasekaran, V. and Jordan, M. I. (2013). Computational and statistical tradeoffs via convex relaxation. *Proceedings of the National Academy of Sciences*, 110(13):E1181–E1190.

[14] Chandrasekaran, V., Recht, B., Parrilo, P. A., and Willsky, A. S. (2012). The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849.

[15] Chandrasekaran, V., Sanghavi, S., Parrilo, P. A., and Willsky, A. S. (2009). Sparse and low-rank matrix decompositions. In *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, pages 962–967. IEEE.

[16] Chen, Y. and Xu, J. (2014). Statistical-computational tradeoffs in planted problems and submatrix localization with a growing number of clusters and submatrices. *arXiv preprint arXiv:1402.1267*.

[17] Decelle, A., Krzakala, F., Moore, C., and Zdeborová, L. (2011). Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106.

[18] Deshpande, Y. and Montanari, A. (2013). Finding hidden cliques of size\ sqrt {N/e} in nearly linear time. *arXiv preprint arXiv:1304.7047*.

[19] Donoho, D. and Gavish, M. (2014). Minimax risk of matrix denoising by singular value thresholding. *The Annals of Statistics*, 42(6):2413–2440.

[20] Donoho, D. and Jin, J. (2004). Higher criticism for detecting sparse heterogeneous mixtures. *Annals of Statistics*, pages 962–994.

[21] Donoho, D. L. and Johnstone, I. M. (1998). Minimax estimation via wavelet shrinkage. *The Annals of Statistics*, 26(3):879–921.

[22] Drineas, P., Kannan, R., and Mahoney, M. W. (2006). Fast monte carlo algorithms for matrices ii: Computing a low-rank approximation to a matrix. *SIAM Journal on Computing*, 36(1):158–183.

[23] Feldman, V., Grigorescu, E., Reyzin, L., Vempala, S., and Xiao, Y. (2013). Statistical algorithms and a lower bound for detecting planted cliques. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 655–664. ACM.

[24] Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American statistical association*, 58(301):13–30.

[25] Hsu, D., Kakade, S. M., and Zhang, T. (2012). A tail inequality for quadratic forms of subgaussian random vectors. *Electron. Commun. Probab*, 17(6).

[26] Javanmard, A., Montanari, A., and Ricci-Tersenghi, F. (2015). Phase transitions in semidefinite relaxations. *arXiv preprint arXiv:1511.08769*.

[27] Johnstone, I. M. (2013). Gaussian estimation: Sequence and wavelet models. *Manuscript*.

[28] Kolar, M., Balakrishnan, S., Rinaldo, A., and Singh, A. (2011). Minimax localization of structural information in large noisy matrices. In *Advances in Neural Information Processing Systems*, pages 909–917.

[29] Latała, R. (2005). Some estimates of norms of random matrices. *Proceedings of the American Mathematical Society*, 133(5):1273–1282.

[30] Ma, Z. and Wu, Y. (2013). Computational barriers in minimax submatrix detection.

arXiv preprint arXiv:1309.5914.

[31] McSherry, F. (2001). Spectral partitioning of random graphs. In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pages 529–537. IEEE.

[32] Montanari, A. and Richard, E. (2014). Non-negative principal component analysis: Message passing algorithms and sharp asymptotics. *arXiv preprint arXiv:1406.4775*.

[33] Ng, A. Y., Jordan, M. I., Weiss, Y., et al. (2002). On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856.

[34] Shabalin, A. A., Weigman, V. J., Perou, C. M., and Nobel, A. B. (2009). Finding large average submatrices in high dimensional data. *The Annals of Applied Statistics*, pages 985–1012.

[35] Tsybakov, A. B. (2009). *Introduction to nonparametric estimation*, volume 11. Springer Series in Statistics.

[36] Tsybakov, A. B. (2014). Aggregation and minimax optimality in high-dimensional estimation.

[37] Tufts, D. W. and Shah, A. A. (1993). Estimation of a signal waveform from noisy data using low-rank approximation to a data matrix. *Signal Processing, IEEE Transactions on*, 41(4):1716–1721.

[38] Vershynin, R. (2010). Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*.

[39] Vu, V. (2008). Random discrete matrices. In *Horizons of combinatorics*, pages 257–280. Springer.

[40] Vu, V. (2014). A simple svd algorithm for finding hidden partitions. *arXiv preprint arXiv:1404.3918*.

[41] Vu, V. Q. and Lei, J. (2012). Minimax rates of estimation for sparse pca in high dimensions. *arXiv preprint arXiv:1202.0786*.

[42] Wainwright, M. J. (2014). Structured regularizers for high-dimensional problems: Statistical and computational issues. *Annual Review of Statistics and Its Application*, 1:233–253.

[43] Wang, T., Berthet, Q., and Samworth, R. J. (2014). Statistical and computational trade-offs in estimation of sparse principal components. *The Annals of Statistics, to appear*.

[44] Zass, R. and Shashua, A. (2006). Nonnegative sparse pca. In *Advances in Neural Information Processing Systems*, pages 1561–1568.

[45] Zhang, Y., Wainwright, M. J., and Jordan, M. I. (2014). Lower bounds on the performance of polynomial-time algorithms for sparse linear regression. *arXiv preprint arXiv:1402.1918*.

Department of Statistics
The Wharton School
University of Pennsylvania
Philadelphia, PA 19104
USA
E-mail: tcai@wharton.upenn.edu
E-mail: tengyuan@wharton.upenn.edu
E-mail: rakhlin@wharton.upenn.edu