Estimating network edge probabilities by neighborhood smoothing

Yuan Zhang, Elizaveta Levina and Ji Zhu
{yzhanghf, elevina, jizhu}@umich.edu

Abstract

The problem of estimating probabilities of network edges from the observed adjacency matrix has important applications to predicting missing links and network denoising. It has usually been addressed by estimating the graphon, a function that determines the matrix of edge probabilities, but is ill-defined without strong assumptions on the network structure. Here we propose a novel computationally efficient method based on neighborhood smoothing to estimate the expectation of the adjacency matrix directly, without making the strong structural assumptions graphon estimation requires. The neighborhood smoothing method requires little tuning, has a competitive mean-squared error rate, and outperforms many benchmark methods on the task of link prediction in both simulated and real networks.

1 Introduction

Statistical network analysis spans a wide range of disciplines (network science, statistics, physics, computer science, sociology, and others) and an equally wide range of applications and analysis tasks (community detection, link prediction, etc). In this paper, we study the problem of inferring the generative mechanism of an undirected network based on a single realization of the network. The data consist of the network adjacency matrix $A \in \{0,1\}^{n\times n}$, where n is the number of nodes, and $A_{ij} = A_{ji} = 1$ if there is an edge between nodes i and j. We assume the observed adjacency matrix A is generated from an underlying probability matrix P, so that for $i \leq j$, A_{ij}'s are independent Bernoulli(P_{ij}) trials, and P_{ij}'s are edge probabilities.

It is obviously impossible to estimate P from a single realization of A unless one assumes some form of structure in P. When the network is expected to have communities, arguably the most popular assumption is that of the stochastic block model, where each node belongs to one of K blocks and the probability of an edge between two nodes is determined by the blocks the nodes

belong to. In this case, the $n \times n$ matrix P is parametrized by the $K \times K$ matrix of within- and between-block edge probabilities, and thus it is possible to estimate P from a single realization. The main challenge in fitting the stochastic block model is estimating the blocks themselves, and that has been the focus of the literature, see for example (Bickel & Chen, 2009; Rohe et al., 2011; Amini et al., 2013; Guédon & Vershynin, 2014; Saade et al., 2014). Once the blocks are estimated, P can be estimated efficiently by a plug-in moment estimator. Many extensions and alternatives to the stochastic block model have been proposed to model networks with communities (Hoff, 2008; Airoldi et al., 2009; Karrer & Newman, 2011; Zhang et al., 2014; Cai et al., 2015), but their properties are generally only known under the correctly specified model with communities, and here we are interested in estimating P for more general networks.

A general representation for the matrix P for unlabeled networks (where any permutation of nodes defines the same network) goes back to Aldous (1981) and Hoover (1979). Formally, a network is *exchangeable* if the distribution of edges is invariant under permutations of node labels. That is, if the adjacency matrix $A = [A_{ij}]$ is drawn from the probability matrix P as described above (which we write as $A \sim P$), then for any permutation π of the set $\{1, \ldots, n\}$,

$$\left[A_{\pi(i)\pi(j)}\right] \sim P \, . \tag{1}$$

Aldous and Hoover showed that an exchangeable network always admits the following representation:

Definition 1.1 (Aldous-Hoover representation)**.** *For any network satisfying* (1)*, there exists a function* $f : [0,1] \times [0,1] \to [0,1]$ *and a set of i.i.d. random variables* $\xi_i \sim Uniform\,[0,1]$*, such that*

$$P_{ij} = f(\xi_i, \xi_j) \, . \tag{2}$$

Following the literature, we call f the graphon function. Unfortunately, f in this representation is neither unique nor identifiable (Diaconis & Janson, 2007), since for any measure-preserving one-to-one transformation $\sigma : [0,1] \to [0,1]$, both $f(\sigma(\cdot), \sigma(\cdot))$ and $f(\cdot, \cdot)$ yield the same distribution of A. A unique canonical representation can be defined if one requires $g(u) = \int_0^1 f(u,v)\mathrm{d}v$ to be non-decreasing (Bickel & Chen, 2009), and it was shown that f and ξ_i's are jointly identifiable when $g(u)$, which can be interpreted as expected node degree, is strictly monotone (Chan & Airoldi, 2014). This assumption is strong and excludes the stochastic block model.

In practice, the main purpose of estimating the function f is to estimate P, and thus identifiability of f or lack thereof may not matter as long as P itself can be estimated. It has been

shown that the measure-preserving map σ is the only source of non-identifiability (Hoover, 1979; Diaconis & Janson, 2007). Wolfe & Olhede (2013) and Choi et al. (2014) proposed estimating f up to a measure-preserving transformation via step-function approximations based on fitting the stochastic block model with a larger number of blocks K. This approximation does not assume the network itself follows the block model, and some theoretical guarantees have been obtained under more general models. In related work, Olhede & Wolfe (2014) proposed to approximate the graphon with "network histograms", that is, stochastic block models with many blocks of equal size, akin to histogram bins. Another method to compute a network histogram was proposed by Amini & Levina (2014), as an application of their semi-definite programming approach to fitting block models with equal size blocks. Quite recently, Gao et al. (2014) established the minimax error rate for estimating P and proposed a least squares type estimator to achieve this rate, which obtains the estimated probability P by averaging the adjacency matrix elements within a given block partition. A similar estimator was proposed in Choi (2015), applicable also to non-smooth graphons. However, these methods are in principle computationally infeasible since they require an exhaustive enumeration of all possible block partitions. Cai et al. (2014) proposed an iterative algorithm to fit a stochastic blockmodel and approximate the graphon, but the error rate of this method for general graphons is not known. A Bayesian approach using block priors proposed in Gao et al. (2015) achieves the minimax error rate adaptively, but it still requires evaluating the posterior likelihood over all possible block partitions to obtain the posterior mode or the expectation for the probability matrix.

Other recent efforts on graphon estimation focus on the case of monotone node degrees, which make the graphon identifiable. The sort and smooth methods (Chan & Airoldi, 2014; Yang et al., 2014) estimate the graphon under this assumption by first sorting nodes by their degrees and then smoothing the matrix A locally to estimate edge probabilities. The monotone degree assumption is crucial for the success of these methods, and as we show later in the paper, the sort and smooth methods perform poorly when it does not hold. Finally, general matrix denoising methods can be applied to this problem if one considers A to be a noisy version of its expectation P; a good general representative of this class of methods is the universal singular value thresholding approach of Chatterjee et al. (2014). Since this is a general method, we cannot expect its error rate to be especially competitive for this specific problem, and indeed its mean squared error rate is slower than the cubic root of the minimax rate.

In this paper, we propose a novel computationally efficient method for probability matrix esti-

mation based on neighborhood smoothing, for piecewise Lipschitz graphon functions. The key to this method is adaptive neighborhood selection, which allows us to avoid making strong assumptions such as monotone node degrees. A node's neighborhood consists of nodes with similar rows in the adjacency matrix, which intuitively correspond to nodes with similar values of the latent node positions ξ_i. To the best of our knowledge, our estimator achieves the best error rate among existing computationally feasible methods. Computationally, the estimator allows easy parallelization. The size of the neighborhood is controlled by a tuning parameter, similar to bandwidth in nonparametric regression; the rate of this bandwidth parameter is determined from theory, and we show empirically the method is robust to the choice of the constant in the tuning parameter. Experiments on synthetic networks demonstrate our method performs very well under a wide range of graphon models (low rank and full rank, with monotone degrees and without, etc). We also test the performance of our method on the link prediction problem, using both synthetic and real networks.

2 The neighborhood smoothing estimator and its error rate

2.1 Neighborhood smoothing for edge probability estimation

Our goal is to estimate the probabilities P_{ij} from the observed network adjacency matrix A, where A_{ij} is drawn from $\mathrm{Bernoulli}(P_{ij})$ and all A_{ij}'s are independent. While $P_{ij} = f(\xi_i, \xi_j)$, where ξ_i's are latent, our goal is to estimate P for the single realization of ξ_i's that gave rise to the data, rather than the function f. We think of f as a fixed unknown smooth function on $[0,1]^2$, with formal smoothness assumptions to be stated later on. Let $e_{ij} = e_{ij}(P_{ij})$ denote the Bernoulli error and omit its dependence on P. We can then write

$$A_{ij} = P_{ij} + e_{ij} = f(\xi_i, \xi_j) + e_{ij}. \tag{3}$$

Formulation (3) resembles a nonparametric regression problem, but with the important difference that ξ_i's are not observed. This has important consequences, for example, assuming further smoothness in f beyond order one does not improve the minimax error rate when estimating P (Gao et al., 2014). The idea of our method is to apply neighborhood smoothing, which would be a natural approach had the latent variables ξ_i's been observed. Intuitively, if we had a set $\mathcal{N}_i$ of neighbors of a node i, in the sense that $\mathcal{N}_i = \{i' : P_{i'\cdot} \approx P_{i\cdot}\}$, where $P_{i\cdot}$ represents the i-th row of P, then we could estimate $P_{i\cdot}$ by averaging $A_{i'\cdot}$ over $i' \in \mathcal{N}_i$. Postponing the question of how to

select $\mathcal{N}_i$ until Section 2.2, we can define a general form of the neighborhood smoothing estimator by

$$\hat{P}_{ij} := \frac{1}{2}\left(\frac{\sum_{i' \in \mathcal{N}_i} A_{i'j}}{|\mathcal{N}_i|} + \frac{\sum_{j' \in \mathcal{N}_j} A_{ij'}}{|\mathcal{N}_j|}\right). \tag{4}$$

It is immediately evident that $\hat{P}$ is symmetric if A is symmetric, although it can be applied to either directed or undirected networks. For simplicity, in this paper we focus on undirected networks. A natural alternative is to average over $\mathcal{N}_i \times \mathcal{N}_j$, but (4) allows vectorization and is thus more computationally efficient. Our estimator can also be viewed as a relaxation of step function approximations such as Olhede & Wolfe (2014). In step function approximations, the neighborhood for each node is the set of nodes from its block, so the neighborhoods for two nodes from the same block are very similar, and the blocks have to be estimated first; in contrast, neighborhood smoothing provides for more flexible neighborhoods that are different from node to node, and an efficient way to select the neighborhood, which we will discuss next.

2.2 Neighborhood selection

Selecting the neighborhood $\mathcal{N}_i$ in (4) is at the core of our method. Since we estimate $P_{i\cdot}$ by averaging over $A_{i'\cdot}$ for $i' \in \mathcal{N}_i$, good neighborhood candidates i' should have $f(\xi_{i'}, \cdot)$ close to $f(\xi_i, \cdot)$, which implies $P_{i'\cdot}$ close to $P_{i\cdot}$. We use the ℓ_2 distance between graphon slices to quantify this, defining

$$d(i, i') := \|f(\xi_i, \cdot) - f(\xi_{i'}, \cdot)\|_2 := \left\{\int_0^1 |f(\xi_i, v) - f(\xi_{i'}, v)|^2 \, \mathrm{d}v\right\}^{1/2} \tag{5}$$

While one may consider more general ℓ_p or other distances, the ℓ_2 distance is particularly easy to work with when it comes to theory. For the purpose of neighborhood selection, it is not necessary to estimate $d(i, i')$; it suffices to provide a tractable upper bound. For integrable functions g_1 and g_2 defined on $[0, 1]$, define $\langle g_1, g_2\rangle = \int_0^1 g_1(u)g_2(u)\mathrm{d}u$. Then we can write

$$d^2(i, i') = \langle f(\xi_i, \cdot), f(\xi_i, \cdot)\rangle + \langle f(\xi_{i'}, \cdot), f(\xi_{i'}, \cdot)\rangle - 2\langle f(\xi_i, \cdot), f(\xi_{i'}, \cdot)\rangle. \tag{6}$$

The third term in (6) can be estimated by $2\langle A_{i\cdot}, A_{i'\cdot}\rangle/n$, where $A_{i\cdot}$ and $A_{i'\cdot}$ are nearly independent (up to a single duplicated entry due to symmetry). The first two terms in (6) are more difficult since $\langle A_{i\cdot}, A_{i\cdot}\rangle/n$ is not a good estimator for $\langle f(\xi_i, \cdot), f(\xi_i, \cdot)\rangle$. Here we present the intuition and provide a full theoretical justification in Theorem 2.2. For simplicity, assume for now f is Lipschitz with a Lipschitz constant of 1. The idea is to use nodes with graphon slices similar to i and i' to make

the terms in the inner product distinct graphon slices. With high probability, for each i, we can find $\tilde{i} \neq i$ such that $|\xi_{\tilde{i}} - \xi_i| \leq e_n$, where $e_n = o(1)$ is the error rate to be specified later. Then we have $\|f(\xi_i, \cdot) - f(\xi_{\tilde{i}}, \cdot)\|_2 \leq e_n$, and we can approximate $\langle f(\xi_i, \cdot), f(\xi_i, \cdot)\rangle$ by $\langle f(\xi_i, \cdot), f(\xi_{\tilde{i}}, \cdot)\rangle$, where the latter can now be estimated by $\langle A_{i\cdot}, A_{\tilde{i}\cdot}\rangle/n$. The same technique can be used to approximate the second term in (6), but all these approximations depend on the unknown ξ's. To deal with this, we rearrange the terms in (6) as follows:

$$\begin{aligned} d^2(i,i') &= \langle f(\xi_i, \cdot) - f(\xi_{i'}, \cdot), f(\xi_i, \cdot)\rangle - \langle f(\xi_i, \cdot) - f(\xi_{i'}, \cdot), f(\xi_{i'}, \cdot)\rangle \\ &\leq \left|\langle f(\xi_i, \cdot) - f(\xi_{i'}, \cdot), f(\xi_{\tilde{i}}, \cdot)\rangle\right| + \left|\langle f(\xi_i, \cdot) - f(\xi_{i'}, \cdot), f(\xi_{\tilde{i'}}, \cdot)\rangle\right| + 2e_n \\ &\leq 2 \max_{k \neq i, i'} |\langle f(\xi_i, \cdot) - f(\xi_{i'}, \cdot), f(\xi_k, \cdot)\rangle| + 2e_n \end{aligned} \tag{7}$$

The inner product on the right side of (7) can be estimated by

$$\tilde{d}^2(i,i') = \max_{k \neq i, i'} |\langle A_{i\cdot} - A_{i'\cdot}, A_{k\cdot}\rangle| / n \,. \tag{8}$$

Intuitively, the neighborhood $\mathcal{N}_i$ should consist of i's with small $\tilde{d}(i,i')$. To formalize this, let $q_i(h)$ denote the h-th sample quantile of the set $\left\{\tilde{d}(i,i') : i' \neq i\right\}$, where h is a tuning parameter, and set

$$\mathcal{N}_i = \left\{i' \neq i : \tilde{d}(i,i') \leq q_i(h)\right\} \tag{9}$$

where for notational simplicity we suppress the dependence of $\mathcal{N}_i$ on h. Thresholding at a quantile rather than at some absolute value is convenient since real networks vary in their average node degrees and other parameters, which leads to very different values and distributions of $\tilde{d}$. Empirically, thresholding at a quantile shows significant advantage in stability and performance compared to an absolute threshold. The choice of h will be guided by both theory, which suggests the order of h (see Section 2.3), and empirical performance which suggests the constant factor (see Section 3.1).

An important feature of this definition is that the neighborhood admits nodes with similar graphon slices, but not necessarily similar ξ's. For example, in the stochastic block model, all nodes from the same block would be equally likely to be included in each other's neighborhoods, regardless of their ξ's. Even though we use ξ_i and $\xi_{i'}$ to motivate (7), we always work with the function values $f(\xi_i, \xi_j)$'s and never attempt to estimate the ξ_i or f by themselves. This sharply contrasts with the approaches of Chan & Airoldi (2014) and Yang et al. (2014), and gives us a substantial computational advantage as well as much more flexibility in assumptions.

2.3 Consistency of the neighborhood smoothing estimator

We study the theoretical properties of our estimator for a family of piecewise Lipschitz graphon functions, defined as follows.

Definition 2.1 (Piecewise Lipschitz graphon family)**.** *For any $\delta, L > 0$, let $\mathcal{F}_{\delta;L}$ denote a family of piecewise Lipschitz graphon functions $f : [0,1]^2 \to [0,1]$ such that (i) there exists an integer $K \geq 1$ and a sequence $0 = x_0 < \ldots < x_K = 1$ satisfying $\min_{0 \leq s \leq K-1}(x_{s+1} - x_s) \geq \delta$, and (ii) both $|f(u_1, v) - f(u_2, v)| \leq L|u_1 - u_2|$ and $|f(u, v_1) - f(u, v_2)| \leq L|v_1 - v_2|$ hold for all $u, u_1, u_2 \in [x_s, x_{s+1}]$, $v, v_1, v_2 \in [x_t, x_{t+1}]$ and $0 \leq s, t \leq K - 1$.*

Then we have the following error rate bound. The proof is included in the appendix.

Theorem 2.2. *Assume that L is a global constant and $\delta = \delta(n)$ depends on n, satisfying $\lim_{n\to\infty} \left(\delta / \sqrt{\frac{\log n}{n}}\right) \to \infty$. Then the estimator $\hat{P}$ defined in (4), with neighborhood $\mathcal{N}_i$ defined in (9) and $h = C\sqrt{\frac{\log n}{n}}$ for any global constant $C \in (0, 1]$, satisfies*

$$\max_{f \in \mathcal{F}_{\delta;L}} \frac{1}{n^2} \|\hat{P} - P\|_F^2 = O_P\left(\sqrt{\frac{\log n}{n}}\right) \tag{10}$$

To the best of our knowledge, this is the best error rate available to date among non-combinatorial cost graphon estimation methods. The minimax error rate $O_P\left(\frac{\log n}{n}\right)$ established by Gao et al. (2014) has (so far) only been achieved by methods that require combinatorial optimizations or evaluations, including Gao et al. (2014), Klopp et al. (2015) and Gao et al. (2015). The rate $O_P\left(\sqrt{\frac{\log n}{n}}\right)$ was also previously only achieved by combinatorial methods, including Wolfe & Olhede (2013) and Olhede & Wolfe (2014). Among computationally efficient methods, the best error rate we are aware of is achieved by singular value thresholding proposed in Chatterjee et al. (2014) at $O_P\left(\sqrt[3]{\frac{1}{n}}\right)$. Additionally, the sort-and-smooth method proposed by Chan & Airoldi (2014) achieves the minimax error rate under the strong assumption that f has strictly monotone expected node degrees $d_f(v) = \int_0^1 f(u, v)\mathrm{d}u$,

Our result also allows for a growing complexity of the graphon function as the network size increases, without affecting the error rate as long as the conditions are satisfied. For example, if all regions where the function is continuous are of equal size, then $K = 1/\delta$ can grow as fast as $K = O\left(\sqrt{\frac{n}{\log n}}\right)$. This almost matches the complexity rate $K = O\left(\sqrt{n \log n}\right)$ for the computationally infeasible ordinary least squares estimator under the stochastic block model (Gao et al., 2014).

3 Probability matrix estimation on synthetic networks

In this section we evaluate the performance of our estimator on two tasks, estimating the probability matrix and link prediction, using synthetic networks. We generate the networks from the four graphons listed in Table 1, selected to have different features in different combinations (monotone degrees or not, low rank or not, etc). These graphons (represented by the corresponding probability matrix P) are also pictured in the first panels of Figures 2 – 5. For all networks, we use $n = 2000$ nodes to generate P from the function f.

Table 1: Synthetic graphons

Graphon	Function $f(u, v)$	Monotone degrees	Rank	Local structure
1	$k/(K+1)$ if $u, v \in ((k-1)/K, k/K)$, $0.3/(K+1)$ otherwise; $K = \lfloor \log n \rfloor$	Yes	$\lfloor \log n \rfloor$	No
2	$\sin\left(5\pi(u+v-1)+1\right)/2 + 0.5$	No	3	No
3	$1 - \left[1 + \exp\left\{15\big(0.8\lvert u-v\rvert\big)^{4/5} - 0.1\right\}\right]^{-1}$	No	Full	No
4	$\left(u^2+v^2\right)/3 \cos\left(1/\left(u^2+v^2\right)\right) + 0.15$	No	Full	Yes

3.1 Choosing the constant factor for the bandwidth

First, we need to choose the quantile cut-off parameter h which controls neighborhood selection. Theorem 2.2 gives the order of h, and the following numerical experiments empirically justify our choice of the constant factor. Figure 1 shows the mean squared error curves for networks with $n = 2000$ nodes generated from the four graphons in Table 1, with the constant factor C varying in the range $\{5^{-4}, 5^{-3.5}, \ldots, 1\}$.

Figure 1 demonstrates that C in the range of $5^{-4} - 5^{-2}$ works equally well for all these very different graphons. This suggests empirically that the method is robust to the choice of C, and therefore we set $C = 0.1 \approx 5^{-2.85}$ for the rest of the paper.

3.2 Comparison with benchmarks

In this experiment, we compare the performance of a number of popular benchmarks for estimating P. From the general matrix denoising methods, we selected the widely used method of universal singular value thresholding (USVT) (Chatterjee et al., 2014) to include in the comparison. We

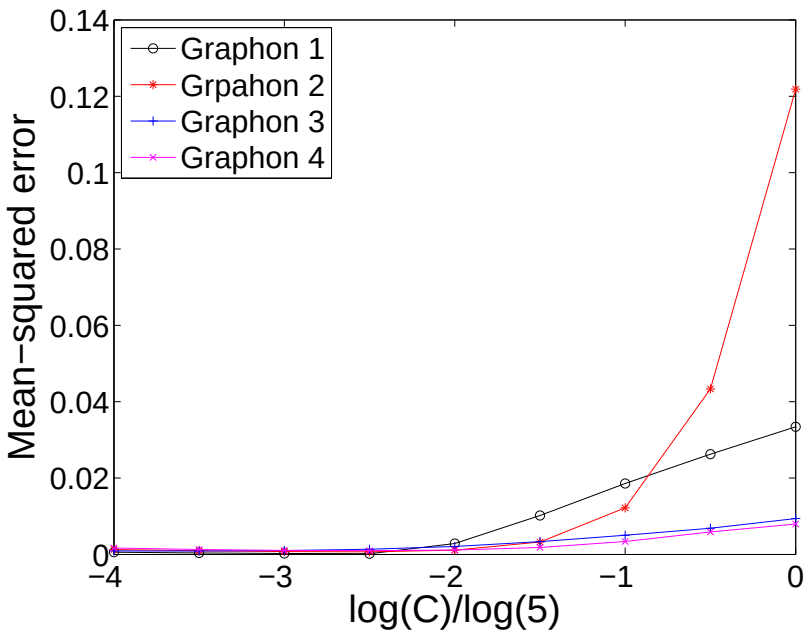

Figure 1: Mean squared error of our method as a function of the constant C in the tuning parameter $h = C\sqrt{\frac{\log n}{n}}$.

also compare to the sort and smooth methods of Chan & Airoldi (2014) and Yang et al. (2014). These two methods are similar, with the difference that the latter method employs singular value thresholding to denoise the network as a pre-processing step. We also include two step function approximations based on fitting a stochastic block model. One is the oracle SBM, where the blocks are formed based on the actual values of the latent ξ_i's. This is obviously not a method that can be implemented in practice, but we use it as the gold standard of what can be achieved with an SBM-based step function approximation at its best. The practical version of this we compare to is step function approximation based on a SBM fit by regularized spectral clustering (Qin & Rohe, 2013). Any other algorithm for fitting the SBM can be used to estimate the blocks; for example, Olhede & Wolfe (2014) used a local updating algorithm initialized with spectral clustering to compute their network histograms. Here we chose regularized spectral clustering because of its speed and good empirical performance. For both SBM-based approximations, we set the number of blocks to $\sqrt{n}$, as proposed by Olhede & Wolfe (2014).

Figure 2 shows the results for Graphon 1. The network contains $\lfloor \log n \rfloor = 7$ blocks with different within-block edge probabilities, which all dominate the low between-block probability. The best results are obtained by our method and the two SBM methods (one of which is the oracle),

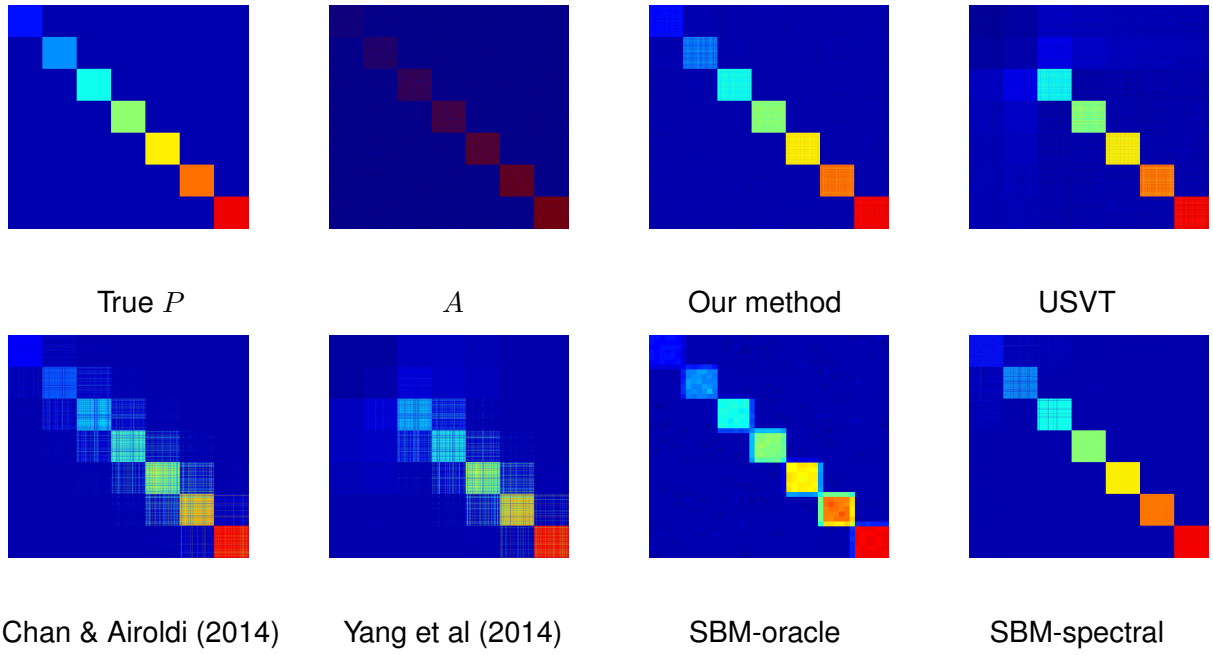

Figure 2: Estimated probability matrices for Graphon 1. The color map is from blue (low) to red (high).

which is expected given that the data are in fact generated from a stochastic block model. The two sort-and-smooth methods correctly estimate the main blocks because the blocks have different expected degrees, but they suffer from boundary effects due to smoothing over the entire region. In contrast, our method, which determines smoothing neighborhoods based on similarities of graphon slices, does not suffer from such boundary effects at all. The USVT method does a good job on blocks with larger expected degrees, but thresholds away sparser blocks; this defect is inherited by the method of Yang et al. (2014), which relies on USVT as pre-processing.

Figure 3 shows the estimation results for Graphon 2. This graphon lacks node degree monotonicity, and thus sort-and-smooth methods do not work here at all. Spectral clustering also performs poorly since the $\sqrt{n}$ eigenvectors it uses turn out to be too noisy. The SBM oracle method gives a grainy but reasonable approximation to P, and the best results are obtained by our method and by USVT, which is expected to work well here since this is a low rank matrix.

Figure 4 shows the estimation results for Graphon 3. Here the probabilities drop off sharply away from the diagonal, and our method captures the main structures but suffers some boundary effects due to smoothing. Nonetheless, it still provides the best approximation, apart from the oracle. The USVT does not perform well because this is not a low rank matrix; spectral clustering, on

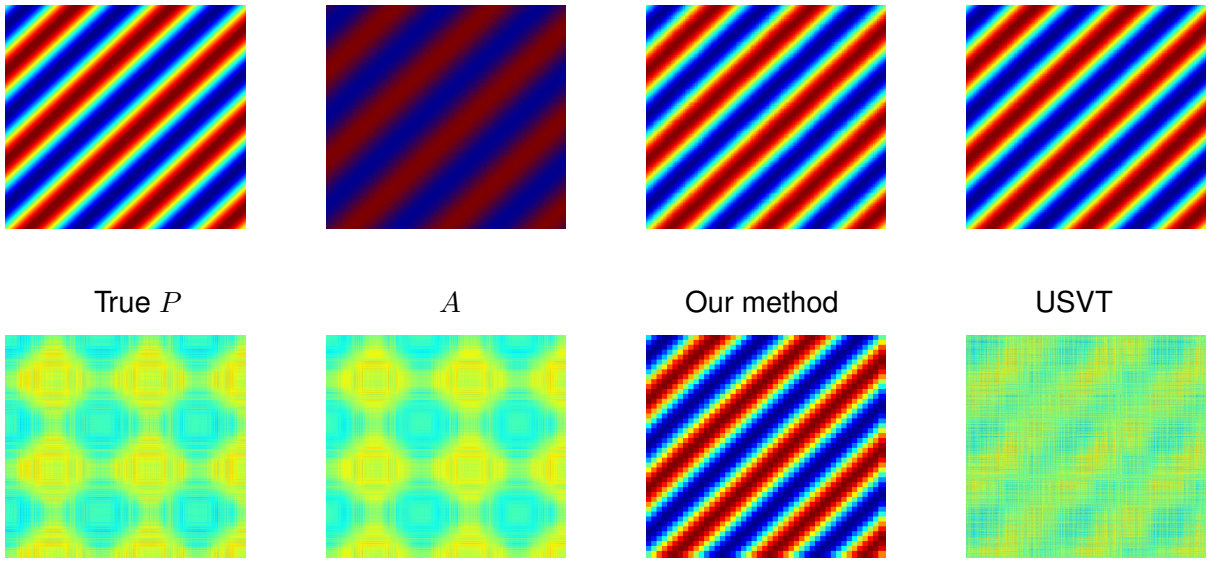

Figure 3: Estimated probability matrices for Graphon 2. The color map is from blue (low) to red (high).

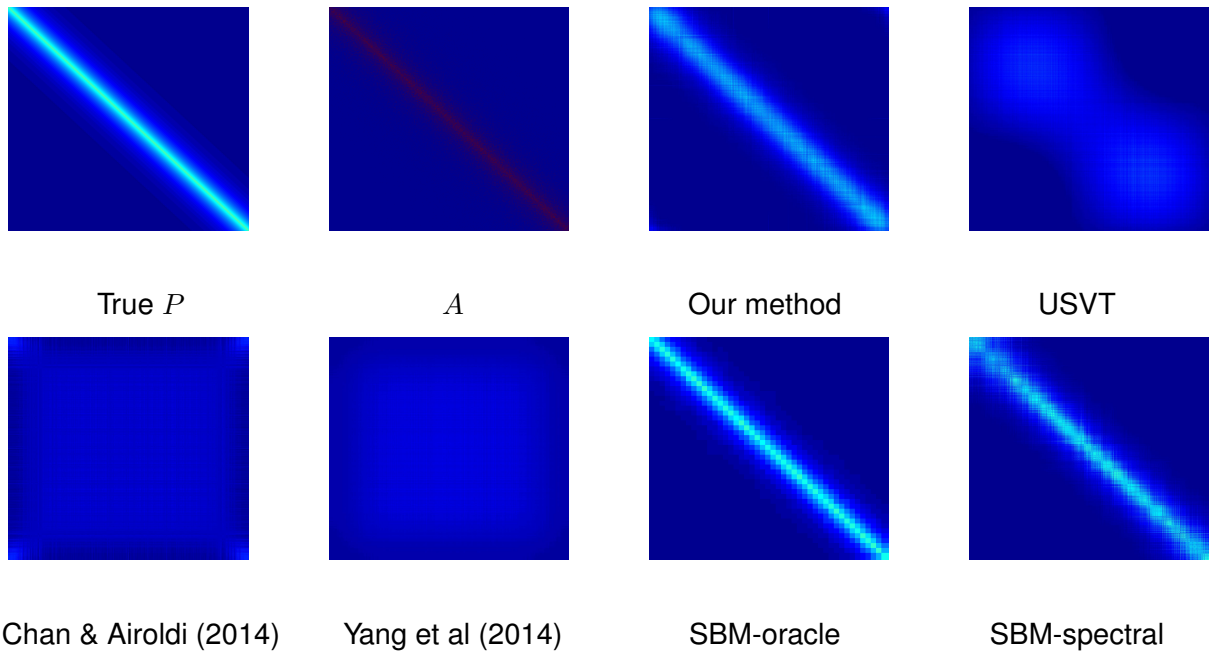

Figure 4: Estimated probability matrices for Graphon 3. The color map is from blue (low) to red (high).

the other hand, does fine, because there are many non-zero eigenvalues and the $\sqrt{n}$ eigenvectors used in spectral clustering contain meaningful information. The sort and smooth methods fail since all node expected degrees are almost the same and the sorting produces nothing but noise.

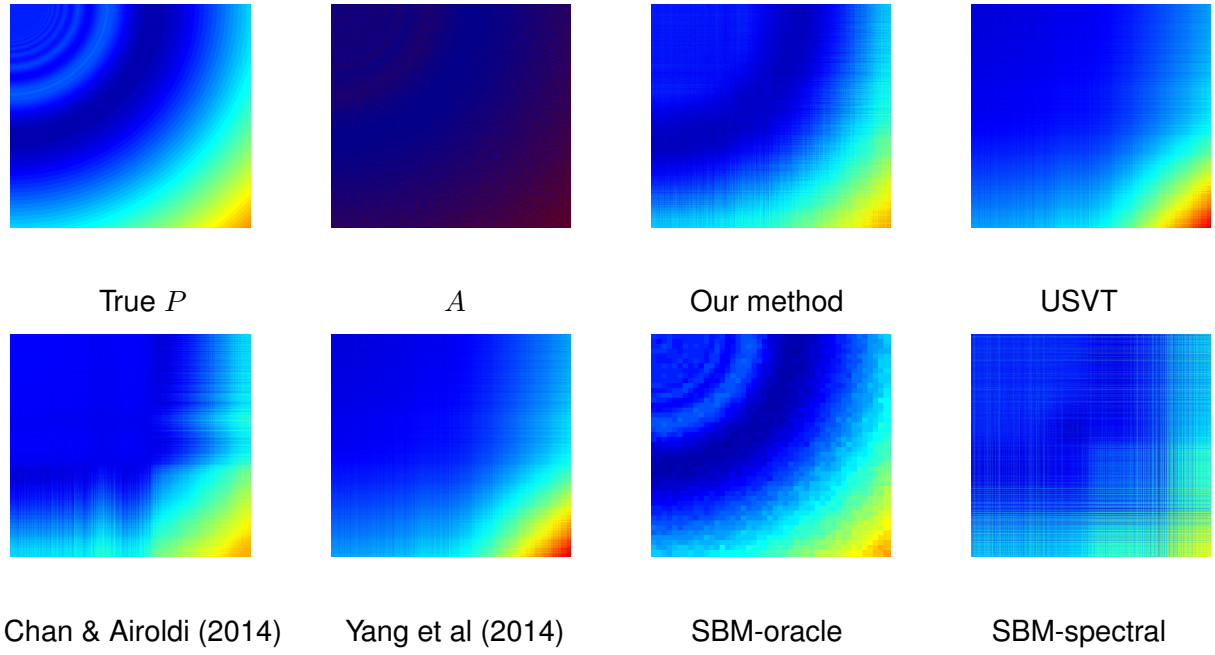

Figure 5: Estimated probability matrices for Graphon 4. The color map is from blue (low) to red (high).

Finally, graphon 4 shown in Figure 5 is difficult to estimate for all methods. The graphon is full rank but with eigenvalues at different scales, and the adjacency matrix tends to have a spectrum very different from the probability matrix. Therefore, this is a very difficult setting for singular value thresholding and spectral clustering. The degrees are monotone for nodes with $\xi \in [0.5, 1]$ but not for $\xi \in [0, 0.5]$, so this graphon is also difficult for the sort and smooth methods, which completely miss the structure in the top left corner of the matrix. Our method successfully picks up the global structure, including non-monotone degrees, though it misses the local variations in the top left corner, as do all other methods except for the oracle approximation. This illustrates a limitation of our method resulting from selecting neighbors based on global similarity of graphon slices, which may miss their local differences.

Overall, the results in this section show that various previously proposed methods can perform very well when their assumptions hold (which may be monotone degrees or low rank or an underlying block model), but they fail when these assumptions are not satisfied. Our method is the

only one among those compared that can perform well in a large range of scenarios, because it learns the structure from data via neighborhood selection instead of imposing a priori structural assumptions.

4 Application to link prediction

Evaluating the performance of probability matrix estimation methods on real networks directly is difficult, since the true probability matrix is unknown. To assess the practical utility of our method, we apply it to the link prediction problem, a practical task that relies on estimating the probability matrix. In this context, we think of the true adjacency matrix A^{true} as unobserved, with binary edges drawn independently according to the probability matrix P, also unobserved. The observed adjacency matrix is defined by $A^{\text{obs}}_{ij} = M_{ij}A^{\text{true}}_{ij}$, where unobserved independent M_{ij}'s $\sim$ Bernoulli$(1-p)$ indicate whether edges are missing and p is the unknown missing rate. A link prediction method usually produces a nonnegative score matrix $\hat{A}$, whose elements represent the estimated propensity of a node pair to form an edge.

We measure link prediction performance by the receiver operating characteristic (ROC) curve defined as follows. For each $t > 0$, we define the false positive rate r_{FP} and the true positive rate r_{TP} by

$$r_{\text{FP}}(t) = \frac{\sum_{ij} 1\left[\hat{A}_{ij} > t, A^{\text{true}}_{ij} = 0, M_{ij} = 0\right]}{\sum_{ij}\left[A^{\text{true}}_{ij} = 0, M_{ij} = 0\right]}$$
$$r_{\text{TP}}(t) = \frac{\sum_{ij} 1\left[\hat{A}_{ij} > t, A^{\text{true}}_{ij} = 1, M_{ij} = 0\right]}{\sum_{ij} 1\left[\hat{A}_{ij} = 1, M_{ij} = 0\right]}$$

Then varying t we obtain the ROC curve.

In this section we include three additional benchmark methods that produce score matrices rather than estimated probability matrices. One standard benchmark is to use the Jaccard index $\langle A_{i\cdot}, A_{j\cdot}\rangle / \{(\sum_k A_{ik})(\sum_k A_{jk})\}$ as the score, see for example Lichtenwalter et al. (2010). The method by Zhao et al. (2013) solves an optimization problem to obtain $\hat{A}_{ij}$ which encourages similar node pairs to have similar predicted scores. The PropFlow algorithm proposed by Lichtenwalter et al. (2010) uses the probability for a random walk starting at one node to reach another node within a certain number of steps as the propensity score. We first compare all methods on simulated networks generated from the graphons in Table 1. We set $n = 2000$ and $p = 10\%$.

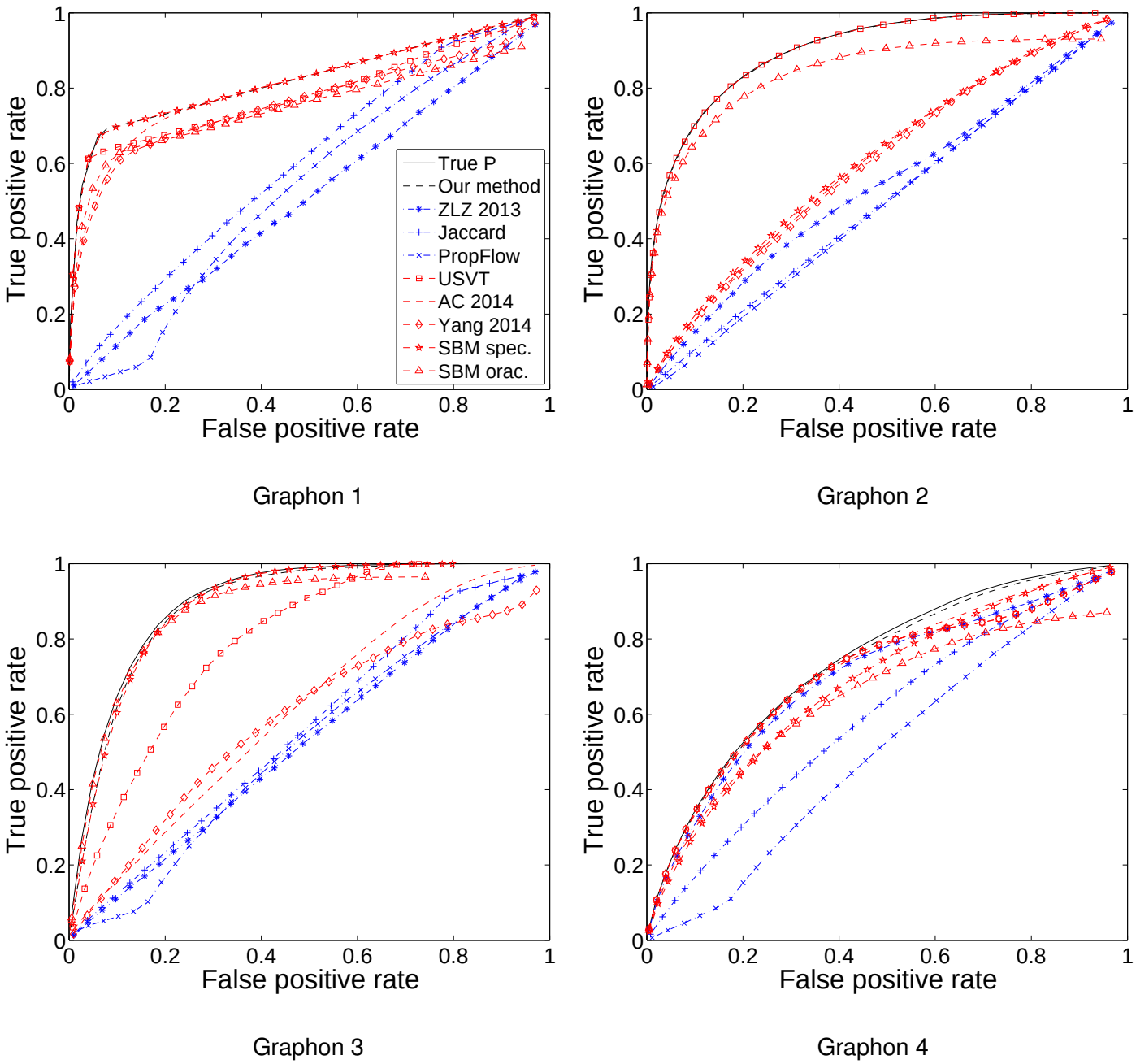

Figure 6: ROC curves for link prediction of different methods under Graphons 1 to 4. All networks have $n = 2000$ nodes, and 10% of edges are missing at random.

Figure 6 shows the ROC curves for four graphons. Most differences between the methods compared in Section 3.2 can be understood from Figures 2 to 5. Overall, the methods based on graphon estimation outperform score-based methods. Our method outperforms all other methods

on this task, producing an ROC curve very close to that based on the true probability matrix P.

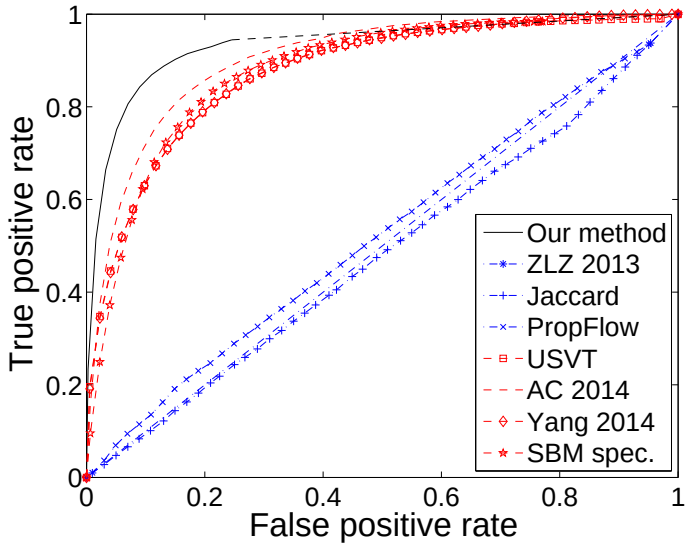

Figure 7: ROC curve for link prediction on the political blogs network. 10% of edges are missing at random.

We also applied our method to the political blogs network (Adamic & Glance, 2005) and compared it to benchmarks. This network consists of 1222 blogs, manually labeled as liberal (586 blogs) or conservative (636 blogs), and the network clearly shows two communities corresponding to these two groups. It also has quite heterogeneous node degrees (some nodes are hubs). We removed 10% of edges at random and then calculated the ROC curve for predicting the missing links, shown in Figure 7. Again, methods based on estimating the probability matrix performed much better than the scoring methods, and our method had the best overall performance. Sort and smooth methods slightly outperformed spectral clustering and the USVT, perhaps due to the presence of hubs. The last point of the ROC curve for our method is at $(\mathrm{FPR}, \mathrm{TPR}) = (0.24, 0.93)$, because removing edges at random from the adjacency matrix for prediction evaluation creates some isolated nodes, and probabilities associated with isolated nodes are estimated to be exactly zero.

5 Discussion

In this paper, we proposed a computationally feasible method to estimate the matrix of edge probabilities from a single network realization under the assumption of a piecewise Lipschitz graphon,

with a competitive mean squared error rate and good empirical performance. The main advantage of our method is the adaptive neighborhood choice which allows for good performance under many different conditions; it is also computationally efficient, very easy to implement, and essentially tuning free. The main limitation of our method is in the piecewise Lipschitz condition, which may lead it to miss small-scale local structures and over-smooth occasionally. Our method does not achieve the minimax error rate, and whether this rate can be achieved by any polynomial time method is, to the best of our knowledge, an open problem. Going forward, a major challenge is to relax the unrealistic assumption of independent edges and resulting exchangeability, extending the model to better describe real world networks.

Appendix: Proof of Theorem 2.2

For convenience, we start with summarizing notation and assumptions made in the main paper.

1. Let $0 = x_0 < x_1 < \ldots < x_K = 1$, $I_k := [x_{k-1}, x_k)$ for $1 \le k \le K-1$ and $I_K = [x_{K-1}, X_K]$. Assume the graphon f is a Lipschitz function on each of $I_k \times I_\ell$ for $1 \le k, \ell \le K$. Let L denote the maximum piece-wise Lipschitz constant.

2. The number of pieces K may grow with n, as long as $\min_k |I_k| / \sqrt{\frac{\log n}{n}} \to \infty$.

For any $\xi \in [0,1]$, let $I(\xi)$ denote the I_k that contains ξ. Let $S_i(\Delta) = [\xi_i - \Delta, \xi_i + \Delta] \cap I(\xi_i)$ denote the neighborhood of ξ_i in which $f(x,y)$ is Lipschitz in $x \in S_i(\Delta)$ for any fixed y. Finally, recall our estimator is defined by

$$\hat{P}_{ij} = \frac{1}{2}\left(\frac{\sum_{i' \in \mathcal{N}_i} A_{i'j}}{|\mathcal{N}_i|} + \frac{\sum_{j' \in \mathcal{N}_j} A_{ij'}}{|\mathcal{N}_j|} \right).$$

We begin the proof of the main theorem with the following decomposition of the mean squared error:

$$\begin{aligned}
&\frac{1}{n^2}\sum_{ij}(\hat{P}_{ij} - P_{ij})^2 = \frac{1}{4n^2}\sum_{ij}\left\{ \frac{\sum_{i' \in \mathcal{N}_i}(A_{i'j} - P_{ij})}{|\mathcal{N}_i|} + \frac{\sum_{j' \in \mathcal{N}_j}(A_{ij'} - P_{ij})}{|\mathcal{N}_j|} \right\}^2 \\
\le &\frac{1}{n^2}\sum_{ij}\left[\frac{1}{2}\left\{ \frac{\sum_{i' \in \mathcal{N}_i}((A_{i'j} - P_{i'j}) + (P_{i'j} - P_{ij}))}{|\mathcal{N}_i|} \right\}^2 + \frac{1}{2}\left\{ \frac{\sum_{j' \in \mathcal{N}_j}((A_{ij'} - P_{ij'}) + (P_{ij'} - P_{ij}))}{|\mathcal{N}_j|} \right\}^2 \right]
\end{aligned} \tag{11}$$

Next, we show how to bound the first term in (11); the second term can be handled similarly. Note that

$$\frac{1}{2}\left\{\frac{\sum_{i'\in\mathcal{N}_i}((A_{i'j}-P_{i'j})+(P_{i'j}-P_{ij}))}{|\mathcal{N}_i|}\right\}^2 \le \left\{\frac{\sum_{i'\in\mathcal{N}_i}(A_{i'j}-P_{i'j})}{|\mathcal{N}_i|}\right\}^2 + \left\{\frac{\sum_{i'\in\mathcal{N}_i}(P_{i'j}-P_{ij})}{|\mathcal{N}_i|}\right\}^2 = J_1(i,j)+J_2(i,j) \tag{12}$$

Our goal is to bound $\frac{1}{n^2}\sum_{ij}\{J_1(i,j)+J_2(i,j)\}$. First, we prove a lemma which estimates the proportion of nodes in a diminishing neighborhood of ξ_i's.

Lemma 5.1. *For arbitrary global constants $C_1, \tilde{C}_1 > 0$, define $\Delta_n = \left(C_1 + \sqrt{\tilde{C}_1+4}\right)\sqrt{\frac{\log n}{n}}$. For n large enough so that $\sqrt{\frac{(\tilde{C}_1+4)\log n}{n}} \le 1$ and $\Delta_n < \min_k |I_k|/2$, we have*

$$\mathbb{P}\left(\min_i \frac{|\{i'\neq i : \xi_{i'}\in S_i(\Delta_n)\}|}{n-1} \ge C_1\sqrt{\frac{\log n}{n}}\right) \ge 1-2n^{-\frac{\tilde{C}_1}{4}}. \tag{13}$$

of Lemma 5.1. For any $0<\epsilon\le 1$ and n large enough to satisfy the assumptions, by Bernstein's inequality we have, for any i,

$$\mathbb{P}\left(\left|\frac{|\{i'\neq i : \xi_{i'}\in S_i(\Delta_n)\}|}{n-1} - |S_i(\Delta_n)|\right| \ge \epsilon\right) \le 2\exp\left(-\frac{\frac{1}{2}(n-1)\epsilon^2}{1+\frac{1}{3}\epsilon}\right) \le 2\exp\left(-\frac{1}{4}n\epsilon^2\right).$$

Taking a union bound over all i's gives

$$\mathbb{P}\left(\max_i \left|\frac{|\{i'\neq i : \xi_{i'}\in S_i(\Delta_n)\}|}{n-1} - |S_i(\Delta_n)|\right| \ge \epsilon\right) \le 2n\exp\left(-\frac{1}{4}n\epsilon^2\right).$$

Letting $\epsilon = \sqrt{\frac{(\tilde{C}_1+4)\log n}{n}}$, we have

$$\mathbb{P}\left(\max_i \left|\frac{|\{i'\neq i : \xi_{i'}\in S_i(\Delta_n)\}|}{n-1} - |S_i(\Delta_n)|\right| \ge \sqrt{\frac{(\tilde{C}_1+4)\log n}{n}}\right) \le 2n^{-\frac{\tilde{C}_1}{4}}. \tag{14}$$

Next we claim that either $[\xi_i-\Delta_n,\xi_i]\subseteq I(\xi_i)$ or $[\xi_i,\xi_i+\Delta_n]\subseteq I(\xi_i)$ holds for all i. If for some i the claim does not hold, by the definition of $I(\xi_i)$, we have $I(\xi_i)\subset[\xi_i-\Delta_n,\xi_i+\Delta_n]$. So we have $|I(\xi_i)|\le 2\Delta_n$, but this contradicts the condition $\Delta_n < \min_k|I_k|/2$. The claim yields that $|S_i(\Delta_n)|\ge\Delta_n$. Finally, by (14), with probability $1-2n^{-\frac{\tilde{C}_1}{4}}$, we have

$$\begin{aligned}\min_i \frac{|\{i'\neq i : \xi_{i'}\in S_i(\Delta_n)\}|}{n-1} &\ge |S_i(\Delta_n)| - \sqrt{\frac{(\tilde{C}_1+4)\log n}{n}} \\ &\ge \Delta_n - \sqrt{\frac{(\tilde{C}_1+4)\log n}{n}} \ge C_1\sqrt{\frac{\log n}{n}}\end{aligned}$$

This completes the proof of Lemma 5.1. □

We now continue with the proof of Theorem 2.2. Recall that we defined a measure of closeness of adjacency matrix slices in Section 2 as

$$\tilde{d}(i,i') = \max_{k\neq i,i'} |\langle A_{i\cdot} - A_{i'\cdot}, A_{k\cdot}\rangle| / n = \max_{k\neq i,i'} \left|(A^2/n)_{ik} - (A^2/n)_{jk}\right| .$$

The neighborhood $\mathcal{N}_i$ of node i consists of nodes (i')'s with $\tilde{d}(i,i')$ below the h-th quantile of $\{\tilde{d}(i,k)\}_{k\neq i}$. The next lemma shows two key properties of $\mathcal{N}_i$.

Lemma 5.2. *Suppose that we select the neighborhood $\mathcal{N}_i$ by thresholding at the lower h-th quantile of $\{\tilde{d}(i,k)\}_{k\neq i}$, where we set $h = C_0\sqrt{\frac{\log n}{n}}$ with an arbitrary global constant C_0 satisfying $0 < C_0 \leq C_1$ for the C_1 from Lemma 5.1. Let $C_2, \tilde{C}_2 > 0$ be arbitrary global constants and assume $n \geq 6$ is large enough so that (i) All conditions on n in Lemma 5.1 are satisfied; (ii) $\sqrt{\frac{(C_2+2)\log n}{n}} \leq 1$; (iii) $C_1\sqrt{n\log n} \geq 4$; and (iv) $\frac{4}{n} \leq \left(\sqrt{C_2+\tilde{C}_2+2} - \sqrt{C_2+2}\right)\sqrt{\frac{\log n}{n}}$. Then the neighborhood $\mathcal{N}_i$ has the following properties:*

1. $|\mathcal{N}_i| \geq C_0\sqrt{n\log n}$.
2. *With probability $1 - 2n^{-\frac{\tilde{C}_1}{4}} - 2n^{-\frac{C_2}{4}}$, for all i and $i' \in \mathcal{N}_i$, we have*

$$\|P_{i'\cdot} - P_{i\cdot}\|_2^2/n \leq \left\{6L\left(C_1 + \sqrt{\tilde{C}_2+4}\right) + 8\sqrt{C_2+\tilde{C}_2+2}\right\}\sqrt{\frac{\log n}{n}}$$

of Lemma 5.2. The first claim follows immediately from the choice of h and the definition of $\mathcal{N}_i$. To show the second claim, we start with concentration results. For any i, j such that $i \neq j$, we have

$$\begin{aligned}
&\left|\left(A^2/n\right)_{ij} - \left(P^2/n\right)_{ij}\right| = \left|\sum_k (A_{ik}A_{kj} - P_{ik}P_{kj})\right| / n \\
\leq &\frac{|\sum_{k\neq i,j}(A_{ik}A_{kj} - P_{ik}P_{kj})|}{n-2} \cdot \frac{n-2}{n} + \frac{|(A_{ii}+A_{jj})A_{ij}| + |(P_{ii}+P_{jj})P_{ij}|}{n} \\
\leq &\frac{|\sum_{k\neq i,j}(A_{ik}A_{kj} - P_{ik}P_{kj})|}{n-2} + \frac{4}{n}
\end{aligned} \tag{15}$$

By Bernstein's inequality, for any $0 < \epsilon \leq 1$ and $n \geq 3$ we have

$$\mathbb{P}\left(\frac{|\sum_{k\neq i,j}(A_{ik}A_{kj} - P_{ik}P_{kj})|}{n-2} \geq \epsilon\right) \leq 2\exp\left(-\frac{\frac{1}{2}(n-2)\epsilon^2}{1+\frac{1}{3}\epsilon}\right) \leq 2\exp\left(-\frac{1}{4}n\epsilon^2\right).$$

Taking a union bound over all $i \neq j$, we have

$$\mathbb{P}\left(\max_{i,j:i\neq j} \frac{|\sum_{k\neq i,j}(A_{ik}A_{kj} - P_{ik}P_{kj})|}{n-2} \geq \epsilon\right) \leq 2n^2\exp\left(-\frac{1}{4}n\epsilon^2\right).$$

Then setting $\epsilon = \sqrt{\frac{(C_2+2)\log n}{n}}$ with n large enough so that $\epsilon \le 1$, we have

$$\mathbb{P}\left(\max_{i,j:i\neq j} \frac{|\sum_{k\neq i,j}(A_{ik}A_{kj} - P_{ik}P_{kj})|}{n-2} \ge \sqrt{\frac{(C_2+2)\log n}{n}} \right) \le 2n^{-\frac{C_2}{4}} \tag{16}$$

Combining (15) and (16), with probability $1 - 2n^{-\frac{C_2}{4}}$, the following holds

$$\max_{i,j:i\neq j} \left| (A^2/n)_{ij} - (P^2/n)_{ij} \right| \le \sqrt{\frac{(C_2+2)\log n}{n}} + \frac{4}{n} \le \sqrt{\frac{(C_2+\tilde{C}_2+2)\log n}{n}} \tag{17}$$

for n large enoug to satisfy (iv).

Next, we prove a useful inequality. For all i and any $\tilde{i}$ such that $\xi_{\tilde{i}} \in S_i(\Delta_n)$, we have

$$\left| (P^2/n)_{ik} - (P^2/n)_{\tilde{i}k} \right| = |\langle P_{i\cdot}, P_{k\cdot}\rangle - \langle P_{\tilde{i}\cdot}, P_{k\cdot}\rangle|/n \le \|P_{i\cdot} - P_{\tilde{i}\cdot}\|_2 \|P_{k\cdot}\|_2/n \le L\Delta_n \tag{18}$$

for all k, where the last inequality follows from

$$|P_{i'\ell} - P_{i\ell}| = |f(\xi_{i'}, \xi_\ell) - f(\xi_i, \xi_\ell)| \le L|\xi_{i'} - \xi_i| \le L\Delta_n$$

for all ℓ, and $\|P_{k\cdot}\|_2 \le \sqrt{n}$ for all k. Note that this holds for all k, including $k = i$ or $k = \tilde{i}$.

We are now ready to upper bound $\tilde{d}(i,i')$ for $i' \in \mathcal{N}_i$. We bound $\tilde{d}(i,i')$ via bounding $\tilde{d}(i,\tilde{i})$ for $\tilde{i}$ with $\xi_{\tilde{i}} \in S_i(\Delta_n)$. By (17) and (18), with probability $1 - 2n^{-\frac{C_2}{4}}$, we have

$$\begin{aligned}\tilde{d}(i,\tilde{i}) = \max_{k\neq i,\tilde{i}} |(A^2/n)_{ik} - (A^2/n)_{\tilde{i}k}| &\le \max_{k\neq i,\tilde{i}} |(P^2/n)_{ik} - (P^2/n)_{\tilde{i}k}| + 2\max_{i,j:i\neq j} |(A^2/n)_{ij} - (P^2/n)_{ij}| \\ &\le L\Delta_n + 2\sqrt{\frac{(C_2+\tilde{C}_2+2)\log n}{n}}\end{aligned} \tag{19}$$

Now since the fraction of nodes contained in $\left|\{\tilde{i} : \xi_{\tilde{i}} \in S_i(\Delta_n)\}\right|$ is at least h, this puts an upper bound on $\tilde{d}(i,i')$ for $i' \in \mathcal{N}_i$, since nodes in $\mathcal{N}_i$ have the lowest h fraction of values in $\{\tilde{d}(i,k)\}_k$. Setting Δ_n as in Lemma 5.1, by Lemma 5.1 and (17), with probability $1 - 2n^{-\frac{\tilde{C}_1}{4}} - 2n^{-\frac{C_2}{4}}$, for all i, at least $C_1\sqrt{\frac{\log n}{n}}$ fraction of nodes $\tilde{i} \neq i$ satisfy both $\xi_{\tilde{i}} \in S_i(\Delta_n)$ and

$$\tilde{d}(i,\tilde{i}) \le L\Delta_n + 2\sqrt{\frac{(C_2+\tilde{C}_2+2)\log n}{n}}\,. \tag{20}$$

Recall that $i' \in \mathcal{N}_i$ have the smallest $h = C_0\sqrt{\frac{\log n}{n}} \le C_1\sqrt{\frac{\log n}{n}}$ fraction of $\tilde{d}(i,i')$'s. Then (20) yields that

$$\tilde{d}(i,i') \le L\Delta_n + 2\sqrt{\frac{(C_2+\tilde{C}_2+2)\log n}{n}} \tag{21}$$

holds for all i and all $i' \in \mathcal{N}_i$ simultaneously with probability $1 - 2n^{-\frac{\tilde{C}_1}{4}} - 2n^{-\frac{C_2}{4}}$.

We are now ready to complete the proof of the second claim of Lemma 5.2. By Lemma 5.1, (17), (18) and (21), with probability $1 - 2n^{-\frac{\tilde{C}_1}{4}} - 2n^{-\frac{C_2}{4}}$, the following holds. For n large enough such that $\min_i |\{i' : \xi_{i'} \in S_i(\Delta_n)\}| \geq C_1\sqrt{n \log n} \geq 4$ (by Lemma 5.1), for all i and $i' \in \mathcal{N}_i$ we can find $\tilde{i} \in S_i(\Delta_n)$ and $\tilde{i}' \in S_{i'}(\Delta_n)$ such that i, i', $\tilde{i}$ and $\tilde{i}'$ are different from each other. Then we have

$$
\begin{aligned}
&\|P_{i\cdot} - P_{i'\cdot}\|_2^2/n = (P^2/n)_{ii} - (P^2/n)_{i'i} + (P^2/n)_{i'i'} - (P^2/n)_{ii'} \\
\leq& \left|(P^2/n)_{ii} - (P^2/n)_{i'i}\right| + \left|(P^2/n)_{i'i'} - (P^2/n)_{ii'}\right| \\
\leq& \left|(P^2/n)_{i\tilde{i}} - (P^2/n)_{i'\tilde{i}}\right| + \left|(P^2/n)_{i'\tilde{i}'} - (P^2/n)_{i\tilde{i}'}\right| + 4L\Delta_n \\
\leq& \left|(A^2/n)_{i\tilde{i}} - (A^2/n)_{i'\tilde{i}}\right| + \left|(A^2/n)_{i'\tilde{i}'} - (A^2/n)_{i\tilde{i}'}\right| + 4\sqrt{\frac{(C_2 + \tilde{C}_2 + 2)\log n}{n}} + 4L\Delta_n \\
\leq& 2\max_{k \neq i,i'} \left|(A^2/n)_{ik} - (A^2/n)_{i'k}\right| + 4\sqrt{\frac{(C_2 + \tilde{C}_2 + 2)\log n}{n}} + 4L\Delta_n \\
=& 2\tilde{d}(i,i') + 4\sqrt{\frac{(C_2 + \tilde{C}_2 + 2)\log n}{n}} + 4L\Delta_n \leq 8\sqrt{\frac{(C_2 + \tilde{C}_2 + 2)\log n}{n}} + 6L\Delta_n \\
=& \left\{6L\left(C_1 + \sqrt{\tilde{C}_2 + 4}\right) + 8\sqrt{C_2 + \tilde{C}_2 + 2}\right\}\sqrt{\frac{\log n}{n}}
\end{aligned}
$$

This completes the proof of Lemma 5.2. □

We are now ready to bound $\frac{1}{n^2}\sum_{ij}\{J_1(i,j) + J_2(i,j)\}$, which will complete the proof of Theorem 2.2. Note that we cannot simply bound each individual $J_1(i,j)$'s by Bernstein's inequality since $A_{i'j}$ is not independent of the event $i' \in \mathcal{N}_i$. Instead, we work with the sum $\frac{1}{n}\sum_j J_1(i,j)$ and decompose it as follows.

$$
\begin{aligned}
\frac{1}{n}\sum_j J_1(i,j) =& \frac{1}{n|\mathcal{N}_i|^2}\sum_j \left\{\sum_{i' \in \mathcal{N}_i}(A_{i'j} - P_{i'j})\right\}^2 \\
=& \frac{1}{n|\mathcal{N}_i|^2}\sum_j \left\{\sum_{i' \in \mathcal{N}_i}(A_{i'j} - P_{i'j})^2 + \sum_{i' \in \mathcal{N}_i}\sum_{i'' \neq i', i'' \in \mathcal{N}_i}(A_{i'j} - P_{i'j})(A_{i''j} - P_{i''j})\right\}. \qquad (22)
\end{aligned}
$$

The first term in (22) satisfies

$$
\sum_j (A_{i'j} - P_{i'j})^2/n = \|A_{i'\cdot} - P_{i'\cdot}\|_2^2/n \leq 1 \qquad (23)
$$

where the inequality is due to $|A_{i'j} - P_{i'j}| \le 1$ for all j. The second term in (22) can be bounded by

$$\begin{aligned}
&\frac{1}{n|\mathcal{N}_i|^2}\sum_j \sum_{i'\in\mathcal{N}_i}\sum_{i''\neq i', i''\in\mathcal{N}_i}(A_{i'j}-P_{i'j})(A_{i''j}-P_{i''j}) \\
\le&\frac{1}{|\mathcal{N}_i|^2}\sum_{i',i''\in\mathcal{N}_i: i'\neq i''}\left|\frac{1}{n}\sum_j (A_{i'j}-P_{i'j})(A_{i''j}-P_{i''j})\right| \\
\le&\frac{1}{|\mathcal{N}_i|^2}\sum_{i',i''\in\mathcal{N}_i: i'\neq i''}\left\{\frac{1}{n-2}\left|\sum_{j\neq i',i''}(A_{i'j}-P_{i'j})(A_{i''j}-P_{i''j})\right|\cdot\frac{n-2}{n}\right. \\
&\left.+\frac{|(A_{i'i''}-P_{i'i''})|\,|(A_{i'i'}-P_{i'i'}+A_{i''i''}-P_{i''i''})|}{n}\right\} \\
\le&\frac{1}{|\mathcal{N}_i|^2}\sum_{i',i''\in\mathcal{N}_i: i'\neq i''}\left\{\frac{1}{n-2}\left|\sum_{j\neq i',i''}(A_{i'j}-P_{i'j})(A_{i''j}-P_{i''j})\right|+\frac{2}{n}\right\}. \qquad (24)
\end{aligned}$$

To bound the first term in (24), for any $i_1 \neq i_2$ and $\epsilon > 0$, by Bernstein's inequality we have

$$\mathbb{P}\left(\frac{1}{n-2}\left|\sum_{j\neq i_1,i_2}(A_{i_1j}-P_{i_1j})\,(A_{i_2j}-P_{i_2j})\right|\ge\epsilon\right)\le 2\exp\left(-\frac{\frac{1}{2}(n-2)\epsilon^2}{1+\frac{1}{3}\epsilon}\right)\le 2n^2e^{-\frac{n\epsilon^2}{4}}.$$

Let $C_3, \tilde{C}_3 > 0$ be arbitrary global constants and let n be large enough so that $\frac{1}{C_0\sqrt{n\log n}}+\frac{2}{n}\le\left(\sqrt{C_3+\tilde{C}_3+8}-\sqrt{C_3+8}\right)\sqrt{\frac{\log n}{n}}$. First, taking $\epsilon=\sqrt{\frac{(C_3+8)\log n}{n}}$ and a union bound over all $i_1 \neq i_2$, we have

$$\mathbb{P}\left(\max_{i_1,i_2,i_1\neq i_2}\frac{1}{n-2}\left|\sum_{j\neq i_1,i_2}(A_{i_1j}-P_{i_1j})\,(A_{i_2j}-P_{i_2j})\right|\ge\sqrt{\frac{(C_3+8)\log n}{n}}\right)\le 2n^{-\frac{C_3}{4}}. \qquad (25)$$

Then plugging (23), (24) and (25) into (22) and combining with claim 1 of Lemma 5.2, with probability $1-2n^{-\frac{\tilde{C}_1}{4}}-2n^{\frac{C_2}{4}}-2n^{-\frac{C_3}{4}}$, for all i simultaneously, we have

$$\begin{aligned}
&\frac{1}{n}\sum_j J_1(i,j)\le\frac{1}{|\mathcal{N}_i|^2}\sum_{i'\in\mathcal{N}_i}\left\{1+(|\mathcal{N}_i|-1)\left(\sqrt{\frac{(C_3+8)\log n}{n}}+\frac{2}{n}\right)\right\} \\
\le&\frac{1}{|\mathcal{N}_i|}+\sqrt{\frac{(8+C_3)\log n}{n}}+\frac{2}{n}\le\frac{1}{C_0\sqrt{n\log n}}+\frac{2}{n}+\sqrt{\frac{(C_3+8)\log n}{n}}\le\sqrt{\frac{(C_3+\tilde{C}_3+8)\log n}{n}}. \qquad (26)
\end{aligned}$$

We now bound $\frac{1}{n^2}\sum_{ij} J_2(i,j)$. By Lemma 5.2, with probability $1 - 2n^{-\frac{\tilde{C}_1}{4}} - 2n^{\frac{C_2}{4}}$, we have

$$\begin{aligned}
&\frac{1}{n^2}\sum_{ij} J_2(i,j) = \frac{1}{n}\sum_i \left\{ \frac{1}{n}\sum_j J_2(i,j) \right\} = \frac{1}{n}\sum_i \left\{ \frac{1}{n}\sum_j \left(\frac{\sum_{i' \in \mathcal{N}_i}(P_{i'j} - P_{ij})}{|\mathcal{N}_i|} \right)^2 \right\} \\
\leq &\frac{1}{n}\sum_i \left\{ \frac{\sum_{i' \in \mathcal{N}_i}\sum_j (P_{i'j} - P_{ij})^2/n}{|\mathcal{N}_i|} \right\} = \frac{1}{n}\sum_i \left\{ \frac{\sum_{i' \in \mathcal{N}_i} \|P_{i'\cdot} - P_{i\cdot}\|_2^2/n}{|\mathcal{N}_i|} \right\} \\
\leq &\left\{ 6L\left(C_1 + \sqrt{\tilde{C}_2 + 4} \right) + 8\sqrt{C_2 + \tilde{C}_2 + 2} \right\} \sqrt{\frac{\log n}{n}}, \qquad (27)
\end{aligned}$$

where the first inequality is the Cauchy-Schwartz inequality and the second inequality follows from claim 2 of Lemma 5.2.

Combining (26) and (27) completes the proof of Theorem 2.2.

References

ADAMIC, L. A. & GLANCE, N. (2005). The political blogosphere and the 2004 US election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery*. pp. 36–43.

AIROLDI, E. M., BLEI, D. M., FIENBERG, S. E. & XING, E. P. (2009). Mixed membership stochastic blockmodels. In *Advances in Neural Information Processing Systems 21*. pp. 33–40.

ALDOUS, D. J. (1981). Representations for partially exchangeable arrays of random variables. *Journal of Multivariate Analysis* **11**, 581–598.

AMINI, A. A., CHEN, A., BICKEL, P. J. & LEVINA, E. (2013). Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics* **41**, 2097–2122.

AMINI, A. A. & LEVINA, E. (2014). On semidefinite relaxations for the block model. *arXiv preprint arXiv:1406.5647* .

BICKEL, P. J. & CHEN, A. (2009). A nonparametric view of network models and Newman–Girvan and other modularities. *Proceedings of the National Academy of Sciences* **106**, 21068–21073.

CAI, D., ACKERMAN, N. & FREER, C. (2014). An iterative step-function estimator for graphons. *arXiv preprint arXiv:1412.2129* .

CAI, T. T., LI, X. et al. (2015). Robust and computationally feasible community detection in the presence of arbitrary outlier nodes. *The Annals of Statistics* **43**, 1027–1059.

CHAN, S. H. & AIROLDI, E. M. (2014). A consistent histogram estimator for exchangeable graph models. *arXiv preprint arXiv:1402.1888* .

CHATTERJEE, S. et al. (2014). Matrix estimation by universal singular value thresholding. *The Annals of Statistics* **43**, 177–214.

CHOI, D. (2015). Co-clustering of nonsmooth graphons. *arXiv preprint arXiv:1507.06352* .

CHOI, D., WOLFE, P. J. et al. (2014). Co-clustering separately exchangeable network data. *The Annals of Statistics* **42**, 29–63.

DIACONIS, P. & JANSON, S. (2007). Graph limits and exchangeable random graphs. *arXiv preprint arXiv:0712.2749* .

GAO, C., LU, Y. & ZHOU, H. H. (2014). Rate-optimal graphon estimation. *arXiv preprint arXiv:1410.5837* .

GAO, C., VAN DER VAART, A. W. & ZHOU, H. H. (2015). A general framework for Bayes structured linear models. *arXiv preprint arXiv:1506.02174* .

GUÉDON, O. & VERSHYNIN, R. (2014). Community detection in sparse networks via Grothendieck's inequality. *arXiv preprint arXiv:1411.4686* .

HOFF, P. (2008). Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in Neural Information Processing Systems 20*. pp. 657–664.

HOOVER, D. N. (1979). Relations on probability spaces and arrays of random variables. *Preprint, Institute for Advanced Study, Princeton, NJ* **2**.

KARRER, B. & NEWMAN, M. E. (2011). Stochastic blockmodels and community structure in networks. *Physical Review E* **83**, 016107.

KLOPP, O., TSYBAKOV, A. B. & VERZELEN, N. (2015). Oracle inequalities for network models and sparse graphon estimation. *arXiv preprint arXiv:1507.04118* .

LICHTENWALTER, R. N., LUSSIER, J. T. & CHAWLA, N. V. (2010). New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. pp. 243–252.

OLHEDE, S. C. & WOLFE, P. J. (2014). Network histograms and universality of blockmodel approximation. *Proceedings of the National Academy of Sciences* **111**, 14722–14727.

QIN, T. & ROHE, K. (2013). Regularized spectral clustering under the degree-corrected stochastic blockmodel. In *Advances in Neural Information Processing Systems 26*. pp. 3120–3128.

ROHE, K., CHATTERJEE, S., YU, B. et al. (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics* **39**, 1878–1915.

SAADE, A., KRZAKALA, F. & ZDEBOROVÁ, L. (2014). Spectral clustering of graphs with the Bethe Hessian. In *Advances in Neural Information Processing Systems*. pp. 406–414.

WOLFE, P. J. & OLHEDE, S. C. (2013). Nonparametric graphon estimation. *arXiv preprint arXiv:1309.5936* .

YANG, J. J., HAN, Q. & AIROLDI, E. M. (2014). Nonparametric estimation and testing of exchangeable graph models. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, vol. 33. pp. 1060–1067.

ZHANG, Y., LEVINA, E. & ZHU, J. (2014). Detecting overlapping communities in networks with spectral methods. *arXiv preprint arXiv:1412.3432* .

ZHAO, Y., LEVINA, E. & ZHU, J. (2013). Link prediction for partially observed networks. *arXiv preprint arXiv:1301.7047* .