On semidefinite relaxations for the block model

Arash A. Amini and Elizaveta Levina

Department of Statistics, University of Michigan

June 24, 2014

Abstract

The stochastic block model (SBM) is a popular tool for community detection in networks, but fitting it by maximum likelihood (MLE) involves an infeasible optimization problem. We propose a new semi-definite programming (SDP) solution to the problem of fitting the SBM, derived as a relaxation of the MLE. Our relaxation is tighter than other recently proposed SDP relaxations, and thus previously established theoretical guarantees carry over, but empirically our approach gives substantially better results. We put all these SDPs into a unified framework and make a connection to the well-known problem of sparse PCA. Our relaxation focuses on the balanced case of a network with equally sized communities, and as we show, that makes it the ideal tool for fitting network histograms, a concept gaining popularity in the graphon estimation literature.

1 Introduction

Community detection, one of the fundamental problems in network analysis, has attracted a lot of attention in a number of fields, including computer science, statistics, physics, and sociology. The stochastic block model (SBM) [1] is a well-established and widely used model for community detection, attractive for its analytical tractability and connections to fundamental properties of random graphs [2, 3, 4], but fitting it to data is a challenge due to the need to optimize over $K^n$ assignments of $n$ nodes to $K$ communities. Many fitting methods have been proposed, including profile likelihood [3], MCMC [5, 6], variational approaches [7, 8, 9], belief propagation [10], and pseudo-likelihood [11], the latter being more or less the current state of the art in speed and accuracy. However, all these methods rely on a good initial value and can be sensitive to starting points. In contrast, spectral clustering methods do not require an initial value, are fast and have also been popular in community detection [12, 13, 14, 15]. Spectral clustering works reasonably well in dense networks with balanced communities but can have difficulties with sparse networks. Regularization can help [13, 11, 16], but even regularized spectral clustering does not achieve the accuracy of likelihood-based methods when they are given a good initial value [11].

Recently, semidefinite programming (SDP) approaches to fitting the SBM have appeared in the literature [17, 18, 19], which rely on a SDP relaxation of the infeasible likelihood optimization problem. They are attractive because, on one hand, they solve a global optimization problem and require no initial value, and on the other hand, they are still maximizing the likelihood and one can therefore hope for better performance than from generic methods like spectral clustering, which does not use the likelihood in any way. As global optimization methods, they
are easier to analyze than iterative methods depending on a starting value. It also appears that SDP relaxations in themselves have a regularization effect, which makes their solutions more robust to noise and outliers (see Remark 1). One drawback of SDP methods is the higher computational cost of SDP solvers. However, by formulating the problem as a SDP, we can benefit from continuous advances in solving large scale SDPs, itself an active area of research.

In this paper, we propose a new SDP relaxation of the likelihood optimization problem, which is tighter than any of the previously proposed SDP relaxations [17, 18, 19]. We also put all these relaxations into a unified framework, by viewing them as relaxations of the MLE restricted to different parameter spaces, and show their connection to the well-studied problem of sparse PCA. Empirically, the tighter relaxation gives better results, and we derive a particular first-order SDP implementation via ADMM which keeps computing costs reasonable.

The particular SDP we propose focuses on the balanced planted partition model, a special case of the SBM which has in itself attracted much study. In particular, our method tends to partition the network into equal-sized blocks, which is a desirable feature in practice, as very large and very small communities are more difficult to interpret. It also makes our method especially suitable for the problem of network histogram estimation [20]. Fitting a SBM can be viewed as a nonparametric approximation to a general mean function of the adjacency matrix, called the graphon, in the same way a piecewise constant function can be used to approximate a reasonably smooth function. Graphon estimation in itself has been a popular topic in recent literature [21, 22], and the network histogram as a graphon estimator has been proposed in [20]. A histogram is appealing because it is controlled by the number of bins (blocks) $K$, which is a single parameter that can be chosen to balance fitting the data with robustness to noise. In this case it is particularly desirable to fit blocks with equal or at least similar number of nodes, as in the usual histogram. Graphon estimation is inherently ambiguous, because graphons are only identifiable up to measure preserving maps, an inevitable consequence of the exchangeability assumption. However, restricting graphon estimators to the class of balanced $K$-block models considerably reduces this ambiguity by collapsing the class of all measure-preserving maps on $[0, 1]^2$ to that of permutations of the $K$ blocks, which is much more acceptable in practice. We show empirically in Section 5 that our SDP relaxation provides the best tool for histogram estimation, as well as generally cleaner solutions, compared to other less tight SDP relaxations and generic methods like spectral clustering.

2 The Stochastic Block model

We now formally introduce the SBM and fix notation. The data is a simple undirected graph on $n$ nodes represented by its adjacency matrix $A$, a binary symmetric matrix with $A_{ii} = 0$ for all $i$ and $A_{ij} = 1$ if there is an edge between nodes $i$ and $j$, and 0 otherwise. Each node belongs to exactly one community, specified by its (latent) membership vector $z_i \in \{0, 1\}^K$, with exactly one nonzero entry, $z_{ik} = 1$ if node $i$ belongs to community $k$. The SBM is parametrized through its probability matrix $B = (b_{kr}) \in [0, 1]^{K \times K}$, where $b_{kr}$ is the probability of an edge forming between a pair of nodes from communities $k$ and $r$. For simplicity, we assume $n$ is a multiple of $K$.

Given $z_i$ and $B$, $\{A_{ij}, i < j\}$ are drawn independently as Bernoulli random variables with means $\mathbb{E}[A_{ij}|z_i, z_j] = z_i^T B z_j$. Let $Z^T = [z_1, \ldots, z_n]$. Then, we can write the model as

$$M_Z := \mathbb{E}[A|Z] = ZBZ^T.$$
$M_Z$ is a block constant, rank $K$ matrix, and we can think of the operation $B \mapsto ZBZ^T$ as a block constant embedding of a $K \times K$ matrix into the space of $n \times n$ matrices. This provides us with a simple but very useful property. For any matrix $M$ and function $f$ on $\mathbb{R}$, let $f \circ M$ be the pointwise application of $f$ to the entries of $M$, $[f \circ M]_{ij} = f(M_{ij})$. Then, we have

$$f \circ (ZBZ^T) = Z(f \circ B)Z^T. \quad (1)$$

Using (1), we can write the log-likelihood of the SBM in a compact form. First note that

$$\ell(Z, B) = \sum_{i<j} A_{ij} \log[MZ]_{ij} + (1 - A_{ij}) \log(1 - [MZ]_{ij})$$

$$= \sum_{i<j} A_{ij} [f \circ MZ]_{ij} + [g \circ MZ]_{ij}$$

where $f(x) := \log \frac{x}{1-x}$ and $g(x) := \log(1-x)$ are functions on $[0, 1]$. For symmetric matrices $A$ and $B$, let $\langle A, B \rangle := \text{tr}(AB)$. Then using (1) and letting $E_n$ be the $n \times n$ matrix of ones, we have

$$2\ell(Z, B) = \langle A, f \circ MZ \rangle + \langle E_n, g \circ MZ \rangle$$

$$= \langle A, Z(f \circ B)Z^T \rangle + \langle E_n, Z(g \circ B)Z^T \rangle. \quad (2)$$

Our focus will be on the following two special cases of the SBM:

(PP) The planted partition (PP) model, PP($p, q$), determined by just two parameters $p$ and $q$ via

$$B = qE_K + (p - q)I_K \quad (3)$$

where $I_K$ is the $K \times K$ identity matrix, and following the PP literature we assume $p > q$.

(PP$^{\text{bal}}$) The balanced planted partition model, PP$^{\text{bal}}$($p, q$), which is PP($p, q$) with the additional assumption that the blocks have equal sizes.

For PP($p, q$) the likelihood greatly simplifies, since $f \circ B$ and $g \circ B$ take only two values,

$$f \circ B = f(q)E_K + [f(p) - f(q)]I_K$$

and similarly for $g \circ B$. Since $ZE_KZ^T = E_n$ (recall the embedding interpretation), (2) becomes

$$2\ell(Z, B) = \langle f(p) - f(q) \rangle \langle A, ZZ^T \rangle + \langle g(p) - g(q) \rangle \langle E_n, ZZ^T \rangle + \text{const.}$$

where the constant term does not depend on $Z$. With the condition $p > q$, we have $f(p) > f(q)$ and $g(p) < g(q)$. Then, we obtain

$$\frac{2\ell(Z, B)}{f(p) - f(q)} = \langle A, ZZ^T \rangle - \lambda \langle E_n, ZZ^T \rangle + \text{const.}, \quad \lambda := \frac{g(q) - g(p)}{f(p) - f(q)} > 0. \quad (4)$$

A similar calculation appears in [19], albeit in a slightly different form.
3 Relaxing the maximum likelihood estimator (MLE)

Given the adjacency matrix $A$, the MLE for $(Z, B)$ is obtained by maximizing the likelihood of the SBM. It is known to have desirable consistency and in some sense optimality properties [3], but the exact computation of the MLE is in general NP-hard, due to the optimization over $Z$. However, it can be relaxed to computationally feasible convex problems.

A class of MLEs can be obtained by restricting the domain over which the likelihood (4) is maximized to different subsets $Z$. That is, we will work with the general estimator

$$\hat{Z} := \arg\max_{Z \in \mathcal{Z}} \langle A, ZZ^T \rangle - \lambda \langle E_n, ZZ^T \rangle.$$  (5)

Each $Z$ corresponds to a co-cluster matrix $X = ZZ^T \in \{0,1\}^{n \times n}$, where $X_{ij} = 1$ if $i$ and $j$ belong to the same community, and $X_{ij} = 0$ otherwise. Any subset $Z$ in the $Z$-space induces a corresponding subset $X$ in the $X$-space. We can alternatively consider estimators of $X$, and our blueprint for deriving different relaxations will be varying the space $X$ in the optimization problem

$$\hat{X} := \arg\max_{X \in \mathcal{X}} \langle A, X \rangle - \lambda \langle E_n, X \rangle.$$  (6)

3.1 Our relaxation (SDP-1)

Our relaxation corresponds to the balanced model PP$^{\text{bal}}(p, q)$, in which each community is of size $n/K$. In this case, all admissible $Z$ can be obtained by permutation of any fixed admissible $Z_0$, and we can take the feasible set $Z$ in (5) to be

$$Z_{\text{orbit}}(Z_0) := \{ Z : Z = PZ_0Q , \ Z_0 = I_K \otimes 1_{n/K} , \ P, Q \text{ are permutation matrices} \},$$

where $\otimes$ is the Kronecker product and $1_{n/K}$ is the vector of all ones of length $n/K$. This choice of $Z_0$ is for convenience and corresponds to assigning nodes consecutively to communities 1 through $K$. Recalling $X = ZZ^T$, the corresponding feasible set in the $X$-space is

$$X_{\text{orbit}}(X_0) := \{ X : X = PX_0P^T , \ X_0 = I_K \otimes E_{n/K} , \ P, Q \text{ are permutation matrices} \}.$$  

Note that $X_0$ is a block diagonal matrix with all the diagonal blocks equal to $E_{n/K}$.

In order to relax $X_{\text{orbit}}(X_0)$, we first note that any $X$ in this set is clearly positive semidefinite (PSD), denoted by $X \succeq 0$, since $X = (PZ_0)(PZ_0)^T$. In addition, $0 \leq X_{ij} \leq 1$ for all $i, j$, which we write as $0 \preceq X \preceq 1$, and $\text{diag}(X) = 1_n$. Note that the latter condition, $X \succeq 0$ and $X \preceq 0$ imply $X \preceq 1$, since $1 - X^2_{ij} = X_{ii}X_{jj} - X^2_{ij} \geq 0$ implying $X_{ij} = |X_{ij}| \leq 1$.

Finally, it is easy to see that each row (column) of $X$ should sum to $n/K$, which can be compactly represented as $X1_n = (n/K)1_n$. Note that we can remove the term $\lambda \langle E_n, X \rangle$ from the objective function in (6), since

$$\langle X, E_n \rangle = \text{tr}(X1_n1_n^T) = 1_n^T X1_n = 1_n^T (n/K)1_n = n^2/K,$$

which is constant. Thus, we arrive at our proposed relaxation, which we will call SDP-1:

$$\arg\max_X \langle A, X \rangle \quad \text{subject to} \quad X1_n = (n/K)1_n, \quad \text{diag}(X) = 1_n, \quad X \succeq 0, \ X \preceq 0.$$  (8)
3.2 Other SDP relaxations (SDP-2 and SDP-3)

Two other interesting SDP relaxations have recently appeared in the literature. First, we will consider the relaxation of Chen & Xu [18] (see also [17]). They essentially work with the same PP$_{bal}(p,q)$, although their model is slightly more general (see Remark 1). The main relaxation proposed in [18] is via constraining the nuclear norm of $X$, based on it being a good heuristic for constraining the rank. Since $X$ is PSD, we obtain $\|X\|_* = \text{tr}(X) = n$. In addition, they impose a single affine constraint, namely (7). Thus, their main focus is on the relaxation which replaces $X_{\text{orbit}}$ with $\{X : \|X\|_* \leq n, \langle X, E_n \rangle = n^2/K, 0 \leq X \leq 1\}$. However, they briefly mention a much tighter SDP relaxation which imposes PSDness directly. This is what we have called SDP-2, shown in Table 1.

Note that $X \succeq 0$ and $\text{tr}(X) = n$ imply $\|X\|_* = n$, which is much tighter than $\|X\|_* \leq n$. The main difference between SDP-2 and our relaxation is that we impose the constraint $\langle E_n, X \rangle = n^2/K$ more restrictively, by breaking it into $n$ separate affine constraints. We also break the $\text{tr}(X) = n$ into $n$ pieces, but that does not seem to be a major difference.

Next, we consider the relaxation of Cai & Li [19], though in a slightly different form. This relaxation works for the more general (not necessarily balanced) model PP$(p,q)$. In this case, we are looking at the feasible set

$$X_{\text{free}} = \{X = ZZ^T : Z \text{ is an admissible membership matrix}\}. \quad (9)$$

For $X \in X_{\text{free}}$, we still have $X \succeq 0$ and $X_{ij} \in \{0,1\}$. Thus, one can simply relax to the problem denoted by SDP-3 in Table 1.

Note that $\lambda \langle E_n, X \rangle$ remains in the objective, since there are no constraints to make it constant. We cannot enforce an affine constraint involving $\langle E_n, X \rangle$ directly for $X_{\text{free}}$ without knowing the block sizes. In fact, let $\mathbf{n} = (n_1, \ldots, n_K)$ be the vector of block sizes, and let $E_\mathbf{n} := \text{diag}(E_{n_1}, \ldots, E_{n_K})$ be the block-diagonal matrix with diagonal blocks of all ones with sizes given by $\mathbf{n}$. Then, it is easy to see that $X_{\text{free}}$ is the union of orbits of all possible $E_\mathbf{n}$, that is,

$$X_{\text{free}} = \bigcup_{\mathbf{n}} X_{\text{orbit}}(E_\mathbf{n}) = \bigcup_{\mathbf{n}, \|\mathbf{n}\|_1 = n} \{X : X = PE_\mathbf{n}P^T \text{ for a permutation matrix } P\} \quad (10)$$

from which it follows that $\langle E_n, X \rangle = \|\mathbf{n}\|_2^2 = \sum_j n_j^2$, a function of the unknown $\{n_j\}$.

The optimal value for parameter $\lambda$, assuming the model is PP$(p,q)$, is given in (4) as a function of $p$ and $q$. However, one can think of $\lambda$ as a general regularization parameter controlling the sparseness of $X$, by noting that $\langle E_n, X \rangle = \|X\|_1$ since $X \succeq 0$. It is well known that the $\ell_1$ norm is a good surrogate for a cardinality constraint when enforcing sparseness, which leads us to a link to sparse PCA discussed in Section 3.3.

Remark 1. Both [18] and [19] consider the effect of outliers on their SDPs. Cai & Li [19] derive the SDP for the model we described but they modify it by penalizing the trace, which is justified by their theory for a fairly general model of outliers. Chen & Xu [18] start with a generalized version of PP$_{bal}(p,q)$ which allows for a subset of nodes that belong to no community, and relax that model. Our relaxation SDP-1 can also work for this generalized model if we replace $X_{1_n} = (n/K)1_n$ with the inequality version $\|X_{1_n}\|_1 \leq (n/K)1_n$. This has an advantage relative to Chen & Xu’s approach, in that one does not need to know the number of outliers a priori.
Table 1: Three SDP relaxations

|      | SDP-1 | SDP-2 | SDP-3               |
|------|-------|-------|--------------------|
| max  | $\langle A, X \rangle$ | $\langle A, X \rangle$ | $\langle A, X \rangle - \lambda \langle E_n, X \rangle$ |
| sub  | $X1_n = (n/K)1_n$ | $\langle E_n, X \rangle = n^2/k$ |                                                |
| to   | $\text{diag}(X) = 1_n$ | $\text{tr}(X) = n$ |                                                |
|      | $X \succeq 0$ | $0 \leq X \leq 1$ |                                                |
| model| $\text{PP}^{\text{bal}}(p, q) \equiv \mathcal{X}_{\text{orbit}}(X_0)$ | $\text{PP}(p, q) \equiv \mathcal{X}_{\text{free}}$ |                                                |

3.3 Connection with nonnegative sparse PCA

Representation (10) suggests another natural direction to restrict the parameter space. Note that $\|n\|_\infty = \max_j n_j \in [n/K, n]$ as a consequence of $\|n\|_1 = n$. The closer $\|n\|_\infty$ is to $n/K$, the more balanced the communities are. This suggests the following class,

$$\mathcal{X}_{\text{free}}^\gamma := \bigcup \{ \mathcal{X}_{\text{orbit}}(E_n) : \|n\|_1 = n, \|n\|_\infty \leq \gamma(n/K) \},$$

where $\gamma \in [1, K]$ measures the deviation from completely balanced communities. Let $\|X\|_2$ be the $\ell_2$-operator norm of $X \in \mathcal{X}_{\text{free}}^\gamma$. We can relax this class by noting that $\|X\|_2 = \|E_n\|_2 = \max_j \|E_{nj}\|_2 = \|n\|_\infty \leq \gamma(n/K)$. As before, we have $\text{tr}(X) = n$, $\|X\|_1 = \langle E_n, X \rangle$, and $X \in \mathcal{N}^n := \{ X : X \succeq 0, X \geq 0 \}$, the doubly nonnegative cone. Letting $\bar{X} = (K/n)X$, we have

$$\begin{align*}
\argmax_{\bar{X}} & \quad \langle A, \bar{X} \rangle - \lambda \|\bar{X}\|_1 \\
\text{subject to} & \quad \|\bar{X}\|_2 \leq \gamma, \\
& \quad \text{tr}(\bar{X}) = K, \\
& \quad \bar{X} \succeq 0, \bar{X} \geq 0.
\end{align*}$$

Apart from the nonnegative constraint $\bar{X} \geq 0$ (which can be removed to obtain a further relaxation), this is a generalization of the SDP relaxation for sparse PCA. Specifically, $\gamma = 1$ corresponds to the now well-known relaxation for recovering a sparse $K$-dimensional leading eigenspace of $A$. The corresponding solution $\bar{X}$ can be considered a generalized projection into this subspace, see for example [23, 24], and note that $\bar{X} \succeq 0, \|\bar{X}\|_2 \leq 1$ is equivalent to $0 \preceq \bar{X} \preceq 1$. We will not pursue this direction here, but it opens up possibilities for leveraging sparse PCA results in network models.

3.4 Implementation of SDP-1

It is straightforward to adapt a first order method to solve the SDP-1 problem (8), so that it is reasonably scalable. Here, we briefly discuss the implementation of an ADMM solver [25]. One can start by rewriting the problem as

$$\min_X \left\{ -\langle A, X \rangle + \delta_{\{A(X)=0\}} + \delta_{\{Z \geq 0\}} + \delta_{\{Y \geq 0\}} \right\} \quad \text{s.t. } X = Z, \ X = Y,$$
Finally, let
\[ L \] be a symmetric matrix with 1 in the off-diagonal elements of the \( i \)-th column and row, and 0 everywhere else. \( F_i \) is a matrix with element \((i, i)\) equal to 1 and 0 everywhere else. Finally, \( b_i = 2((n/K) - 1) \) for \( i = 1, \ldots, n \) and \( b_i = 1 \) otherwise.

The only real work in deriving ADMM updates is to find the projection operator \( \Pi_L \) for \( L := \{ X : A(X) = b \} \). For any \( Y \), this projection is given by
\[
\Pi_L(Y) := Y - A^*(AA^*)^{-1}[A(Y) - b].
\]
Note that \( \langle H_i, F_j \rangle = 0 \) for all \( i, j = 1, \ldots, n \). Hence, \( AA^* \) is block diagonal with two blocks \((\langle H_i, H_j \rangle) = 2[(n - 2)I_n + 1_n1_n^T] \) and \((\langle F_i, F_j \rangle) = I_n \). It follows that
\[
(AA^*)^{-1} = \text{diag} \left( \frac{1}{2(n - 2)}[I_n - \frac{1_n1_n^T}{2n - 2}], I_n \right).
\]
We also have \( A^*(\mu, \nu) = \sum_i \mu_i E_i + \sum_i \nu_i F_i = (\mu_i + \mu_j)_{i \neq j} + \text{diag}(\nu) \), which gives a complete recipe to compute \( \Pi_L(Y) \). Note that due to the simplicity of \( (AA^*)^{-1} \) and \( A^* \), implementing this projection has essentially the same cost as projecting onto an affine set with two constraints \( \{ X : \text{tr}(X) = n \}, \langle E_n, X \rangle = n^2/K \} \), which is needed for implementing SDP-2.

The ADMM updates are easily derived to be
\[
\begin{align*}
X^{k+1} &= \Pi_L \left( \frac{1}{2}(Z^k - U^k + Y^k - V^k + \frac{1}{\rho}A) \right), \\
Z^{k+1} &= \max \{0, X^{k+1} + U^k\}, \\
Y^{k+1} &= \Pi_{S^+_n}(X^{k+1} + V^k), \\
U^{k+1} &= U^k + X^{k+1} - Z^{k+1}, \\
V^{k+1} &= V^k + X^{k+1} - Y^{k+1},
\end{align*}
\]
where \( \Pi_{S^+_n} \) is the projection onto the PSD cone \( S^+_n \), which can be done by truncating to non-negative eigenvalues. The ADMM updates for SDP-2 and SDP-3 can be derived similarly, as in [19].

### 3.5 Theoretical guarantees for SDP-1

Since our relaxation is tighter than SDP-2, theoretical guarantees established in [18] automatically apply to SDP-1. More precisely, for model PP\textsuperscript{bal}(p, q), SDP-1 recovers the true \( X \) with high probability if \( (p - q)^2 \frac{n^2}{\bar{p}K} \gtrsim p \frac{n}{K} \log n + qn \). A slightly weaker condition is in fact sufficient, namely
\[
(\bar{p} - \bar{q})^2 \gtrsim (\bar{p} + \bar{q}K) \log n \iff \lambda \gtrsim \left[ \frac{1 + K\beta}{1 - \beta} \right]^2 \log n,
\]
where \( \bar{p} := p \frac{n}{K}, \bar{q} := q \frac{n}{K} \) and \((\lambda, \beta)\) is an alternative parametrization in terms of the expected degree \( \lambda \asymp \bar{p} + K\bar{q} \) and the out-in-ratio \( \beta := q/p = \bar{q}/\bar{p} \). In particular, for fixed \( \beta \), it is enough to have \( \lambda = O(K^2 \log n) \) for SDP-1 to succeed. Interestingly, exactly the condition (12) was established in [19] for SDP-3, specialized to PP\textsuperscript{bal}(p, q). In fact, similar conditions (amounting to \( (\bar{p} - \bar{q})^2 \gtrsim K^2\bar{p} \) and \( np = K\bar{p} \geq \log n \)) can be found in [14] for spectral clustering based on the adjacency matrix, which we will call eigenvalue truncation (EVT). Current theoretical results essentially coincide for all these estimators, but empirically they exhibit quite different finite sample behavior.
4 Application to network histograms

As mentioned earlier, the balanced model $\text{PP}^{\text{bal}}(p,q)$ is ideally suited for computing network histograms as defined by [20]. Graphon estimation requires estimating the mean matrix $M_Z$, which is fairly straightforward once we have a good estimate of the co-cluster matrix $X$. Algorithm 1 details the procedure based on eigenvalue truncation and $K$-means. Note that the permutation ambiguity is resolved by fixing a particular ordering in steps 4–5. In a slight abuse of notation, we will call the output, $\hat{M}_Z$, the graphon estimator corresponding to $\hat{X}$, since it contains all the information for a block-constant function estimate of the graphon.

The graphon estimator can be computed from any estimate $\hat{X}$, but in practice SDP-1 has advantages over other ways of estimating $\hat{X}$. The general likelihood based estimators discussed in the introduction have no way of enforcing equal number of nodes in each block, whereas our empirical results in Section 5) show that the SDP-1 has a high tendency to form equal-sized blocks, more so than SDP-2, making it an ideal choice for performing network histogram analysis. Note that SDP-3 is not well suited for this task since it cannot enforce either a particular number of blocks or a particular block size. It is more flexible due to the tuning parameter $\lambda$, but that flexibility is a disadvantage for constructing histograms.

Algorithm 1 Graphon estimation by fitting $\text{PP}^{\text{bal}}(p,q)$

**Input:** Estimate $\hat{X}$ of co-cluster matrix, and number of blocks $K$.

**Output:** Graphon estimator $\hat{M}_Z$.

1. Form the eigendecomposition $\hat{X} = \hat{U}\Lambda\hat{U}^T$ and set $\hat{U}^K \leftarrow \hat{U}(1:1;K)$.
2. Apply $K$-means to rows of $\hat{U}^K$ to get a label vector $e \in [K]^n$. Set $\hat{Z}(i,e(i)) \leftarrow 1$, otherwise 0.
3. Set $\hat{B}_{rk} \leftarrow \frac{1}{n} \sum_{i,r,e_i=k} A_{ij}$ for $r \neq k$ and $\frac{1}{n(n-1)} \sum_{i,e_i=r} A_{ij}$ otherwise.
4. Permute $\hat{B} \leftarrow Q\hat{B}Q^T$ so that its diagonal is decreasing. Set $Z \leftarrow ZQ^T$.
5. Permute $\hat{Z} \leftarrow P\hat{Z}$ so that corresponding labels are in increasing order.
6. Set $\hat{M}_Z \leftarrow \hat{Z}\hat{B}\hat{Z}^T$.

5 Numerical Results

Here we report experimental results, comparing SDP-1 with SDP-2, SDP-3, and EVT, which amounts to spectral clustering on the adjacency matrix $A$. We chose EVT rather than a version based on the graph Laplacian because SDPs also operate on $A$ itself. For SDP-3, we set the tuning parameter $\lambda$ to the optimal value recommended by [19]. We show results for the balanced symmetric model $\text{PP}^{\text{bal}}(p,q)$, reparametrized in terms of the average expected degree $\lambda = p(\frac{nK}{K} - 1) + q\frac{K}{n}\frac{1}{K} - 1$ and the out-in-ratio $\beta = q/p < 1$. Estimation becomes harder when $\lambda$ decreases (fewer edges) and when $\beta$ increases (communities are not well separated). Figure 1 shows typical behavior of the four methods compared, for a network of size $n = 60$ with $K = 3$ communities, $\lambda = 5$, and $\beta = 0.05$. The first row shows the input adjacency matrix and the resulting estimates of the co-cluster matrix $\hat{X}$ by different algorithms. In the second row, the matrices are permuted according to the ordering computed by Algorithm 1, which shows clear block structure. One can see that while all algorithms recover the block structure to some extent, the output of SDP-1 provides the cleanest solution, consistent with it being the tightest relaxation. This can also be seen from the scatter plots of rows of $\hat{U} \in \mathbb{R}^{n \times 3}$ shown in the bottom row, whose columns are the three leading eigenvectors of $\hat{X}$. The eigenvectors corresponding to SDP-1 clearly provide the best separation, and in fact for sufficiently small $\beta$ SDP-1 often recovers the exact $X$ or a
very minor variant of it, as seen in the figure. We have found empirically that the three SDP-1 eigenvectors are much more geometrically organized and always produce a planar figure in the 3D space, with points concentrating around vertices of a triangle. We also note that all the SDPs are far superior to EVT.

Figure 2 shows the agreement of estimated labels with the truth in various settings, as measured by the normalized mutual information (NMI) averaged over 25 Monte Carlo replications. NMI takes values between 0 and 1, with 0 corresponding to random guessing and 1 to a perfect match. The label estimates are obtained from \( \hat{X} \) by Algorithm 1. As expected, here the SDPs are ranked according the tightness of relaxation, with SDP-1 dominating the other two, and all SDPs outperform EVT.

Figure 3 shows the result of application of SDPs and EVT to the dolphins network [26], with \( K = 3, 10 \). For SDP-3, we use the median connectivity to set \( \lambda \) as suggested in [19]. The adjacency matrices in row 1 and the graphon estimators in row 2 are both permuted according to the ordering from Algorithm 1. The SDPs again provide a much cleaner picture than EVT, and
in addition, SDP-1 forms blocks of nearly equal size. We have also compared results with $K = 2$ to the partition suggested by Fig. 1(b) in [26]. SDP-1, SDP-2, SDP-3, and EVT misclassify 7, 1, 4, and 11 nodes, respectively, out of 62. Since the “true” partition here has unbalanced blocks (20 and 42), we expect SDP-1 might not do as well. However, if we replace equality constraints with the inequality ones as discussed in Remark 1, SDP-1 misclassifies only 2 nodes. Also note that EVT gives very poor representations of this network.

![Image of network results](Figure 3: Results for the dolphins network for $K = 3$ (a–d) and $K = 10$ (e–h). Row 1: adjacency matrix sorted according to the permutation of Algorithm 1. Row 2: Graphon estimator $\hat{M}_2$ of Algorithm 1. Row 3: Scatter plot of $\hat{U}_K$ from Algorithm 1.)

In summary, the simulations and the dolphins network example show that SDP-1 produces the best estimates for the PP\(_\text{bal}\)($p, q$) model, as we would expect from the tightest relaxation, and provides us with the best tool for fitting block models with equal block sizes, which is needed in network histogram estimation. It can also be made more flexible by relaxing the equality constraints if unbalanced blocks are desirable.

6 Discussion

Further theoretical analysis is needed to understand the behavior of different estimators. Current theory available applies equally to all the estimators we considered, as well as to regularized spectral clustering [13, 15, 16], but in practice these estimators behave very differently. Empirically SDP-1 and SDP-2 degrade much slower than EVT, at a similar rate but with a gap between them. Whether the gap remains asymptotically as $n \to \infty$ and/or as one approaches the threshold of non-detectability is an interesting open question. Our conjecture is that the gap remains if $K$ does not grow too fast with $n$, otherwise it vanishes. The comparison with SDP-3 is harder to make directly as it depends on the tuning parameter. However, we note that Lagrange duality implies that for every $A$ and $K$ there exists a $\lambda$ that makes SDP-3 equivalent to SDP-2. Also, SDP-3 is not suitable for histogram estimation.

While we focused on the special case of the planted partition model PP($p, q$), this is not as restrictive as it seems. For a general edge probability matrix $B$, let $p^− := \min_i B_{ii}$ and $q^+ := \max_{i \neq j} B_{ij}$. Then, as long as $p^− > q^+$, fitting the more general model is no harder than fitting PP($p^−, q^+$), as argued for SDP-2 in [19]. Nonetheless, an interesting direction
for future work is finding other SDP relaxations that retain the flexibility of a general SBM, and in particular can accommodate both assortative and disassortative communities within one network. Relaxing equality constraints in SDP-1 may make it more robust to outliers, as we saw in the dolphins example, and this alternative relaxation also warrants further investigation. In fact, there is another robust version, a hybrid between SDP-1 and SDP-3, which is obtained by replacing equality constraints with \( X1_n = (n/K)1_n - \nu, \nu \geq 0 \) and the objective function with \( \langle A, X \rangle + \lambda \langle 1_n, \nu \rangle \) as suggested by (6). Another future direction is exploring further the connection to sparse PCA, in particular the continuous tuning between the balanced and the fully unbalanced cases allowed by the tuning parameter \( \gamma \). Finally, any future advances in large scale SDP solvers will help improve scalability of all the SDP relaxations.

References

[1] P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: first steps. Social Networks, 5(2):109–137, 1983.

[2] D. J. Aldous. Representations for partially exchangeable arrays of random variables. J. Multivariate Analysis, 11:581–598, 1981.

[3] P. J. Bickel and A. Chen. A nonparametric view of network models and Newman-Girvan and other modularities. Proc. Natl. Acad. Sci. USA, 106:21068–21073, 2009.

[4] E. Mossel, J. Neeman, and A. Sly. Stochastic block models and reconstruction. arXiv:1202.1499, 2012.

[5] T. Snijders and K. Nowicki. Estimation and prediction for stochastic block-structures for graphs with latent block structure. Journal of Classification, 14:75–100, 1997.

[6] K. Nowicki and T. A. B. Snijders. Estimation and prediction for stochastic blockstructures. Journal of the American Statistical Association, 96(455):1077–1087, 2001.

[7] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. J. Machine Learning Research, 9:1981–2014, 2008.

[8] A. Celisse, J.-J. Daudin, and L. Pierre. Consistency of maximum-likelihood and variational estimators in the stochastic block model. Electronic Journal of Statistics, 6:1847–1899, 2012.

[9] P. J. Bickel, D. Choi, X. Chang, and H. Zhang. Asymptotic normality of maximum likelihood and its variational approximation for stochastic blockmodels. Annals of Statistics, 41:1922–1943, 2013.

[10] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. Physical Review E, 84:066106, 2012.

[11] A.A. Amini, A. Chen, P.J. Bickel, and E. Levina. Fitting community models to large sparse networks. Annals of Statistics, 41(4):2097–2122, 2013.

[12] K. Rohe, S. Chatterjee, and B. Yu. Spectral clustering and the high-dimensional stochastic block model. Annals of Statistics, 39(4):1878–1915, 2011.

[13] K. Chaudhuri, F. Chung, and A. Tsiatas. Spectral clustering of graphs with general degrees in the extended planted partition model. JMLR Workshop and Conference Proceedings, 23:35.1 – 35.23, 2012.

[14] J. Lei and A. Rinaldo. Consistency of spectral clustering in sparse stochastic block models. 2013. arxiv:1312.2050.

[15] T. Qin and K. Rohe. Regularized spectral clustering under the degree-corrected stochastic blockmodel. 2013. arxiv:1309.4111.
[16] Antony Joseph and Bin Yu. Impact of regularization on spectral clustering. 2013. arxiv:1312.1733.

[17] Y. Chen, S. Sanghavi, and H. Xu. Clustering Sparse Graphs. NIPS, 2012.

[18] Y. Chen and J Xu. Statistical-Computational Tradeoffs in Planted Problems and Submatrix Localization with a Growing Number of Clusters and Submatrices. arXiv preprint arXiv:1402.1267, 2014.

[19] T. Cai and X. Li. Robust and Computationally Feasible Community Detection in the Presence of Arbitrary Outlier Nodes. arXiv preprint arXiv:1404.6000, 2014.

[20] S. C. Olhede and P. J. Wolfe. Network histograms and universality of blockmodel approximation. arXiv preprint arXiv:1312.5306, 2013.

[21] E. M. Airoldi, T. B. Costa, and S. H. Chan. Stochastic blockmodel approximation of a graphon: Theory and consistent estimation. In Advances in NIPS 26, pages 692–700. 2013.

[22] P. J. Wolfe and S. C. Olhede. Nonparametric graphon estimation. arXiv preprint arXiv:1309.5936, 2013.

[23] V. Q. Vu, J Cho, J Lei, and K Rohe. Fantope Projection and Selection: A near-optimal convex relaxation of sparse PCA. In NIPS, 2013.

[24] A. D’Aspremont, L. El Ghaoui, M. I. Jordan, and G. R.G. Lanckriet. A direct formulation for sparse PCA using semidefinite programming. NIPS, 2004.

[25] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. Foundations and Trends in Machine Learning, 3(1):1—-122, 2010.

[26] D. Lusseau and M. E. J. Newman. Identifying the role that animals play in their social networks. Proceedings of the Royal Society, Ser. B: Biological Sciences, 271:S477–S481, 2004.