NEURAL-PRIOR STOCHASTIC BLOCK MODEL

O. Duranthon & L. Zdeborová
SPOC laboratory, EPFL
Lausanne, Switzerland

ABSTRACT

The stochastic block model (SBM) is widely studied as a benchmark for graph clustering aka community detection. In practice, graph data often come with node attributes that bear additional information about the communities. Previous works modelled such data by considering that the node attributes are generated from the node community memberships. In this work, motivated by recent surge of works in signal processing using deep neural networks as priors, we propose to model the communities as being determined by the node attributes rather than the opposite. We define the corresponding model; that we call the neural-prior SBM. We propose an algorithm, stemming from statistical physics, based on a combination of belief propagation and approximate message passing. We argue it achieves Bayes-optimal performance for the considered setting. The proposed model and algorithm can hence be used as a benchmark for both theory and algorithms. To illustrate this, we compare the optimal performances to the performance of a simple graph convolution network.

1 INTRODUCTION

The stochastic block model (SBM) is widely studied as a benchmark for graph clustering aka community detection, see e.g. reviews Fortunato (2010); Abbe (2017). There are several possible sources of information on communities one can use: the structure of the graph, and the features or attributes of the nodes. Past work developed algorithms and models accounting for such node information. Among the well known is the CESNA model of Yang et al. (2013) where the attributes are generated via logistic regression on the community membership. Another model that recently became popular in the context of benchmarking graph neural networks (e.g. Chien et al. (2021); Fountoulakis et al. (2022); Tsitsulin et al. (2021)) is the the contextual-SBM Binkiewicz et al. (2017); Deshpande et al. (2018), where communities determine centroids for a Gaussian mixture model generating the features. In both these examples the node attributes are generated via conditioning on the community label of the node.

In signal processing, another separate line of work, that witnesses a surge of interest, is modelling signals as the output of a deep generative neural network; for recent reviews see e.g. Ongie et al. (2020); Shlezinger et al. (2020). Deep generative neural networks can be trained on data, and due to their expressivity are able to capture any structural properties of the signal. In community detection the signal are the community memberships; following the line of work on deep generative priors it is hence of interest to propose a model where the node attributes are an input of a generative neural network and the node community memberships are the output thereof.

One of the attires of the stochastic block model is that it is amenable to exact statistical analysis of what is the best achievable performance from an information-theoretic and from an algorithmic point of view. This has led to a line of work, originating in statistical physics, where statistical and computational thresholds are analyzed; see e.g. Decelle et al. (2011a); Abbe et al. (2015); Abbe (2017). It is valuable to have a solvable case for which we know what is statistically and algorithmically possible; because in the context of modern machine learning it is rarely known if or how much the observed performance can be further improved. Asymptotically exact analysis of the detectability threshold was also performed for the contextual stochastic block model Deshpande et al. (2018); Lu & Sen (2020). The main topic of the present paper is the statistical physics analysis of optimal algorithmic performance for the proposed neural-prior stochastic block model.

The model we propose can be used for benchmarking graph neural networks (GNNs). Since our model is analysable, we can compare the performance of the evaluated GNN to the optimal algorithmic performance.

As far as we found, the model we propose has one previous use in Cho et al. (2022). In that work it is used as a building block for a large neural network; it was not analyzed per se.

2 THE NEURAL-PRIOR STOCHASTIC BLOCK MODEL

We consider a set V of $|V| = N$ nodes, a graph $G(V, E)$ on those nodes. Nodes have features/attributes $F_\mu \in \mathbb{R}^M$ of dimension M, $\mu = 1 \ldots, N$. The features and the graphs are observed. We aim to divide the set of nodes into q communities with labels $s_\mu \in \{1, \ldots, q\}$ in such a way that (a) the graph structure correlates with the labels, e.g. nodes being in the same community are more likely to be connected, and (b) the node attributes F_μ are correlated with the labels.

Standard SBM: In the stochastic block model the edges $A_{\mu\nu}$ of the graph G are generated conditioned on the group-memberships s_μ as follows

$$P(A_{\mu\nu} = 1 | s_\mu, s_\nu) = \begin{cases} c_i/N & \text{if} \quad s_\mu = s_\nu, \\ c_o/N & \text{if} \quad s_\mu \neq s_\nu, \end{cases} \tag{1}$$

and $A_{\mu\nu} = 0$ otherwise. Here c_i and c_o are the affinity coefficients common to the SBM. We define the affinity matrix whose elements are $c_{s,t} = c_i \delta_{s=t} + c_o \delta_{s \neq t}$. In the usual stochastic block model the ground truth group memberships s_μ are generated at random from a prior that only accounts for the sizes of the q groups.

Neural-prior SBM: In neural-prior SBM, that we define here, the group memberships s_μ can be a generic function on the attributes F_μ. Such a function can be represented by a deep neural networks and learned from ground-truth data. The training data would be pairs $\{F_\mu, s_\mu\}$ where attributes act as the neural network inputs and the group memberships as output labels. For instance for a L-layer fully connected neural network this reads $s_\mu = \varphi^{(L)}\big(W^{(L)} \ldots \varphi^{(2)}\big(W^{(2)}\varphi^{(1)}(W^{(1)}F)\big) \ldots\big)$; for the last activation function $\varphi^{(L)}$ chosen as in multi-class classification tasks.

The aim of this paper is to provide a benchmark model where the optimal performance can be analyzed asymptotically exactly. For this we need to (a) define the corresponding asymptotic limit, (b) consider a simple neural network prior that is amenable to asymptotic analysis. We will also limit ourselves to consider community detection with two groups only, $q = 2$ (this is not a strong limitation, but is considered in the follow up for simplicity). With this in mind, in the rest of the paper we will consider the following model generating the group memberships s_μ.

GLM–SBM: In order to make analysis amenable we will consider the features F to be random and drawn independently as $F_{\mu l} \sim \mathcal{N}(0, 1/M)$. We then consider M latent variables $w_l \sim P_w$, $l = 1, \ldots, M$ and generate the community memberships as

$$s_\mu = \operatorname{sign}\big(\sum_l^M F_{\mu l} w_l\big) \tag{2}$$

This corresponds to a single-layer neural network with a sign activation function. Such a neural network is also often referred to as the generalized linear model (GLM). We will hence call this variant of the neural-prior SBM the GLM–SBM.

Concerning the asymptotic limit, we work in the challenging sparse case of SBM. We parameterize the SBM by the standard parameterization $c_i = c + \sqrt{c}\lambda$ and $c_o = c - \sqrt{c}\lambda$. We then consider $N \to \infty$ with $c = (c_i + c_o)/2 = \mathcal{O}(1)$ is the average degree, and $\lambda = \mathcal{O}(1)$ is the signal-to-noise ratio. We further work in the high-dimensional limit of the GLM where $\frac{N}{M} = \alpha = \mathcal{O}(1)$, with α being the aspect ratio that will play a role of another signal-to-noise ratio. This is because the higher α the more correlation there is between the group memberships and the easier the community detection should be.

The GLM–SBM differs from the standard SBM because communities are not independent, conditionally on the features. For instance, in the extreme case $M = 1$, all memberships are known, up to a global flip given by w_1; that is to say they are all very strongly correlated. The GLM–SBM tends toward a standard SBM when $\alpha \to 0$. Indeed, for large M, $\sum_l^M F_{\mu l} w_l$ tend to independent Gaussian variables.

3 THE AMP–BP ALGORITHM

The algorithm we propose is based on belief propagation (BP) and approximate message-passing (AMP). BP was used to solve SBM in Decelle et al. (2011b) and conjectured asymptotically optimal among efficient algorithms in doing so; AMP to solve GLM, see e.g. Krzakala et al. (2012), and again conjecture asymptotically optimal among efficient algorithms in doing so. We glue these two algorithms together along the lines of Manoel et al. (2017); Aubin et al. (2019) to solve the GLM–SBM. The derivation of the resulting AMP–BP algorithm is given in the appendix. Using statistical physics arguments analogous to those in Decelle et al. (2011b); Krzakala et al. (2012) we conjecture that this algorithm provides asymptotically optimal performance in the considered cases. The AMP–BP algorithm can be used in the unsupervised case as well as in the semi-supervised case where group memberships are known for a fraction of nodes. In the next section, we compare it to a graph-convolution baseline and a GNN baseline.

In the following the χs and ψs are probability distributions; the Zs are normalization factors. We introduce the denoising function:

$$g_o(\omega, \chi, V) = \frac{\int \mathrm{d}z \sum_s \chi_s P_0(s|z)(z-\omega)e^{-(z-\omega)^2/2V}}{V \int \mathrm{d}z \sum_s \chi_s P_0(s|z) e^{-(z-\omega)^2/2V}} \tag{3}$$

We define the input functions as $f_a(\Lambda, \Gamma) = \frac{\int \mathrm{d}w\, P_w(w) w e^{-\Lambda w^2/2+\Gamma w}}{\int \mathrm{d}w P_w(w) e^{-\Lambda w^2/2+\Gamma w}}$ and $f_v(\Lambda, \Gamma) = \partial_\Gamma f_a(\Lambda, \Gamma)$ We introduce also the output distribution $P_o(s|z) = \delta_{s=\mathrm{sign}(z)}$ and the prior distribution $P_{s,\mu}$, which is used to inject additional information about the membership of node μ: $P_{s,\mu}(t) = 1/2$ in the unsupervised case, $P_{s,\mu}(t) = \delta_{t=s_\mu}$ if s_μ is given in the semi-supervised case. The algorithm is:

AMP–BP

input features $F_{\mu l}$, graph G, affinity matrix $c_{s,t}$, prior information $P_{s,\mu}$.

Initialize $a_l^{(0)} = \epsilon_l$, $v_l^{(0)} = 1$, $g_{o,\mu}^{(0)} = 0$, $\chi_{s_\mu}^{\mu\to\nu,(0)} = \frac{1}{2} + s_\mu \epsilon^{\mu\to\nu}$, $\chi_{s_\mu}^{\mu\to\mu,(0)} = \frac{1}{2}$, $\chi_{s_\mu}^{\mu,(0)} = \frac{1}{2}$, $t = 0$; where ϵs are zero-mean small random variables.

repeat

AMP update of ω_μ, V_μ

$$V^{(t+1)} \leftarrow \frac{1}{M} \sum_l v_l^{(t)}$$

$$\omega_\mu^{(t+1)} \leftarrow \sum_l F_{\mu l} a_l^{(t)} - V^{(t+1)} g_{o,\mu}^{(t)}$$

AMP update of $\psi^{\mu\to\mu}, g_{o,\mu}, \Lambda, \Gamma_l$

$$\psi_{s_\mu}^{\mu\to\mu,(t+1)} \leftarrow \int \frac{\mathrm{d}z P_0(s_\mu|z)}{\sqrt{2\pi V_\mu^{(t+1)}}} e^{-\frac{(z-\omega_\mu^{(t+1)})^2}{2V_\mu^{(t+1)}}}$$

$$g_{o,\mu}^{(t+1)} \leftarrow g_o(\omega_\mu^{(t+1)}, \chi^{\mu\to\mu,(t)}, V^{(t+1)})$$

$$\Lambda^{(t+1)} \leftarrow \frac{1}{M} \sum_\mu g_{o,\mu}^{2,(t+1)}$$

$$\Gamma_l^{(t+1)} \leftarrow \Lambda^{(t+1)} a_l^{(t)} + \sum_\mu F_{\mu l} g_{o,\mu}^{(t+1)}$$

AMP update of the estimated marginals a_l, v_l

$$a_l^{(t+1)} \leftarrow f_a(\Lambda^{(t+1)}, \Gamma_l^{(t+1)})$$

$$v_l^{(t+1)} \leftarrow f_v(\Lambda^{(t+1)}, \Gamma_l^{(t+1)})$$

BP update of the field h

$$h_s^{(t+1)} \leftarrow \frac{1}{N} \sum_\mu \sum_{s_\mu} c_{s,s_\mu} \chi_{s_\mu}^{\mu,(t)}$$

BP update of the messages $\chi^{\mu\to\nu}$ for $(\mu\nu) \in G$ and of the marginals χ^μ

$$\chi_{s_\mu}^{\mu\to\nu,(t+1)} \leftarrow \frac{P_{s,\mu}(s_\mu)}{Z^{\mu\to\nu}} e^{-h_{s_\mu}^{(t+1)}} \psi_{s_\mu}^{\mu\to\mu,(t+1)} \prod_{\eta\in\partial\mu\setminus\nu} \sum_{s_\eta} c_{s_\eta,s_\mu} \chi_{s_\eta}^{\eta\to\mu,(t)}$$

$$\chi_{s_\mu}^{\mu,(t+1)} \leftarrow \frac{P_{s,\mu}(s_\mu)}{Z^{\mu}} e^{-h_{s_\mu}^{(t+1)}} \psi_{s_\mu}^{\mu\to\mu,(t+1)} \prod_{\eta\in\partial\mu} \sum_{s_\eta} c_{s_\eta,s_\mu} \chi_{s_\eta}^{\eta\to\mu,(t)}$$

BP update of the SBM-to-GLM messages $\chi^{\mu\to\mu}$

$$\chi_{s_\mu}^{\mu\to\mu,(t+1)} \leftarrow \frac{P_{s,\mu}(s_\mu)}{Z^{\mu\to\mu}} e^{-h_{s_\mu}^{(t+1)}} \prod_{\eta\in\partial\mu} \sum_{s_\eta} c_{s_\eta,s_\mu} \chi_{s_\eta}^{\eta\to\mu,(t)}$$

$t \leftarrow t+1$

until convergence of a_l, v_l, χ^μ

output estimated mean a_l and variance v_l of w_l and marginal distribution χ^μ of s_μ.

We draw attention on the output function g_o that covers the difference between AMP for GLM–SBM and AMP for GLM alone. In AMP for GLM alone g_o depends on the observed labels while here we use their estimated marginals. On the other side, the difference between BP for GLM–SBM and BP for SBM alone are the messages $\psi^{\mu\to\mu}$ in BP update. $\psi^{\mu\to\mu}$ can be interpreted as the conditional probability of s_μ given w without SBM. We provide an implementation of AMP–BP for GLM-SBM in the supplementary material.

4 Performance comparison

We measure the performance of the recovery thanks to the overlaps between the ground truths s_μ or w_l and their estimators. Their estimators are $\hat{s}_\mu = \operatorname{sign}(2\chi_+^\mu - 1)$ and $\hat{w}_l = a_l$. The overlaps are $q_S = |\hat{s}.s|/N$ and $q_W = |\hat{w}.w|/||\hat{w}||_2||w||_2$.

The performance is depicted on figs. 2 and 3 in appendix D. The larger λ or α the better the recovery. The recovery is eased when community memberships are explained by few features. In the unsupervised case with $q = 2$ the problem admits a sharp transition from a non-informative fixed-point ($q_S = q_W = 0$) to an informative fixed-point ($q_S > 0, q_W > 0$). The transition is located at a particular λ_c, called critical lambda. This transition is well known for standard SBM, where $\lambda_c = 1$. The transition is of 2nd order; this means that the overlaps vary continuously with respect to λ. In the semi-supervised case the sharp transition disappears.

We compute λ_c by linearizing the algorithm around its non-informative fixed-point and studying its stability. At a given λ, if it is not stable then the algorithm will move away to the informative fixed-point. In the appendix B we derive that $\lambda_c = \left(1 + 4\alpha/\pi^2\right)^{-1/2}$.

An unsupervised baseline The algorithm to which we compare is inspired by graph convolution networks; it performs unsupervised clustering. We compare its performances to the optimal ones given by AMP–BP. The algorithm is detailed in appendix C. Its performances are shown on fig. 1.

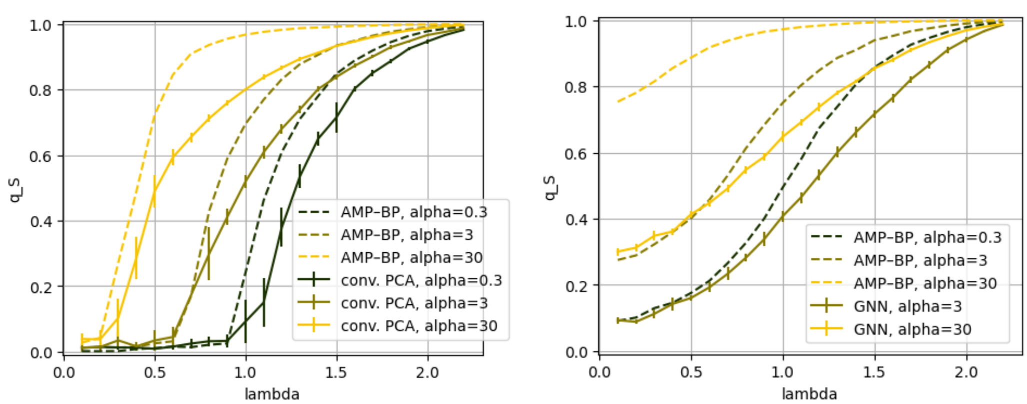


Figure 1: Overlap q_S of the baseline algorithms, vs λ. We compare to the overlap obtained by AMP–BP. *Left:* unsupervised; for the parameters of the graph convolution we choose $a = 0.1$ and $n = 4$. *Right:* semi-supervised; for the hyperparameters of the GNN we choose $n = 2$, learning rate 10^{-3} and L2 penalty 10^{-2}. The train set is $1/10^{\text{th}}$ of the nodes. $N = 10^4$, $c = 5$. We ran ten experiments per datapoint.

A semi-supervised baseline The algorithm to which we compare is a simple GNN, trained in a semi-supervised way for node classification. The GNN is made of a perceptron and a readout layer for the binary classification. It reads: $F_\mu^{(t+1)} = F_\mu^{(t)} + A.\sum_{\nu\in\partial\mu} F_\nu^{(t)}$, $\hat{s}_\mu = w^T.F_\mu^{(n)}$, where A is $M \times M$ learnable, $w \in \mathbb{R}^M$ learnable and n is the number of steps. We train it given the labels of a subset of nodes. We use gradient descent with logistic loss and L2 regularization. We do not fine-tune the hyperparameters. Its performances are shown on fig. 1.

Conclusion on the comparison Fig. 1 illustrates that both in the unsupervised and the semi-supervised settings the used baseline methods have a considerable gap to the performance of the AMP–BP algorithm. The GLM–SBM setting is hence suitable to develop GNN algorithms that are able to provide higher accuracy.

REFERENCES

Emmanuel Abbe. Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531, 2017.

Emmanuel Abbe, Afonso S Bandeira, and Georgina Hall. Exact recovery in the stochastic block model. *IEEE Transactions on information theory*, 62(1):471–487, 2015.

Benjamin Aubin, Bruno Loureiro, Antoine Maillard, Florent Krzakala, and Lenka Zdeborová. The spiked matrix model with generative priors. In *Advances in Neural Information Processing Systems*, 2019. arxiv:1905.12385.

Norbert Binkiewicz, Joshua T. Vogelstein, and Karl Rohe. Covariate-assisted spectral clustering. *Biometrika*, 104(2):361–377, 2017. arxiv:1411.2158.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. In *International Conference on Learning Representations*, 2021. arxiv:2006.07988.

Sungjun Cho, Seonwoo Min, Jinwoo Kim, Moontae Lee, Honglak Lee, and Seunghoon Hong. Transformers meet stochastic block models: Attention with data-adaptive sparsity and cost. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=w_jvWzNXd6n`.

Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106, 2011a.

Aurélien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Phys. Rev. E*, 84, 2011b. arxiv:1109.3041.

Yash Deshpande, Subhabrata Sen, Andrea Montanari, and Elchanan Mossel. Contextual stochastic block models. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31, 2018. arxiv:1807.09596.

Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3-5):75–174, 2010.

Kimon Fountoulakis, Dake He, Silvio Lattanzi, Bryan Perozzi, Anton Tsitsulin, and Shenghao Yang. On classification thresholds for graph attention with edge features. arxiv:2210.10014, 2022.

Florent Krzakala, Marc Mézard, Francois Sausset, Yifan Sun, and Lenka Zdeborová. Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and threshold achieving matrices. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(08), aug 2012. arxiv:1206.3953.

Chen Lu and Subhabrata Sen. Contextual stochastic block model: Sharp thresholds and contiguity. *arXiv preprint arXiv:2011.09841*, 2020.

Andre Manoel, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. Multi-layer generalized linear estimation. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 2098–2102. IEEE, 2017.

Gregory Ongie, Ajil Jalal, Christopher A Metzler, Richard G Baraniuk, Alexandros G Dimakis, and Rebecca Willett. Deep learning techniques for inverse problems in imaging. *IEEE Journal on Selected Areas in Information Theory*, 1(1):39–56, 2020.

Nir Shlezinger, Jay Whang, Yonina C Eldar, and Alexandros G Dimakis. Model-based deep learning. *arXiv preprint arXiv:2012.08405*, 2020.

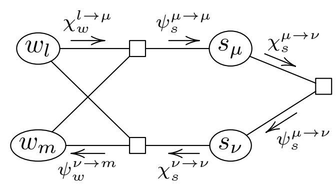
Anton Tsitsulin, Benedek Rozemberczki, John Palowitch, and Bryan Perozzi. Synthetic graph generation to benchmark graph learning. In *Workshop on Graph Learning Benchmarks*, 2021. arxiv:2204.01378.

Jaewon Yang, Julian McAuley, and Jure Leskovec. Community detection in networks with node attributes. In *2013 IEEE 13th international conference on data mining*, pp. 1151–1156. IEEE, 2013.

Lenka Zdeborová and Florent Krzakala. Statistical physics of inference: Thresholds and algorithms. *Advances in Physics*, 65(5):453–552, 2016.

A Derivation of the algorithm

We write belief propagation for this problem. We start from the factor graph. It contains six different messages:



These messages satisfy these equations:

$$\chi_{w_l}^{l\to\mu} \propto P_w(w_l) \prod_{\nu\neq\mu} \psi_{w_l}^{\nu\to l} \tag{4}$$

$$\psi_{w_l}^{\nu\to l} \propto \sum_{s_\nu} \chi_{s_\nu}^{\nu\to\nu} \int \prod_{m\neq l} \left(\mathrm{d}w_m \chi_{w_m}^{m\to\nu}\right) P_0(s_\nu|F_\nu.w) \tag{5}$$

$$\psi_{s_\nu}^{\nu\to\nu} \propto \int \prod_{m} \left(\mathrm{d}w_m \chi_{w_m}^{m\to\nu}\right) P_0(s_\nu|F_\nu.w) \tag{6}$$

$$\chi_{s_\mu}^{\mu\to\mu} \propto P_{s,\mu}(s_\mu) \prod_{\nu\neq\mu} \psi_{s_\mu}^{\nu\to\mu} \tag{7}$$

$$\chi_{s_\mu}^{\mu\to\nu} \propto P_{s,\mu}(s_\mu) \psi_{s_\mu}^{\mu\to\mu} \prod_{\eta\neq\mu,\nu} \psi_{s_\mu}^{\eta\to\mu} \tag{8}$$

$$\psi_{s_\nu}^{\mu\to\nu} \propto \sum_{s_\mu} \chi_{s_\mu}^{\mu\to\nu} P(A_{\mu\nu}|s_\mu, s_\nu) \tag{9}$$

We can plug ψ messages into the χ to obtain, for the GLM part:

$$\chi_{w_l}^{l\to\mu} = \frac{P_w(w_l)}{Z^{l\to\mu}} \prod_{\nu\neq\mu} \left[\sum_{s_\nu} \chi_{s_\nu}^{\nu\to\nu} \int \prod_{m\neq l} \left(\mathrm{d}w_m \chi_{w_m}^{m\to\nu}\right) P_0(s_\nu|F_\nu.w) \right] \tag{10}$$

the marginals

$$\chi_{w_l}^{l} = \frac{P_w(w_l)}{Z^{l}} \prod_{\nu} \left[\sum_{s_\nu} \chi_{s_\nu}^{\nu\to\nu} \int \prod_{m\neq l} \left(\mathrm{d}w_m \chi_{w_m}^{m\to\nu}\right) P_0(s_\nu|F_\nu.w) \right] \tag{11}$$

and for the SBM part:

$$\chi_{s_\mu}^{\mu\to\nu} \propto P_{s,\mu}(s_\mu) \psi_{s_\mu}^{\mu\to\mu} \prod_{\eta\neq\mu,\nu} \sum_{s_\eta} \chi_{s_\eta}^{\eta\to\mu} P(A_{\mu\eta}|s_\mu, s_\eta) \tag{12}$$

the marginals

$$\chi_{s_\mu}^{\mu} \propto P_{s,\mu}(s_\mu) \psi_{s_\mu}^{\mu\to\mu} \prod_{\eta\neq\mu} \sum_{s_\eta} \chi_{s_\eta}^{\eta\to\mu} P(A_{\mu\eta}|s_\mu, s_\eta) \tag{13}$$

and

$$\chi_{s_\mu}^{\mu\to\mu} \propto P_{s,\mu}(s_\mu) \prod_{\nu\neq\mu} \sum_{s_\nu} \chi_{s_\nu}^{\nu\to\mu} P(A_{\mu\nu}|s_\mu, s_\nu) \tag{14}$$

A.1 SBM

We can apply the standard simplifications for sparse SBM Decelle et al. (2011b), Zdeborová & Krzakala (2016). We consider only messages on G. This gives

$$\chi^{\mu\to\mu}_{s_\mu} = \frac{1}{Z^{\mu\to\mu}} P_{s,\mu}(s_\mu) e^{-h_{s_\mu}} \prod_{\eta\in\partial\mu} \sum_{s_\eta} c_{s_\eta,s_\mu} \chi^{\eta\to\mu}_{s_\eta} \tag{15}$$

$$\chi^{\mu\to\nu}_{s_\mu} = \frac{1}{Z^{\mu\to\nu}} P_{s,\mu}(s_\mu) \psi^{\mu\to\mu}_{s_\mu} e^{-h_{s_\mu}} \prod_{\eta\in\partial\mu\setminus\nu} \sum_{s_\eta} c_{s_\eta,s_\mu} \chi^{\eta\to\mu}_{s_\eta} \tag{16}$$

and the marginals

$$\chi^{\mu}_{s_\mu} = \frac{1}{Z^{\mu}} P_{s,\mu}(s_\mu) \psi^{\mu\to\mu}_{s_\mu} e^{-h_{s_\mu}} \prod_{\eta\in\partial\mu} \sum_{s_\eta} c_{s_\eta,s_\mu} \chi^{\eta\to\mu}_{s_\eta} \tag{17}$$

where $h_s = \frac{1}{N}\sum_\mu \sum_{s_\mu} c_{s,s_\mu}\chi^\mu_{s_\mu}$.

A.2 GLM

For the GLM, we follow closely Zdeborová & Krzakala (2016).

A.2.1 r-BP

We apply first the simplifications that lead to r-BP. We define and consider the inner part of the $\chi^{l\to\mu}_{w_l}$ message:

$$\tilde{\psi}^{\nu\to l}_{w_l} = \sum_{s_\nu} \chi^{\nu\to\nu}_{s_\nu} \int \prod_{m\neq l} \left(\mathrm{d}w_m \chi^{m\to\nu}_{w_m}\right) P_0(s_\nu|F_\nu.w) \tag{18}$$

We set $z_\nu = F_{\nu l}w_l + \sum_{m\neq l} F_{\nu m}w_m$. By independence of the ws the partial sum behaves like a Gaussian with mean and variance

$$\omega_{\nu\to l} = \sum_{m\neq l} F_{\nu m} a_{m\to\nu} \quad V_{\nu\to l} = \sum_{m\neq l} F^2_{\nu m} v_{m\to\nu} \tag{19}$$

with

$$a_{m\to\nu} = \int \mathrm{d}w_m \chi^{m\to\nu}_{w_m} w_m \quad v_{m\to\nu} = \int \mathrm{d}w_m \chi^{m\to\nu}_{w_m} w_m^2 - a^2_{m\to\nu} \tag{20}$$

We replace the integral over all ws by a Gaussian integral over z_ν; we obtain

$$\tilde{\psi}^{\nu\to l}_{w_l} = \sum_{s_\nu} \chi^{\nu\to\nu}_{s_\nu} \int \frac{\mathrm{d}z_\nu}{\sqrt{2\pi V_{\nu\to l}}} e^{-(z_\nu - F_{\nu l}w_l - \omega_{\nu\to l})^2/2V_{\nu\to l}} P_0(s_\nu|z_\nu) \tag{21}$$

We can simplify. $F_{\nu l}$ is small, we expand the exponential:

$$e^{-\frac{(z_\nu - F_{\nu l}w_l - \omega_{\nu\to l})^2}{2V_{\nu\to l}}} = e^{-\frac{(z_\nu - \omega_{\nu\to l})^2}{2V_{\nu\to l}}} \left(1 - \frac{F^2_{\nu l}w_l^2}{2V_{\nu\to l}} + \frac{(z_\nu - \omega_{\nu\to l})F_{\nu l}w_l}{V_{\nu\to l}} + \frac{(z_\nu - \omega_{\nu\to l})^2 F^2_{\nu l}w_l^2}{2V^2_{\nu\to l}}\right) \tag{22}$$

We introduce the denoising function; its expression differs from the one of Zdeborová & Krzakala (2016):

$$g_o(\omega,\chi,V) = \frac{\int \mathrm{d}z \sum_s \chi_s P_0(s|z)(z-\omega)e^{-(z-\omega)^2/2V}}{V\int \mathrm{d}z \sum_s \chi_s P_0(s|z) e^{-(z-\omega)^2/2V}} \tag{23}$$

So

$$\tilde{\psi}^{\nu\to l}_{w_l} \propto \left(1 - \frac{F^2_{\nu l}w_l^2}{2V_{\nu\to l}} + g_o F_{\nu l}w_l + \frac{1}{2}\left(\frac{1}{V_{\nu\to l}} + \partial_\omega g_o + g_o^2\right) F^2_{\nu l}w_l^2\right) \tag{24}$$

where we evaluate g_o in $(\omega_{\nu\to l}, \chi^{\nu\to\nu}, V_{\nu\to l})$. We exponentiate:

$$\tilde{\psi}^{\nu\to l}_{w_l} \propto e^{g_o F_{\nu l}w_l + \frac{1}{2}\partial_\omega g_o F^2_{\nu l}w_l^2} \tag{25}$$

We take the product of the $\tilde{\psi}$ to obtain

$$\chi_{w_l}^{l\to\mu} \propto P_w(w_l) e^{-\Lambda_{l\to\mu} w_l^2/2 + \Gamma_{l\to\mu} w_l} \tag{26}$$

where

$$\Lambda_{l\to\mu} = -\sum_{\nu\neq\mu} \partial_\omega g_o(\omega_{\nu\to l}, \chi^{\nu\to\nu}, V_{\nu\to l}) F_{\nu l}^2 \quad , \quad \Gamma_{l\to\mu} = \sum_{\nu\neq\mu} g_o(\omega_{\nu\to l}, \chi^{\nu\to\nu}, V_{\nu\to l}) F_{\nu l} \tag{27}$$

We close the loop defining the input functions

$$f_a(\Lambda, \Gamma) = \frac{\int \mathrm{d}w\, P_w(w) w e^{-\Lambda w^2/2 + \Gamma w}}{\int \mathrm{d}w P_w(w) e^{-\Lambda w^2/2+\Gamma w}} \quad , \quad f_v(\Lambda, \Gamma) = \partial_\Gamma f_a(\Lambda, \Gamma) \tag{28}$$

so

$$a_{l\to\mu} = f_a(\Gamma_{l\to\mu}, \Lambda_{l\to\mu}) \quad , \quad v_{l\to\mu} = f_v(\Gamma_{l\to\mu}, \Lambda_{l\to\mu}) \tag{29}$$

The mean and the variance of the marginals are estimated by

$$a_l = f_a(\Gamma_l, \Lambda_l) \quad , \quad v_l = f_v(\Gamma_l, \Lambda_l) \tag{30}$$

where

$$\Lambda_l = -\sum_{\nu} \partial_\omega g_o(\omega_{\nu\to l}, \chi^{\nu\to\nu}, V_{\nu\to l}) F_{\nu l}^2 \quad , \quad \Gamma_l = \sum_{\nu} g_o(\omega_{\nu\to l}, \chi^{\nu\to\nu}, V_{\nu\to l}) F_{\nu l} \tag{31}$$

We obtain also the expression of the GLM-to-SBM message

$$\psi_{s_\mu}^{\mu\to\mu} = \frac{1}{\sqrt{2\pi V_\mu}} \int \mathrm{d}z P_0(s_\mu|z) e^{-(z-\omega_\mu)^2/2V_\mu} \tag{32}$$

where

$$\omega_\mu = \sum_m F_{\mu m} a_{m\to\mu} \quad , \quad V_\mu = \sum_m F_{\mu m}^2 v_{m\to\mu} \tag{33}$$

A.2.2 TIME INDICES

There are two possibilities for mixing the GLM part and the SBM part:

$$\begin{tikzcd}
a, v^{(t)} \arrow[r] & \omega, V^{(t+1)} \arrow[r] \arrow[d] & \Gamma, \Lambda^{(t+1)} \arrow[r] & a, v^{(t+1)} \\
& \psi^{(t+1)} \arrow[dr] & g_o^{(t+1)} \arrow[u] & \\
& \chi^{(t)} \arrow[r] & \chi^{(t+1)} \arrow[u, Rightarrow] &
\end{tikzcd}$$

or

$$\begin{tikzcd}
a, v^{(t)} \arrow[r] & \omega, V^{(t+1)} \arrow[r] \arrow[d] & \Gamma, \Lambda^{(t+1)} \arrow[r] & a, v^{(t+1)} \\
& \psi^{(t+1)} \arrow[dr] & g_o^{(t+1)} \arrow[u] & \\
& \chi^{(t)} \arrow[r] \arrow[ur, Rightarrow] & \chi^{(t+1)} &
\end{tikzcd}$$

We try both; we do not observe any numerical difference.

A.2.3 AMP

Then we go from r-BP to AMP. We remove the dependence of the messages on the target. We keep only the marginals. The derivation is given by Zdeborová & Krzakala (2016). We obtain that

$$V_\mu^{(t+1)} = \sum_l F_{\mu m}^2 v_l^{(t)} \tag{34}$$

$$\omega_\mu^{(t+1)} = \sum_l F_{\mu l} a_l^{(t)} - V_\mu^{(t+1)} g_{o,\mu}^{(t)} \tag{35}$$

$$g_{o,\mu}^{(t+1)} = g_o(\omega_\mu^{(t+1)}, \chi^{\mu\to\mu,(t)}, V_\mu^{(t+1)}) \tag{36}$$

$$\Lambda_l^{(t+1)} = -\sum_\mu F_{\mu l}^2 \partial_\omega g_o(\omega_\mu^{(t+1)}, \chi^{\mu\to\mu,(t)}, V_\mu^{(t+1)}) \tag{37}$$

$$\Gamma_l^{(t+1)} = \Lambda_l^{(t+1)} a_l^{(t)} + \sum_\mu F_{\mu l} g_{o,\mu}^{(t+1)} \tag{38}$$

A.2.4 Further simplifications

$F_{\mu m}^2$ self-averages. We can replace it by its average $1/M$ in eq. equation 34 and equation 37. So Λ and V become scalars. Also, on average, $-\partial_\omega g_{o,\mu} = g_{o,\mu}^2$. We obtain the algorithm given in the main part.

B Linearization and partial recovery threshold

We take $P_{s,\mu}(s) = 1/2$. The non-informative point $q_S = q_W = 0$ is a fixed-point of the AMP–BP algorithm. At this point, we have $\chi^{\mu\to\nu} = \frac{1}{2}$, $\chi^{\mu\to\mu} = \frac{1}{2}$, $\chi^{\mu\to\nu} = \frac{1}{2}$, $a_l = 0$, $v_l = 1$, $\omega_\mu = 0$, $V = 1$, $\psi^{\mu\to\mu} = \frac{1}{2}$, $g_{o,\mu} = 0$, $\Lambda = 0$ and $\Gamma_l = 0$.

We linearize the equations of the algorithm around this point. We write $|_*$ the evaluation of functions in this point. We have

$$\delta\chi^{\mu\to\nu,(t+1)} = \sum_{\eta\in\partial\mu\setminus\nu} \frac{1}{2}\left(\frac{c_{.,.}}{c} - 1\right).\delta\chi^{\eta\to\mu,(t)} + \partial_\omega \psi^{\mu\to\mu}|_* \delta\omega_\mu^{(t+1)} + \partial_V \psi^{\mu\to\mu}|_* \delta V^{(t+1)} \tag{39}$$

$$\delta\chi^{\mu\to\mu,(t+1)} = \sum_{\eta\in\partial\mu} \frac{1}{2}\left(\frac{c_{.,.}}{c} - 1\right).\delta\chi^{\eta\to\mu,(t)} \tag{40}$$

$$\delta a_l^{(t+1)} = \partial_\Lambda f_a|_* \delta\Lambda^{(t+1)} + \partial_\Gamma f_a|_* \delta\Gamma_l^{(t+1)} \tag{41}$$

$$\delta v_l^{(t+1)} = \partial_{\Lambda\Gamma} f_a|_* \delta\Lambda^{(t+1)} + \partial_{\Gamma\Gamma} f_a|_* \delta\Gamma_l^{(t+1)} \tag{42}$$

$$\delta g_{o,\mu}^{(t+1)} = \partial_\omega g_o|_* \delta\omega_\mu^{(t+1)} + \nabla_\chi g_o|_* .\delta\chi^{\mu\to\mu,(t+1)} + \partial_V g_o|_* \delta V^{(t+1)} \tag{43}$$

where we write $c_{.,.}$ for the affinity matrix and where we have used the standard linearization for SBM. We have also

$$\delta\omega_\mu^{(t+1)} = \sum_l F_{\mu l} \delta a_l^{(t)} - \delta V^{(t+1)} g_o|_* - V|_* \delta g_{o,\mu}^{(t)} \tag{44}$$

$$\delta V^{(t+1)} = \frac{1}{M}\sum_l \delta v_l^{(t)} \tag{45}$$

$$\delta\Lambda^{(t+1)} = \frac{2}{M}\sum_\mu g_o|_* \delta g_{o,\mu}^{(t+1)} \tag{46}$$

$$\delta\Gamma_l^{(t+1)} = \delta\Lambda^{(t+1)} a_l|_* + \Lambda|_* \delta a_l^{(t)} + \sum_\mu F_{\mu l} \delta g_{o,\mu}^{(t+1)} \tag{47}$$

We simplify: $g_o|_* = 0$, $\partial_\omega g_o|_* = 0$, $\partial_V g_o|_* = 0$, $\partial_V \psi|_* = 0$ and $\partial_\Gamma f_a|_* = 1$. We compute that $\partial_\omega \psi|_* = \frac{1}{\sqrt{2\pi}} \left(\begin{smallmatrix} 1 \\ -1 \end{smallmatrix}\right)$ and $\nabla_\chi g_o|_* = \frac{2}{\sqrt{2\pi}} \left(\begin{smallmatrix} 1 \\ -1 \end{smallmatrix}\right)^T$. We assemble equations together:

$$\delta\chi^{\mu\to\nu,(t+1)} = \sum_{\eta\in\partial\mu\setminus\nu} \frac{1}{2}\left(\frac{c_{.,.}}{c} - 1\right).\delta\chi^{\eta\to\mu,(t)} + \partial_\omega\psi|_* \left(\sum_l F_{\mu l}\delta a_l^{(t)} - \nabla_\chi g_o|_*.\delta\chi^{\mu\to\mu,(t)}\right) \tag{48}$$

$$\begin{aligned}\delta\chi^{\mu\to\nu,(t+1)} = &\sum_{\eta\in\partial\mu\setminus\nu} \frac{1}{2}\left(\frac{c_{.,.}}{c} - 1\right).\delta\chi^{\eta\to\mu,(t)} \\ &+ \sum_{\eta,l} F_{\mu l}F_{\eta l}(\partial_\omega\psi|_*.\nabla_\chi g_o|_*).\delta\chi^{\eta\to\eta,(t)} - (\partial_\omega\psi|_*.\nabla_\chi g_o|_*).\delta\chi^{\mu\to\mu,(t)}\end{aligned} \tag{49}$$

$$\delta\chi^{\eta\to\eta,(t)} = \sum_{\rho\in\partial\eta} \frac{1}{2}\left(\frac{c_{.,.}}{c} - 1\right).\delta\chi^{\rho\to\eta,(t-1)} \tag{50}$$

The matrices $\frac{1}{2}\left(\frac{c_{.,.}}{c} - 1\right)$ and $\partial_\omega\psi|_*.\nabla_\chi g_o|_*$ share the same eigenvectors. They have one null eigenvalue and one positive: $\frac{c_i - c_o}{2c} = \frac{\lambda}{\sqrt{c}}$ and $\frac{2}{\pi}$. We project to obtain

$$x^{\mu\to\nu,(t+1)} = \frac{\lambda}{\sqrt{c}}\left(\sum_{\eta\in\partial\mu\setminus\nu} x^{\eta\to\mu,(t)} + \frac{2}{\pi}\sum_\eta (F.F^T)_{\mu,\eta}\sum_{\rho\in\partial\eta} x^{\rho\to\eta,(t-1)} - \frac{2}{\pi}\sum_{\eta\in\partial\mu} x^{\eta\to\mu,(t-1)}\right) \tag{51}$$

$$x^{\mu\to\nu,(t+1)} = \frac{\lambda}{\sqrt{c}}\left(\sum_{\eta\in\partial\mu\setminus\nu} x^{\eta\to\mu,(t)} + \frac{2}{\pi}\sum_\eta (F.F^T - I_N)_{\mu,\eta}\sum_{\rho\in\partial\eta} x^{\rho\to\eta,(t-1)}\right) \tag{52}$$

where the xs are real random variables.

We obtain the threshold λ_c of partial recovery taking the mean of the variance of the expression 52, discarding the time indices. We use that $(F.F^T - I_N)^2_{\mu,\nu}$ averages to $1/M$ if $\mu \neq \nu$ and to $\mathcal{O}(1/M)$ otherwise. We obtain:

$$1 = \lambda_c^2\left(1 + \frac{4\alpha}{\pi^2}\right) \tag{53}$$

C Baseline unsupervised algorithm

We perform n steps of graph convolution on the features; perform PCA on the transformed features and keep the largest component; threshold its left vector to obtain the membership of each node. Formally, we consider the rows $F_\mu^{(0)} \in \mathbb{R}^M$ of the feature matrix F; we apply n times

$$F_\mu^{(t+1)} = F_\mu^{(t)} + a\sum_{\nu\in\partial\mu} F_\nu^{(t)} \tag{54}$$

where a is a scalar. We apply PCA on the new matrix $\hat{F}$ whose rows are $F_\mu^{(n)}$. Writing $u \in \mathbb{R}^N$ the left vector of its largest component, the estimator is $\hat{s} = \mathrm{sign}(u)$. We tune n and a empirically to optimize the recovery. We observe that roughly it depends on n and a only by their product an. Also, the optimal a scales like $1/c$.

D Performances of AMP–BP

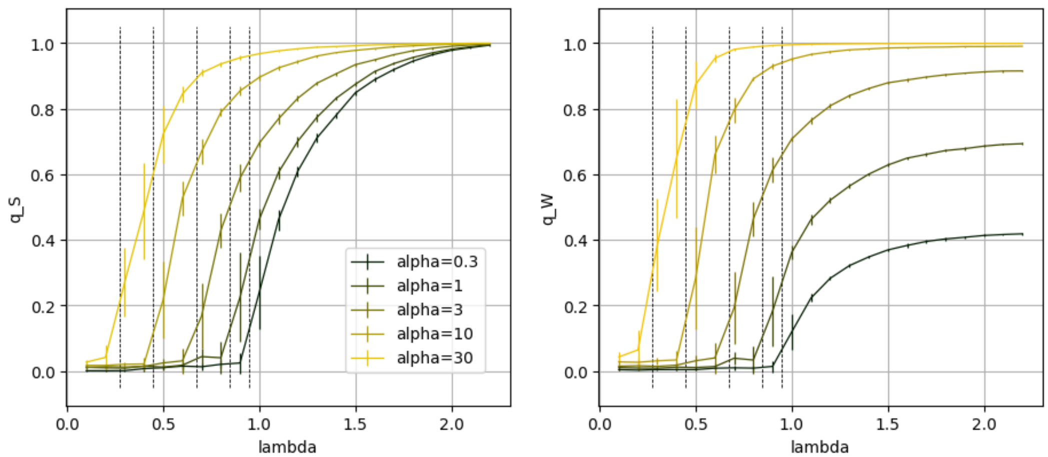


Figure 2: Left and right: overlaps q_S and q_W of the fixed-point of the algorithm AMP–BP, vs λ for many compression ratios αs. Vertical dashed lines: theoretical thresholds λ_c to partial recovery. $N = 10^4$, $c = 5$. We ran ten experiments per point.

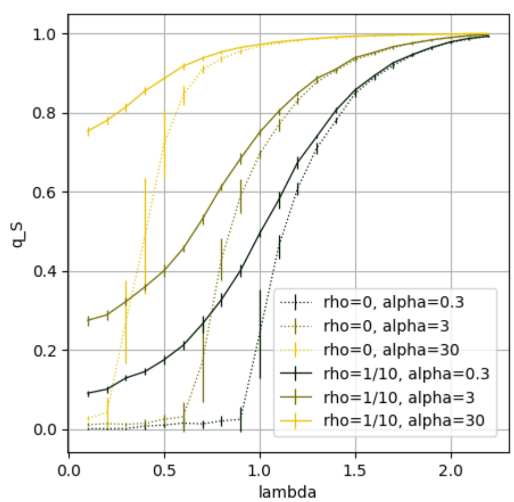


Figure 3: Semi-supervised case: test overlap q_S of the fixed-point of the algorithm AMP–BP, vs λ for many compression ratios αs. ρ is the proportion of train nodes; $\rho = 0$ for unsupervised. Semi-supervised always performs better than unsupervised. $N = 10^4$, $c = 5$. We ran ten experiments per point.