A MAXIMUM A POSTERIORI RELAXATION FOR CLUSTERING THE LABELED STOCHASTIC BLOCK MODEL

Thomas Dittrich and Gerald Matz

Institute of Telecommunications, Technische Universität Wien
Email: firstname.lastname@tuwien.ac.at

***Abstract*—We consider the clustering problem for the labeled stochastic block model (LSBM) with non-uniform class priors. We introduce a novel relaxation of the maximum a posteriory (MAP) estimator for the cluster labels and develop an algorithm for the numerical solution of this relaxation, assuming that the number of clusters, the class priors, and the label distributions are known in advance. Semi-supervised operation is enabled by allowing each node to have a distinct prior. Numerical experiments confirm that our method outperforms state-of-the-art approaches in terms of clustering accuracy.**

I. Introduction

Clustering of data is a widely studied topic with numerous applications. Probably the best known approach is unsupervised spectral clustering on unsigned graphs [1]. Examples for semi-supervised clustering methods are [2]–[4]. Ideas for clustering on signed graphs [5] originate in [6], [7] and have recently been developed further both for unsupervised [8]–[15] and semi-supervised scenarios [16]–[18]. The common rationale of many clustering algorithms is to minimize some variation of the graph cut, i.e., to maximize separation (few positive edges between distinct communities) and homogeneity (few negative edges within communities).

The stochastic block model (SBM) is a practically useful and analytically tractable random graph model that is commonly studied in the context of graph clustering (see [19] for an overview of the current state of the art). For an SBM with two symmetric clusters, it was shown that minimizing the graph cut is equivalent to MAP estimation of the cluster labels [19, Page 34]. For the case of multiple clusters with possibly different sizes, [20] devised an algorithm that is asymptotically equivalent to MAP label estimation. In our experiments we select the model parameters based on the results on weak recovery from [21] (weak recovery means that the cluster overlap is bounded away from a random guess).

The LSBM generalizes the SBM by assigning a label to each edge [22]–[24]. These labels can be discrete, continuous, or even categorical. [22] provided a conjecture on the feasibility of reconstructing the cluster associations in the LSBM, special cases of which have been proven in [23], [25]. For the LSBM with two clusters, [23] showed that MAP coincides with cut minimization. Since the MAP estimator in general is NP-hard, several relaxations have been proposed based on semi-definite programming [23], [26]. The LSBM with multiple clusters has been studied in [24].

In this paper we propose and solve a relaxation of the MAP estimator for the LSBM with arbitrary number of clusters and non-uniform class priors. We assume that the number of clusters, the class priors, and the label distributions are known. In our formulation, distinct nodes may have different priors, which can be used e.g. to enable semi-supervised operation.

This work is organized as follows. Section II gives an introduction to the LSBM and formulates the MAP label estimator. In Section III we develop the proposed relaxation and its numerical solution. An experimental comparison of our approach to existing MAP based approaches and to some state-of-the-art clustering algorithms is provided in Section IV. Finally, in Section V we give a brief conclusion.

II. MAP Estimation for the LSBM

We adopt the notation from [22] for an LSBM with N nodes, edge set $\mathcal{E}$, and K clusters. The cluster labels $\sigma_i \in \{1,\dots,K\}$, $i \in \{1,\dots,N\}$, of all nodes are independent with prior distribution $\mathrm{P}\{\sigma_i\}$ (note that distinct nodes can have different priors). For each pair of nodes (i,j) with $\sigma_i = \sigma_j$, an (intra-cluster) edge is placed with probability p and assigned a label l_{ij} via a distribution $\mu(\cdot)$; if $\sigma_i \neq \sigma_j$, an (inter-cluster) edge (i,j) occurs with probability q and receives a label l_{ij} via the distribution $\nu(\cdot)$.

Since all edges and labels are mutually independent and the graph is symmetric without self loops, the distribution of the edge set $\mathcal{E}$ and the label set $\mathcal{L} = \{l_{ij}\}$ for given cluster labels $\boldsymbol{\sigma}$ can be written as

$$\mathrm{P}\{\mathcal{E},\mathcal{L}|\boldsymbol{\sigma}\} = \prod_{i=1}^{N-1}\prod_{j=i+1}^{N} \mathrm{P}\{(i,j),l_{ij}|\sigma_i,\sigma_j\}\,. \tag{1}$$

with

$$\mathrm{P}\{(i,j),\,l_{ij}|\sigma_i,\sigma_j\} = \begin{cases} p\,\mu(l_{ij}), & \sigma_i=\sigma_j,\ (i,j)\in\mathcal{E},\\ 1-p, & \sigma_i=\sigma_j,\ (i,j)\notin\mathcal{E},\\ q\,\nu(l_{ij}), & \sigma_i\neq\sigma_j,\ (i,j)\in\mathcal{E},\\ 1-q, & \sigma_i\neq\sigma_j,\ (i,j)\notin\mathcal{E}. \end{cases}$$

We now assume that we are given a graph with edge set $\mathcal{E}$ and label set $\mathcal{L}$ and our goal is to estimate the cluster labels $\boldsymbol{\sigma}$. The MAP estimator of the labels is given by

$$\hat{\boldsymbol{\sigma}} = \operatorname*{arg\,max}_{\boldsymbol{\sigma}\in\{1,\dots,K\}^N} \mathrm{P}\{\mathcal{E},\mathcal{L}|\boldsymbol{\sigma}\}\,\mathrm{P}\{\boldsymbol{\sigma}\}\,, \tag{2}$$

Taking logarithms and using (1) and the independence of the cluster labels, the MAP-estimator becomes

$$\hat{\boldsymbol{\sigma}}_{\mathrm{MAP}} = \operatorname*{arg\,max}_{\boldsymbol{\sigma}\in\{1,\dots,K\}^N} \sum_{i=1}^{N} \ln \mathrm{P}\{\sigma_i\} + \sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \ln \mathrm{P}\{(i,j),l_{ij}|\sigma_i,\sigma_j\}\,. \tag{3}$$

III. Proposed MAP Relaxation

A. Derivation

The MAP estimator is an integer programming problem and thus NP-complete. To obtain a tractable problem, we are proposing a relaxation with a continuous but non-convex objective function and a convex constraint set. To this end, we define a binary-valued matrix $\boldsymbol{X} \in \{0,1\}^{N\times K}$ of one-hot encodings for the cluster associations, i.e.,

$$X_{ik} = \begin{cases} 1, & \sigma_i = k, \\ 0, & \text{else.} \end{cases} \tag{4}$$

Furthermore, using the shorthand notation $p_{ij} = p\,\mu(l_{ij})$ and $q_{ij} = q\,\nu(l_{ij})$ we define the edge-likelihood matrix $\boldsymbol{P}_{ij} \in \mathbb{R}^{K\times K}$ as

$$\boldsymbol{P}_{ij} = \begin{cases} q_{ij}\mathbf{1}\mathbf{1}^T + (p_{ij}-q_{ij})\boldsymbol{I}, & (i,j)\in\mathcal{E}, \\ (1-q)\mathbf{1}\mathbf{1}^T + (q-p)\boldsymbol{I}, & (i,j)\notin\mathcal{E}. \end{cases} \tag{5}$$

With $\boldsymbol{x}_i^T$ denoting the ith row of $\boldsymbol{X}$ (i.e., the one-hot encoding of the cluster label σ_i), it then follows that

$$\boldsymbol{x}_i^T \boldsymbol{P}_{ij} \boldsymbol{x}_j = \mathrm{P}\{(i,j), l_{ij} | \sigma_i, \sigma_j\}\,. \tag{6}$$

Similarly, we write the the class membership distribution as

$$\mathrm{P}\{\sigma_i\} = \boldsymbol{p}_i^T \boldsymbol{x}_i,$$

where the kth entry of $\boldsymbol{p}_i$ equals the probability that node i belongs to cluster k. Exploiting the symmetry of $\boldsymbol{P}_{ij}$, the MAP estimator can be expressed in terms of the one-hot encodings $\boldsymbol{X}$ as (cf. (3))

$$\min_{\boldsymbol{X}\in\{0,1\}^{N\times K}} f(\boldsymbol{X}) \quad \text{s.t.} \quad \boldsymbol{X}\mathbf{1} = \mathbf{1}, \tag{7}$$

with

$$f(\boldsymbol{X}) = -\sum_{i=1}^{N} \ln\big(\boldsymbol{p}_i^T \boldsymbol{x}_i\big) - \frac{1}{2}\sum_{i\neq j} \ln\big(\boldsymbol{x}_i^T \boldsymbol{P}_{ij} \boldsymbol{x}_j\big)\,. \tag{8}$$

Note that the side-constraint in (7) ensures that $\boldsymbol{X}$ is indeed a one-hot encoding matrix. Since (7) is still a combinatorial optimization problem, we relax the condition $X_{ik} \in \{0,1\}$ to $X_{ik} \in [0,1]$ (i.e., $0 \leq X_{ik} \leq 1$). With the convex set

$$\mathcal{X} = \big\{\boldsymbol{X} \in [0,1]^{N\times K} \big| \boldsymbol{X}\mathbf{1} = \mathbf{1}\big\}\,,$$

the proposed MAP relaxation then is given by

$$\hat{\boldsymbol{X}} = \arg\min_{\boldsymbol{X}\in\mathcal{X}} f(\boldsymbol{X}). \tag{9}$$

The corresponding estimated cluster labels read

$$\hat{\sigma}_i = \operatorname*{arg\,max}_{k\in\{1,\ldots,K\}} \hat{X}_{ik}. \tag{10}$$

Algorithm 1 Projected Gradient Descent Estimator

Input: $\mathcal{E}$, $\mathcal{L}$, p, q, $\mu(\cdot)$, $\nu(\cdot)$, $\boldsymbol{p}$, $\boldsymbol{X}^{(0)}$, α_0, β, γ

1: compute $\boldsymbol{P}_{ij}$ for all (i,j) according to (5)
2: $t = 0$
3: **repeat**
4: $\quad t \leftarrow t+1$
5: $\quad \boldsymbol{J}^{(t)} \leftarrow \nabla f(\boldsymbol{X}^{(t-1)})$
6: $\quad \boldsymbol{D}^{(t)} \leftarrow \pi_{\mathcal{X}}\left(\boldsymbol{X}^{(t-1)} - \boldsymbol{J}^{(t)}\right) - \boldsymbol{X}^{(t-1)}$
7: $\quad \alpha \leftarrow \alpha_0$
8: $\quad$ **repeat**
9: $\quad\quad \alpha \leftarrow \beta\alpha$
10: $\quad\quad \boldsymbol{X}^{(t)} \leftarrow \boldsymbol{X}^{(t-1)} + \alpha \boldsymbol{D}^{(t)}$
11: $\quad$ **until** $f(\boldsymbol{X}^{(t-1)}) - f(\boldsymbol{X}^{(t)}) \geq \gamma\alpha \left\langle \boldsymbol{J}^{(t)}, \boldsymbol{D}^{(t)} \right\rangle$
12: **until** $\|\boldsymbol{D}^{(t)}\| \leq \epsilon$

Output: $\sigma_i = \arg\max_k X_{ik}^{(t)}, \quad i = 1,\ldots,N$

B. Solution

In general, some of the matrices $\boldsymbol{P}_{ij}$ are indefinite and hence the objective function $f(\boldsymbol{X})$ in (9) is not guaranteed to be convex. However, $f(\boldsymbol{X})$ is differentiable if the elements of $\boldsymbol{p}_i$ and $\boldsymbol{P}_{ij}$ are all strictly positive. To solve the (possibly non-convex) problem (9), we resort to a feasible direction method, choosing the directions based on projections of the gradient and the step-size according to the Armijo rule along the feasible direction with unit step-size [27, Section 2.3]. The overall projected gradient descent method is listed in Algorithm 1.

The elements of the gradient matrix $\boldsymbol{J} = \nabla f(\boldsymbol{X})$ in line 5 of Algorithm 1 are given by the partial derivatives

$$\frac{\partial}{\partial X_{ik}} f(\boldsymbol{X}) = -\frac{\boldsymbol{p}_i^T \boldsymbol{e}_k}{\boldsymbol{p}_i^T \boldsymbol{x}_i} - \sum_{j\neq i} \frac{\boldsymbol{x}_j^T \boldsymbol{P}_{ij} \boldsymbol{e}_k}{\boldsymbol{x}_j^T \boldsymbol{P}_{ij} \boldsymbol{x}_i} \tag{11}$$

Furthermore, line 6 involves the projection $\pi_{\mathcal{X}}(\cdot)$ on the constraint set $\mathcal{X}$, which imposes that each row of $\boldsymbol{X}$ belongs to the convex probability simplex. Hence, the projection onto $\mathcal{X}$ can be computed with the algorithm from [28].

As we are dealing with a possibly non-convex problem, the initialization $\boldsymbol{X}^{(0)}$ can be quite critical. In our experiments we use the labeling results obtained with clustering algorithms that feature good performance but are not computationally expensive.

C. Censored Block Model

The censored blockmodel (CBM) (see [29]) is an interesting special case of the LSBM with $p = q$ and $l_{ij} \in \{-1, 1\}$. This implies that for $(i,j) \notin \mathcal{E}$ all elements of $\boldsymbol{P}_{ij}$ are identical (cf. (5)) and hence absent edges in the graph are non-informative regarding the cluster affiliation. Note, however, that $p_{ij} \neq q_{ij}$ for $(i,j) \in \mathcal{E}$ since $\mu(\cdot) \neq \nu(\cdot)$.

These observations allow us to simplify the objective function in (8) since only terms with $(i,j) \in \mathcal{E}$ need to be retained. More specifically, since $\boldsymbol{P}_{ij} = c\mathbf{1}\mathbf{1}^T$ for $(i,j) \notin \mathcal{E}$, for these node pairs we have (here we use the one-hot constraint $\boldsymbol{x}_i^T\mathbf{1} = 1$)

$$\ln\left(\boldsymbol{x}_i^T \boldsymbol{P}_{ij} \boldsymbol{x}_j\right) = \ln\left(c\,\boldsymbol{x}_i^T \mathbf{1}\mathbf{1}^T \boldsymbol{x}_j\right) = \ln(c),$$

which is independent of $\boldsymbol{X}$ and hence can be dropped in the objective function, thus leading to

$$f(\boldsymbol{X}) = -\sum_{i=1}^{N} \ln\left(\boldsymbol{p}_i^T \boldsymbol{x}_i\right) - \frac{1}{2} \sum_{(i,j)\in\mathcal{E}} \ln\left(\boldsymbol{x}_i^T \boldsymbol{P}_{ij} \boldsymbol{x}_j\right), \quad (12)$$

which involves only $|\mathcal{E}| + N$ terms instead of N^2. A further simplification is obtained for uniform cluster label distribution, i.e., $\boldsymbol{p}_i = \mathbf{1}/K$ (this amounts to the assumption of equally-sized clusters); here, $\ln\left(\boldsymbol{p}_i^T \boldsymbol{x}_i\right) = \ln\left(\mathbf{1}^T \boldsymbol{x}_i / K\right) = -\ln(K)$, which is constant and can hence also be disregarded, leading to the objective function

$$f(\boldsymbol{X}) = -\frac{1}{2} \sum_{(i,j)\in\mathcal{E}} \ln\left(\boldsymbol{x}_i^T \boldsymbol{P}_{ij} \boldsymbol{x}_j\right). \quad (13)$$

In addition to substantially reduced computational complexity, this sparse simplification further enables an efficient distributed implementation of Algorithm 1 (e.g., the sum in the gradient computation in (11) only involves neighboring nodes).

D. Non-differentiable Cases

In some cases (specifically if $\boldsymbol{x}_i^T \boldsymbol{P}_{ij} \boldsymbol{x}_j = 0$ for some i and j), the objective in (7) is not differentiable on the whole feasible set $\mathcal{X}$. This can occur only if there is a label $l \in \mathcal{L}$ for which either $\mu(l) = 0$ or $\nu(l) = 0$. It can be shown that the points $\mathcal{X}^\dagger \subset \mathcal{X}$ where the objective is not differentiable lie on the relative boundary of $\mathcal{X}$. By choosing an initialization $\boldsymbol{X}^{(0)}$ in the relative interior of $\mathcal{X}$ and $\alpha_0 < 1$, we can still apply Algorithm 1 and it is guaranteed that $\boldsymbol{X}^{(t)} \notin \mathcal{X}^\dagger$ for any iteration t. Furthermore, the closer $\boldsymbol{X}^{(t)}$ gets to $\mathcal{X}^\dagger$, the stronger the gradient points away from $\mathcal{X}^\dagger$.

IV. Experiments

In this section we compare our MAP relaxation (MAPR) algorithm with several state-of-the art methods. We adopt the setting of [23], which means that we consider edge labels in $\{-1, 1\}$ and label distributions such that $\mu(1) = \nu(-1) = 0.5 + \epsilon$ with $\epsilon \in [0, 0.5]$. Here, $\epsilon = 0.5$ corresponds to the perfectly balanced case (i.e., only positive edges within clusters and only negative edges between distinct clusters). Since perfectly balanced graphs can be trivially clustered by dropping negative edges and identifying the resulting components, we restrict to $\epsilon \leq 0.45$ in our simulations.

We compare our MAPR method to the following four benchmark algorithms:

- the spectral method from [23] (denoted SMO), which attempts to maximize the overlap for the case of two clusters;
- the semi-definite programming (SDP) algorithm from [30] that was designed to achieve asymptotically exact recovery in the (standard) SBM;
- the spectral partitioning algorithm from [24] (denoted SP), which recovers the clusters asymptotically accurately in the general LSBM;
- the SPONGE algorithm [14], which provides bounds for weak recovery in the LSBM with $K = 2$.

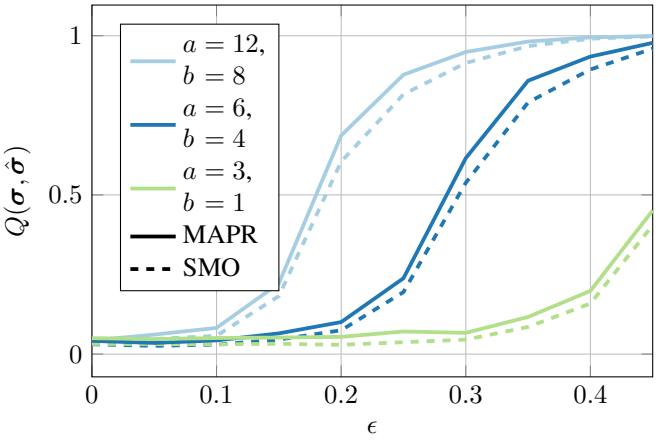

Figure 1. Average cluster overlap versus balancedness ϵ for MAPR and SMO.

The performance metric in all our experiments is the cluster overlap $Q(\boldsymbol{\sigma}, \hat{\boldsymbol{\sigma}})$ as defined in [22, eq. (9)] ($Q(\boldsymbol{\sigma}, \hat{\boldsymbol{\sigma}}) = 1$ amounts to perfect cluster recovery).

In all of our experiments we choose the parameters in Algorithm 1 as $\alpha_0 = 0.2$ and $\beta = \gamma = 1/2$. As initialization, we use $\boldsymbol{X}^{(0)} = \pi_{\mathcal{X}}(\widetilde{X})$ where

$$\widetilde{X}_{ik} = \begin{cases} 0.55, & \sigma_i^{(0)} = k, \\ 0.45, & \text{else.} \end{cases} \quad (14)$$

Here, $\boldsymbol{\sigma}^{(0)}$ denotes the labels obtained by some other algorithm (to be specified below for each experiment separately).

A. Two Clusters

For the comparison with SMO in the two class setting with equally sized clusters, we use an LSBM with $N = 1000$ nodes and the same configuration for $a = Np$ and $b = Nq$ as in [23], i.e., $(a, b) \in \{(12, 8), (6, 4), (3, 1)\}$. The results $\boldsymbol{\sigma}^{(0)}$ obtained with SMO are used to initialize Algorithm 1. Fig. 1 compares the cluster overlap (averaged over 100 independent realizations of the LSBM) obtained with MAPR and with SMO.

We can see that Algorithm 1 outperforms SMO by a small margin even though SMO was designed to maximize the cluster overlap. This small margin comes from the different types of relaxation. SMO uses a spectral (i.e., a quadratic) relaxation which favors small steps in many different positions of the label vector, which in turn can lead to errors in the final label quantization step. Conversely, with MAPR we obtain a matrix $\hat{\boldsymbol{X}}$ that—in the vast majority of cases—indeed belongs to $\{0, 1\}^{N\times K}$, so that the final quantization step (10) becomes obsolete.

B. Multiple Clusters: Unsigned SDP

Next we compare MAPR to SDP in order to see the impact of adding labels to an SBM. We generated an LSBM with labels in $\{-1, 1\}$, interpreted it as a signed graph and only considered the unsigned part to obtain an SBM. The resulting intra- and inter-cluster edge probabilities are given by $\tilde{p} = \mu(1)p$ and $\tilde{q} = \nu(1)q$, respectively. The probabilities p and q are chosen as $p = q = \frac{K(K+2)}{N}$; this implies that absent edges

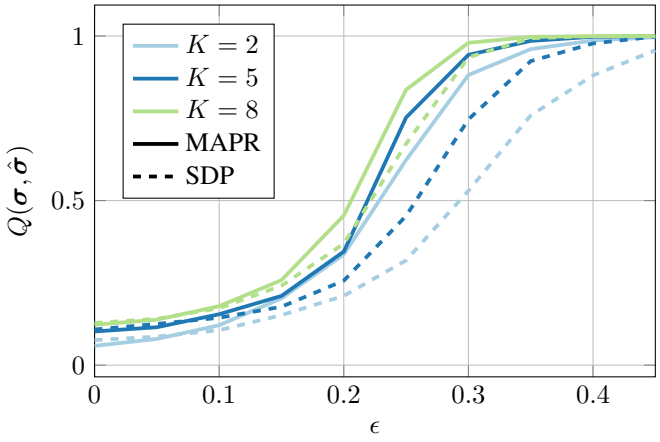

Figure 2. Average cluster overlap versus balancedness ϵ for MAPR and SDP.

are non-informative and that the left-hand side in the weak recovery threshold [19, Theorem 37]

$$\frac{N(\tilde{p}-\tilde{q})^2}{K(\tilde{p}+(K-1)\tilde{q})} > 1. \tag{15}$$

equals 1 for $\epsilon = 0.25$.

The implementation of SDP using standard solvers for semi-definite programming requires to store a full matrix of size $N(N+1)/2 \times N(N+1)/2$, which drastically limits the number of nodes in the LSBM. In this experiment we thus only consider graphs wih $N = 120$ nodes and equally sized clusters. We furthermore used SPONGE in the background to initialize Algorithm 1.

The clustering performance achieved with MAPR and SDP for $K = 2, 5, 8$ is depicted in Fig. 2 (the cluster overlap was averaged over 100 independent realizations of the LSBM). Interestingly, the performance in all cases indeed is seen to improve dramatically around the weak recovery threshold $\epsilon = 0.25$. Furthermore, MAPR clearly outperforms SDP for all K and ϵ since it better exploits the information underlying the LSBM. We also see that for increasing K the cluster labels obtained with MAPR become more accurate even at the threshold for weak recovery, even though the threshold requires strict inequality. However, the theoretical threshold holds only asymptotically for $N \to \infty$ and thus deviations for finite N are possible. Furthermore, with a fixed number of nodes the values for p and q increase quadratically with K, which directly influences the overall sparsity of the graph.

C. Multiple clusters: SP and SPONGE

In our last experiment, we compare MAPR to SP and SPONGE in the case of multiple clusters. We consider an LSBM with $N = 1000$ nodes and $K = 2, 5, 10$ equally sized clusters. Depending on the number of clusters, the edge probabilities were set again to $p = q = \frac{K(K+2)}{N}$.

For SPONGE we resort to the original Python implementation of [14] provided on GitHub (https://github.com/alan-turing-institute/SigNet). The SP method from [24] consists of two steps (we used the algorithm listed in the preprint version of [24] on arXiv). In the first step, SP finds estimates for the parameters of the LSBM and calculates a spectral decomposition of the graph. This results in a first estimate of the cluster labels, which are used as initial value for the second step. For the case that all graph parameters are known, [24] suggests to use an SVD in combination with the K-means clustering algorithm in the first step. We refer to this version as SP/SVD. Alternatively, we use the results from SPONGE as initial value for the second step (referred to as SP/SPONGE).

Fig. 3 compares the cluster overlap (averaged over 100 realizations) for all three algorithms. It can be seen that MAPR and SP both improve on SPONGE. This can again be attributed to the quantization problem discussed in Section IV.A. On top of that, MAPR and SP exploit the full parameter set underlying the LSBM whereas SPONGE only implicitly uses the information of equal cluster sizes. Furthermore, SP/SVD performs uniformly worst by a large margin, thereby emphasizing the importance of an appropriate initialization.

Interestingly, the (second) step of SP can be seen as a discrete version of MAPR. In every iteration, each node changes its cluster label to the most likely cluster association given the current state of all nodes. In [24] it was proved that asymptotically this converges to a solution where only a fixed number of nodes is assigned incorrect cluster labels. However, for the finite case the method often oscillates between two fixed points. This behaviour is reflected in the results in Fig. 3, since MAPR outperforms SP even with the same initialization.

In the case of two clusters, we can adapt the condition for asymptotically exact recovery from [24] and get

$$\frac{8}{\log(N)}\left(\sqrt{\frac{1}{2}+\epsilon}-\sqrt{\frac{1}{2}-\epsilon}\right)^2 > 1,$$

which amounts to a threshold of $\epsilon = 0.4953$.

V. Conclusion

We presented a new way to formulate the MAP estimator for the cluster structure of the LSBM using one-hot encoding. Based on this formulation we proposed a relaxation that we solve using a projected gradient approach. Our optimization algorithm tends to avoid the (often error-prone) label quantization problem afflicting several exting approaches (specifically spectral methods). Based on these advantages, our simulations revealed consistent performance gains for our MAP relaxation compared to spectral methods and to discrete optimization methods in the LSBM. Our experiments further confirmed that the additional information in the LSBM improves the clustering accuracy relative to the (unlabeled) SBM.

References

[1] U. von Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.

[2] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples.," *J. Mach. Learn. Res.*, vol. 7, no. 11, 2006.

[3] L. Xu, W. Li, and D. Schuurmans, "Fast normalized cut with linear constraints," in *IEEE CVPR 2019*, IEEE, 2009, pp. 2866–2873.

[4] X. Bresson, T. Laurent, D. Uminsky, and J. H. v. Brecht, "Multiclass total variation clustering," in *Proc 26th Int. Conf. Neur. Inf. Proc. Sys.*, 2013, pp. 1421–1429.

[5] T. Dittrich and G. Matz, "Signal processing on signed graphs: Fundamentals and potentials," *IEEE Sig. Proc. Mag.*, vol. 37, no. 6, pp. 86–98, 2020.

[6] F. Harary, "On the notion of balance of a signed graph," *Mich. Math. J.*, vol. 2, no. 2, pp. 143–146, 1953.

[7] D. Cartwright and F. Harary, "Structural balance: A generalization of heider's theory," *Psychol. Rev.*, vol. 63, no. 5, pp. 277–293, 1956.

[8] A. B. Goldberg, X. Zhu, and S. Wright, "Dissimilarity in graph-based semi-supervised classification," in *Proc. Int. Conf. on Art. Int. Stat.*, San Juan (Puerto Rico), Mar. 2007, pp. 155–162.

[9] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. De Luca, and S. Albayrak, "Spectral analysis of signed graphs for clustering, prediction and visualization," in *Proc. SIAM Int. Conf. Data Mining*, Columbus (OH), May 2010, pp. 559–570.

[10] K.-Y. Chiang, J. J. Whang, and I. S. Dhillon, "Scalable clustering of signed networks using balance normalized cut," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manag.*, ACM, 2012, pp. 615–624.

[11] P. Mercado, F. Tudisco, and M. Hein, "Clustering signed networks with the geometric mean of laplacians," in *Adv. Neur. Inf. Proc. Sys.*, 2016, pp. 4421–4429.

[12] ——, "Spectral clustering of signed graphs via matrix power means," *arXiv preprint arXiv:1905.06230*, 2019.

[13] P. Mercado, J. Bosch, and M. Stoll, "Node classification for signed social networks using diffuse interface methods," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, 2019, pp. 524–540.

[14] M. Cucuringu, P. Davies, A. Glielmo, and H. Tyagi, "Sponge: A generalized eigenproblem for clustering signed networks," *arXiv preprint arXiv:1904.08575*, 2019.

[15] T. Dittrich and G. Matz, "Unsupervised clustering on signed graphs with unknown number of clusters,"

[16] P. Berger, T. Dittrich, and G. Matz, "Semi-Supervised Clustering Based on Signed Total Variation," in *GlobalSIP 2018*, IEEE, 2018, pp. 793–797.

[17] P. Berger, T. Dittrich, G. Hannak, and G. Matz, "Semi-Supervised Multiclass Clustering Based on Signed Total Variation," in *ICASSP 2019*, IEEE, Brighton, UK, May 2019, pp. 4953–4957.

[18] M. Cucuringu, A. Pizzoferrato, and Y. van Gennip, "An mbo scheme for clustering and semi-supervised clustering of signed networks," *arXiv preprint arXiv:1901.03091*, 2019.

[19] E. Abbe, "Community detection and stochastic block models: Recent developments," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6446–6531, 2017.

[20] T. P. Peixoto, "Bayesian stochastic blockmodeling," *Adv. Net. Clust. Blockm.*, pp. 289–332, 2019.

[21] E. Abbe and C. Sandon, "Detection in the stochastic block model with multiple clusters: Proof of the achievability conjectures, acyclic bp, and the information-computation gap," *arXiv preprint arXiv:1512.09080*, 2015.

[22] S. Heimlicher, M. Lelarge, and L. Massoulié, "Community detection in the labelled stochastic block model," *arXiv preprint arXiv:1209.2910*, 2012.

[23] M. Lelarge, L. Massoulié, and J. Xu, "Reconstruction in the labelled stochastic block model," *IEEE Trans. Netw. Sci. Eng.*, vol. 2, no. 4, pp. 152–163, 2015.

[24] S.-Y. Yun and A. Proutiere, "Optimal cluster recovery in the labeled stochastic block model," in *Conf. Neur. Inf. Proc. Sys.*, Neural information processing systems foundation, 2016, pp. 973–981.

[25] A. Saade, M. Lelarge, F. Krzakala, and L. Zdeborová, "Spectral detection in the censored block model," in *IEEE ISIT 2015*, IEEE, 2015, pp. 1184–1188.

[26] E. Abbe, A. S. Bandeira, and G. Hall, "Exact recovery in the stochastic block model," *IEEE Trans. Inf. Theory*, vol. 62, no. 1, pp. 471–487, 2015.

[27] D. P. Bertsekas, "Nonlinear programming," *J. Oper. Res. Soc.*, vol. 48, no. 3, pp. 334–334, 1997.

[28] W. Wang and M. A. Carreira-Perpinán, "Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application," *arXiv preprint arXiv:1309.1541*, 2013.

[29] E. Abbe, A. S. Bandeira, A. Bracher, and A. Singer, "Decoding binary node labels from censored edge measurements: Phase transition and efficient recovery," *IEEE Trans. Netw. Sci. Eng.*, vol. 1, no. 1, pp. 10–22, 2014.

[30] B. Hajek, Y. Wu, and J. Xu, "Achieving exact cluster recovery threshold via semidefinite programming: Extensions," *IEEE Trans. Inf. Theory*, vol. 62, no. 10, pp. 5918–5937, 2016.

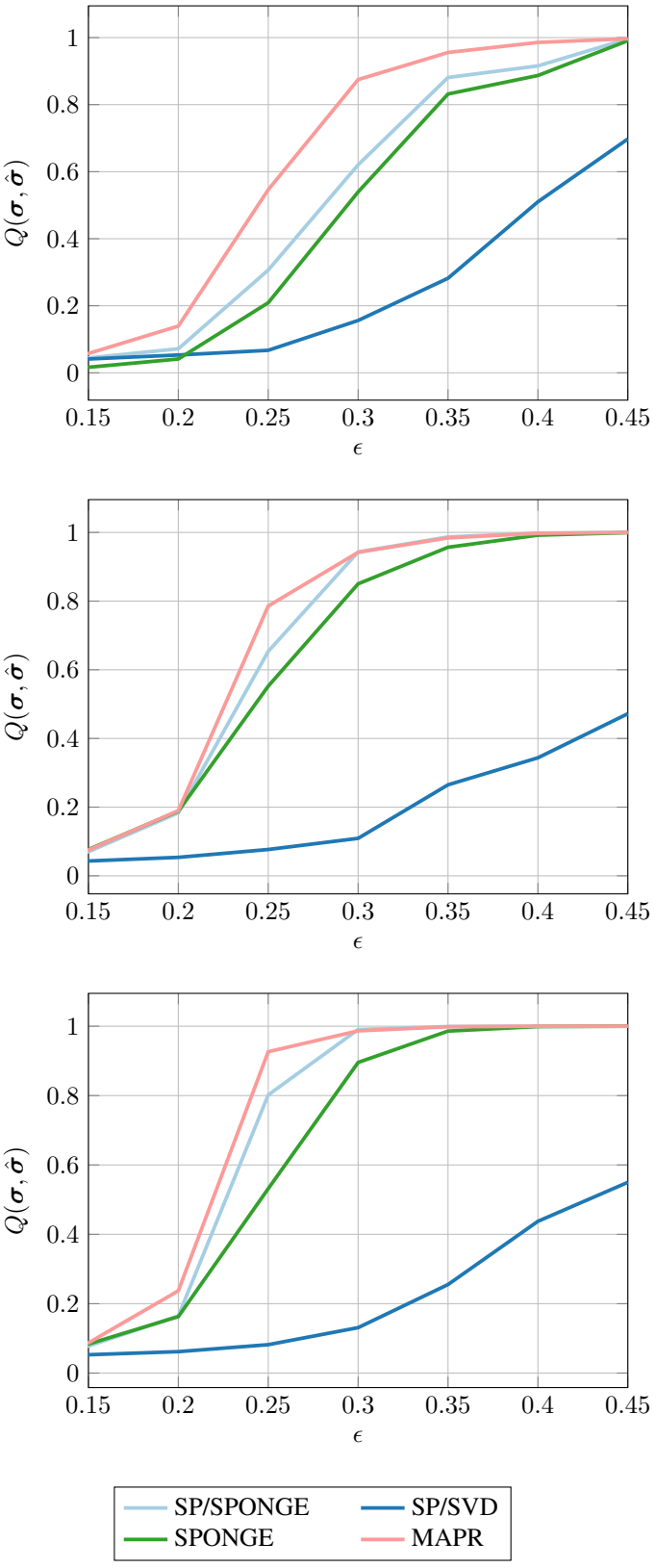

Figure 3. Average cluster overlap versus balancedness ϵ for MAPR, SP (with two initializations), and SPONGE (top: $K = 2$, middle: $K = 5$, bottom: $K = 10$).