Inferring Block Structure of Graphical Models in Exponential Families

Siqi Sun* **Hai Wang*** **Jinbo Xu**
Toyota Technological Institute at Chicago, USA 60637
{siqi.sun, haiwang, j3xu}@ttic.edu

Abstract

Learning the structure of a graphical model is a fundamental problem and it is used extensively to infer the relationship between random variables. In many real world applications, we usually have some prior knowledge about the underlying graph structure, such as degree distribution and block structure. In this paper, we propose a novel generative model for describing the block structure in general exponential families, and optimize it by an Expectation-Maximization(EM) algorithm with variational Bayes. Experimental results show that our method performs well on both synthetic and real data. Furthermore, our method can predict overlapping block structure of a graphical model in general exponential families.

1 INTRODUCTION

Graphical models are an important tool to describe and model real world data in various fields such as nature language processing, computation biology and image analysis. Learning the structure of a graphical model is thus essential since it provides a convenient way to model conditional independence among variables for further analysis. Plenty of work has been done for some specific settings of this structure learning problem. For Gaussian Graphical Models (GGMs), it is well known that conditional independence is encoded in the precision matrix. Neighborhood estimation [19] and log-likelihood maximization with l_1 penalty [5, 11, 30] have been developed to estimate the structure of GGMs. Furthermore, for the Ising, Poisson and other models in exponential families, a consistent neighborhood estimator is proposed by Ravikumar et al [24] and Yang et al [29], which apply a logistic regression and a generalized linear model with l_1 penalty, respectively, to learn the underlying graph structure.

Recently, incorporating prior knowledge into structure learning has drawn much attention because the graph under estimation in many real-world applications usually has some intrinsic properties, such as scale free [9, 16], block structure [3, 17, 22, 26] and other topological constraints [10]. Amongst those properties, the block structure of a graph is of special interest for biological networks. For example, in a protein-protein interaction network, proteins are more likely to form a pathway or complex to expression a specific function [25]. It is straightforward to estimate the graph first, and then apply a clustering algorithm such as spectral clustering [27, 28] or Mixed-Membership Stochastic Blockmodel (MMSB) [1, 12, 13] to obtain block structure. However, simultaneously inferring the graph and block structure may improve the result in terms of both cluster accuracy and graph estimation [22, 26], especially when the data is limited. To the best of our knowledge, existing works that focusing on block structure in graphical models are all based on Gaussian graphical models [3, 15, 17, 18, 26] or factor models [22], and can only be used to infer hard clustering and non-overlapping block structure.

This paper proposes a generative model that can (1) apply to a graphical model on exponential families, (2) infer the underlying soft clusters as well as overlapping blocks, and (3) generate graph and block structure at the same time with better accuracy than those generated by a heuristic approach. The reason why we want to infer them together is that the hidden block structure can actually help decide the penalty parameters. For example, we might increase the penalty parameters if two nodes are from different blocks because the probability that there is an edge between them is small, and also for the converse. In the rest of the paper, we first describe some related work and notations in Section 2, and then present our method in Section 3. Finally, the results for synthetic data and a microRNA network data and the conclusions are presented in Section 4 and 5, respectively.

Appearing in Proceedings of the 18^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. JMLR: W&CP volume 38.

*These two authors contributed equally to this work

2 RELATED WORK & NOTATION

Let $x = (x_1, x_2, \ldots, x_p)$ be a random vector from a multivariate Gaussian distribution, $X_{n\times p}$ the observed data, and n the number of samples. It has been proved that x_i and x_j are conditionally independent if and only if $\Omega_{ij} = 0$, where Ω is the precision matrix of the Gaussian distribution [19]. We can also use a graph $G = (V, E)$ to represent the relationship among the p random variables $x_1, x_2, ..., x_p$ where V and E are the vertex and edge sets, respectively. Meanwhile, V is the set of the p variables and E models conditional dependency between the variables. To obtain a sparse graph, some methods have been proposed to estimate the sparse precision matrix [5, 11, 19, 30] by maximizing the l_1 penalized log-likelihood, i.e.,

$$\hat{\Omega} = \arg\max_{\Omega\succ 0} \log\det(\Omega) - tr(\Omega\hat{\Sigma}) - \lambda\sum_{i\le j}|\Omega_{ij}|, \quad (1)$$

where λ is the regularizer that controls the sparsity level and $\hat{\Sigma}$ is the empirical covariance matrix. The formulation defined in Eq. (1) can be efficiently optimized by graphical lasso [11].

Another line of work first estimates the neighborhood $\hat{N}(x_a)$ for each random variable x_a, i.e. node a, and then constructs the graph as the union of all neighborhoods.

In GGMs, the neighborhood of one node can be estimated by Lasso as follows,

$$\hat{w}_a = \arg\min \frac{1}{n}\|X_a - X_{-a}w_a\|_2^2 + \lambda\|w_a\|_1, \quad (2)$$

where X_a is the a-th column of X and X_{-a} is the whole data matrix excluding column a.

More generally, consider the graphical model where its conditional distribution can be written as an exponential form with linear interactions, i.e.,

$$P(x_a|x_{-a}, w_a) = q_0(x_a)\exp(\sum_b w_{ab}x_a x_b - D(x_{-a}, w_a)), \quad (3)$$

where x_{-a} indicates all random variables except x_a, $q_0(x)$ is the base measure, w_a is the model parameter and $D(x_{-a}, w_a)$ is the log-normalizing constant. Given the observed data $X_{n\times p}$, the conditional log-likelihood is

$$l(X_a, w_a) = \frac{1}{n}\sum_{i=1}^{n}\sum_{b\neq a} w_{ab}X_a^i X_b^i + D(w_a, X_{-a}), \quad (4)$$

and its l_1 regularized estimator is

$$\hat{w}_a = \arg\min_{w_a} l(X_a, w_a) + \lambda_a\|w_a\|_1. \quad (5)$$

The neighbor of a can then be estimated as $\hat{N}(a) = \{b \in V/a : \hat{w}_{ab} \neq 0\}$. The problem (5) is usually a linear model (in GGMs) or a generalized linear model (in Ising or Potts model) with l_1 penalty, which can be solved efficiently by methods like iterative soft-thresholding [7]. The advantage of this estimator is that it has both sparsity and consistency [19, 24, 29]. Further it can be easily implemented and parallelized, so it is scalable for very large scale data.

For graphical models with block structure, most of existing work focused on GGMs. Marlin et al [17] proposed a two-stage Bayesian model, in which a spike and slab like prior based on network with block structure is applied to generate model parameters (w), and the data is then generated by linear regression given w. The posterior is optimized by variational inference.

Ambroise et al [3] used hidden indicator variables Z to denote the cluster assignment, where only one element of Z is equal to one, and all the others are zero. Then the precision matrix is estimated by maximizing the log complete-data likelihood spreading over Z. Since summing up all configurations of Z is intractable, an EM algorithm is employed to perform the inference.

Determining the number of clusters K is still challenging, and the two previous methods use a heuristic split approach and ICL (integrated complete likelihood) criterion respectively. To solve this issue, Marlin et al [18] introduced a novel prior and performed MAP estimation on the model so that the model can automatically determine the number of blocks. Furthermore, Sun et al [26] proposed a Bayesian method that used Chinese Restaurant Process and Wishart prior to model the number of clusters and precision matrix Ω, respectively, and Gibbs sampling to estimate the posterior of block membership variables. Besides, Palla et al [22] proposed a nonparametric Bayesian method to cluster variables in factor models. Note that nearly all of these work only focus on GGMs, and assume a non-overlapping or hard clustering block structure.

3 Methods

In this section we present a generative process to model the dependency between random variables. Assuming that the number of clusters K is known, our method can be briefly described as follows. We first use MMSB to generate a block structured network. That is, we sample the hidden variables Θ from some prior distribution $P(\Theta|\alpha, \eta)$ specified by MMSB, where α and η are hyper parameters. Then given Θ, we sample the model parameters w from a Laplace prior $P(w|\Theta)$. Finally, we sample the data from $P(X|w)$ in a pseudo-likelihood manner.

3.1 A Generative Model

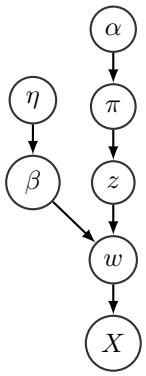

Figure 1: Graphical model representation for cluster structure

We describe our model in a top-down fashion:

- Sample cluster strength $\beta_k \sim$ Beta (η) for each cluster $1, \ldots, K$.
- Sample cluster membership $\pi_a \sim$ Dirichlet(α) for each node $a = 1, \ldots, p$.
- For each pair of nodes a and b,
  - Sample interaction indicator $z_{a\to b} \sim \pi_a$.
  - Sample interaction indicator $z_{a\leftarrow b} \sim \pi_b$.
  - Compute $r_{ab} = \beta_{z_{a\to b}}^{\mathbf{1}(z_{a\to b}=z_{a\leftarrow b})} \epsilon^{1-\mathbf{1}(z_{a\to b}=z_{a\leftarrow b})}$
  - Sample $w_{ab} \sim$ Laplace$(\rho_{ab}(r_{ab})) = \frac{1}{2\rho_{ab}} \exp(-\frac{|w_{ab}|}{\rho_{ab}})$
- For each node a,
  - Fit the data by generalized linear model, i.e. $x_a|x_{-a} \sim q_0(x_a)\exp(\sum_{b\neq a} w_{ab}x_a x_b - D(x_{-a}, w_a))$,

where ϵ is the probability that there is an edge between different clusters. $z_{a\to b}, z_{a\leftarrow b}$ are K dimension indicator vectors, i.e. $z_{a\to b}^k = 1$ means node a is in cluster k. We also use $z_{a\to b} = k$ to denote the same thing if the context is clear. The relationship between random variables is also described in Figure 1. In the first three steps, the algorithm generates a graph with overlapped block structure, and in the last step, it fits the data by conditional distribution in Eq. 3. Note that in the third step, $\rho_{ab}(r_{ab})$ is a function that decides the penalty for edge (a, b). The higher the value of ρ_{ab}, the more likely that there is an edge between nodes a and b. Since r_{ab} is the probability that there is an edge between a and b, we set $\rho_{ab} = c \cdot r_{ab}$, where c is a hyper parameter, so that the penalty for higher r_{ab} is lower, and vice versa. To simplify the notation, we denote Θ as the union of hidden parameters $\{\beta, \pi, z\}$.

Based upon the proposed model above, we can write down the complete data likelihood $P(X, w, \Theta)$.

Proposition 1. The complete data likelihood of the model described above can be written as

$$\begin{aligned} P(X, w, \Theta) &= P(X|w)P(w|\Theta)P(\Theta) \\ &= \prod_{i=1}^{n} \exp(\sum_{a=1}^{p}\sum_{b=1}^{p} w_{ab}X_a^i X_b^i + C(X_a) - D(X_{-a})) \\ &\quad \prod_{a,b} \frac{1}{2\rho_{ab}(r_{ab})} \exp(-\frac{|w_{ab}|}{\rho_{ab}(r_{ab})}) \\ &\quad \prod_k \frac{1}{B(\eta_k, \eta_k)} \beta_k^{\eta_k - 1}(1-\beta_k)^{\eta_k - 1} \\ &\quad \prod_{a\leq b}\prod_k \pi_{a,k}^{z_{a\to b}^k} \pi_{b,k}^{z_{a\leftarrow b}^k} \prod_a \frac{1}{B(\alpha)} \prod_k \pi_{a,k}^{\alpha_k - 1}, \end{aligned} \tag{6}$$

where $B(\eta, \eta)$ is a Beta function and $B(\alpha) = \frac{\prod_{i=1}^K \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^k \alpha_i)}$ is the multivariate Beta function. The proof is obvious according to Bayes rule.

3.2 Optimization

We estimate w by maximizing its posterior given data X, i.e.

$$\hat{w} = \arg\max_w \log P(w|X) = \arg\max_w \log P(X, w). \tag{7}$$

However, the distribution of w depends on the hidden variable Θ, which encodes all the cluster structure information. If we marginalize over the hidden variables Θ, we obtain

$$\hat{w} = \arg\max_w \log \sum_\Theta P(X, w, \Theta), \tag{8}$$

where the summation is intractable because it is over all possible values of the hidden variables. Here we introduce an EM algorithm to optimize it. To do so we need to compute the following conditional expectation:

$$\begin{aligned} Q(w|w^t) &= E_{\Theta|X,w^t}[\log P(X, w, \Theta)] \\ &= \sum_\Theta P(\Theta|w^t) \log P(X, w, \Theta). \end{aligned} \tag{9}$$

Unfortunately, the calculation of $Q(w|w^t)$ is still intractable since $P(\Theta|w^t)$ cannot be factorized. To deal with this, we first use variational approach to approximate $P(\Theta|w^t)$ and then the classical EM approach.

3.2.1 Expectation Step

In the E step, the parameter w is assumed to be known as w^t. Here we need to approximate the posterior of Θ by its

mean field variational probability density $q(\Theta)$. Therefore $Q(w|w^t)$ can be approximated by

$$\hat{Q}(w|w^t) = \sum_{\Theta} q(\Theta) \log P(X, w, \Theta).$$

To simplify the calculation, we approximate the Laplace distribution $P(w_{ab}|\Theta)$ by discretizing w_{ab} into a binary variables $\hat{w}_{ab}$. More specifically, we approximate $P(\Theta|w^t)$ as

$$\begin{aligned} P(\beta, \pi, z|w) =& \frac{1}{P(w)} P(w|z, \beta) P(z|\pi) P(\pi|\alpha) P(\beta|\eta) \\ \approx& \prod_k q(\beta_k|\lambda_k) \prod_a q(\pi_a|\gamma_a) \\ & \prod_{a \le b} q(z_{a\to b}|\phi_{a\to b}) q(z_{a\leftarrow b}|\phi_{a\leftarrow b}), \end{aligned} \quad (10)$$

where $q(\cdot)$ is variational distribution, and λ, γ, ϕ are variational parameters. To narrow the gap between its variational approximation and posterior distribution, we maximize the evidence lower bound (ELBO), which is equivalent to minimizing the KL divergence. To prove the equivalence, note that

$$KL(q(\Theta) \| P(\Theta|w^t)) = \sum_{\Theta} q(\Theta) \log \frac{q(\Theta)}{P(\Theta, w_t)} + \log P(w_t), \quad (11)$$

where the first term in the right hand side of Eq. (11) is negative ELBO.

Proposition 2. Supposing that $q(\Theta)$ can be decomposed according to Eq. 10, the approximated ELBO (denoted as L) then can be written as:

$$\begin{aligned} L \approx& E_q[\log P(\bar{w}|z, \beta) P(z|\pi) P(\beta) P(\pi)] \\ & - E_q[\log q(\beta|\lambda) q(z|\phi) q(\pi|\gamma)] \\ =& \sum_k E_q[\log p(\beta_k|\eta_k)] - \sum_k E_q[\log q(\beta_k|\lambda_k) \\ & + \sum_a E_q[\log p(\pi_a|\alpha) - \sum_a E_q[\log q(\pi_a|\gamma_a)] \\ & + \sum_{a,b} E_q[\log p(z_{a\to b}|\pi_a)] + E_q[\log p(z_{a\leftarrow b}|\pi_b)] \\ & - \sum_{a,b} E_q[\log q(z_{a\to b}|\phi_{a\to b})] + E_q[\log q(z_{a\leftarrow b}|\phi_{a\leftarrow b})] \\ & + \sum_{a,b} E_q[\log p(\bar{w}_{ab}|z_{a\to b}, z_{a\leftarrow b}, \beta)]. \end{aligned} \quad (12)$$

For detailed derivation, which will be used for next proposition, please refer to appendix A.

The maximization procedure is usually optimized by coordinate ascent algorithm. However, it would be extremely slow when the number of nodes is large. The reason is that before updating global variables λ and γ, we need to compute all n^2 pairs of local variables $(\phi_{a\to b}, \phi_{a\leftarrow b})$, which is a waste in the first several iterations because the parameters are initialized randomly. Hence we apply stochastic variational inference (SVI) [12, 14] to further speed up computation. In each step of SVI, we use a noisy estimate of gradient from a subsample of nodes for global variables.

Proposition 3. Given a pair of nodes (a, b), the estimated gradient for each global variable is

$$\begin{aligned} \partial\gamma_{a,k} &= \alpha_k + \frac{N(N-1)}{2} \phi^k_{a\leftarrow b} - \gamma_{a,k} \\ \partial\lambda_{k,i} &= \eta_{k,i} + \frac{N(N-1)}{2} \phi^k_{a\leftarrow b} \cdot \phi^k_{a\to b} \cdot \hat{w}_{ab,i} - \lambda_{k,i}, \end{aligned} \quad (13)$$

and the optimal for local variables are

$$\begin{aligned} \phi^k_{a\to b}|\hat{w}_{ab} = 1 \propto \exp\Big(& \psi(\gamma_{ak}) - \psi(\gamma_a) + \phi^k_{a\leftarrow b}(\psi(\lambda_{k1}) \\ & - \psi(\lambda_k)) + (1 - \phi^k_{a\leftarrow b}) \log \epsilon \Big) \\ \phi^k_{a\to b}|\hat{w}_{ab} = 0 \propto \exp\Big(& \psi(\gamma_{ak}) - \psi(\gamma_a) + \psi^k_{a\leftarrow b}(\psi(\lambda_{k2}) \\ & - \psi(\lambda_k)) + (1 - \phi^k_{a\leftarrow b}) \log(1 - \epsilon) \Big), \end{aligned} \quad (14)$$

where $\psi(x)$ is the digamma function, $\hat{w}_{ab,1} = \hat{w}_{ab}$, $\hat{w}_{ab,2} = 1 - \hat{w}_{ab}$, $\gamma_a = \sum_k \gamma_{ak}$ and $\lambda_k = \lambda_{k1} + \lambda_{k2}$. The derivation for $\phi_{a\leftarrow b}$ is similar.

Sketched Proof:
Following proposition 2 and detailed computation in Appendix, the gradient for $\gamma_{a,j}$ is

$$\frac{\partial L}{\gamma_{a,j}} = \sum_{k=1}^{K} -\frac{\partial E_q[\log \pi_{a,k}]}{\partial \gamma_{a,j}} \left(\frac{N(N-1)}{2} \phi_{a\leftarrow b,k} + \alpha_k - \gamma_{a,k} \right) \quad (15)$$

Given exponential family

$$P(x|w) = q_0(x) \exp(w \cdot T(x) - D(w)), \quad (16)$$

we have

$$\begin{aligned} E_q[\log \pi_{a,k}] =& \frac{\partial D(\gamma)}{\partial \gamma_{a,k}} \\ \frac{\partial^2 \log p(x|w)}{\partial w_i \partial w_j} =& -\frac{\partial^2 D(w)}{\partial w_i \partial w_j} \end{aligned} \quad (17)$$

Substituting $\frac{\partial E_q[\log \pi_{a,k}]}{\partial \gamma_{a,j}}$ in Eq. 15, we have

$$\frac{\partial L}{\gamma_{a,j}} = \sum_{k=1}^{K} -\frac{\partial^2 \log q(\pi_a|\gamma_a)}{\partial \gamma_{a,j} \partial \gamma_{a,k}} \left(\frac{N(N-1)}{2} \phi_{a \leftarrow b,k} + \alpha_k - \gamma_{a,k}\right) \tag{18}$$

After multiplying Eq.18 by the inverse of the Fisher information matrix of q, we obtain the update for global variable γ. The proof for λ is similar.
The derivation for local variables is obvious by using the fact that $E[\log \pi_{ak}] = \psi(\gamma_{ak}) - \psi(\gamma_a)$ if π_a follows the Dirichlet distribution.

We summarize the E step of our algorithm in algorithm 1. We also use the technique proposed by Gopalan et al [13] to determine the number of blocks (i.e., K).

Algorithm 1 Stochastic Variational Inference (E-Step)

Initialize γ_a, λ_k randomly, τ is a parameter
for $t = 1$ to MAX_ITERATION **do**
  Sample a pair of nodes (a, b) randomly
  Compute optimal of local variables based on Eq. 14
  Compute gradient of global variables based on Eq. 13
  Compute step size $s_t = 1/t^{\tau}$
  Update global variables by $g \rightarrow g + s_t \partial g$
end for

3.2.2 Maximization Step

Now we can approximate $Q(w|w^t)$ by substituting $P(\Theta|w^t)$ with its variational approximation $q(\Theta)$, and infer w from $q(\Theta)$. Plugging in the complete log-likelihood in Eq. 6,

$$\begin{aligned} Q(w|w^t) \approx & \hat{Q}(w|w^t) = E_{q(\Theta)}[\log P(X|w) P(w|\Theta) P(\Theta)] \\ = & \sum_i \sum_{a,b} w_{ab} X_a^i X_b^i + D(X_{-a}, w) \\ & - \sum_{a,b} c\Big(\sum_k \phi_{a \rightarrow b}^k \phi_{a \leftarrow b}^k \frac{\lambda_k}{\lambda_{k1}} \\ & + (1 - \sum_k \phi_{a \rightarrow b}^k \phi_{a \leftarrow b}^k) \frac{1}{\epsilon}\Big) |w_{ab}| + C. \end{aligned} \tag{19}$$

Therefore we have the following proposition.

Proposition 5. The M step is equivalent to the following general Lasso-like optimization problem.

$$\begin{aligned} \hat{w} = & \arg\max_w \sum_i \sum_{a,b} w_{ab} X_a^i X_b^i + D(X_{-a}, w) \\ & - \sum_{a,b} \rho_{ab} |w_{ab}|, \end{aligned} \tag{20}$$

where

$$\rho_{ab} = c\Big(\sum_k \phi_{a \rightarrow b}^k \phi_{a \leftarrow b}^k \frac{\lambda_k}{\lambda_{k1}} + (1 - \sum_k \phi_{a \rightarrow b}^k \phi_{a \leftarrow b}^k) \frac{1}{\epsilon}\Big) \tag{21}$$

Note that ρ incorporates all the information we need from hidden variables to compute the penalty term. Therefore we can treat it as a LASSO-like problem, which can be solved efficiently by the fast iterative shrinkage-thresholding algorithm (FISTA) [7] or other similar algorithms.

A Poisson Graphical Model Example
In this part we present an example to show how Proposition 5 works on an exponential family besides GGMs. For Poisson Graphical Models, the log conditional probability density function for node a can be written as

$$\begin{aligned} \log P(X_a|X_{-a}, w_a) = & X_a(w_a + \sum_{b \neq a} w_{ab} X_b) \\ & - \exp(w_a + \sum_{b \neq a} w_{ab} X_b) - \log(X_a!) \end{aligned}$$

Therefore the optimization problem is

$$\begin{aligned} \max_w & -\frac{1}{n} \sum_a \sum_i \Bigg(\exp(w_a + \sum_{b \neq a} w_{ab} X_b^i) + X_a^i w_a + \\ & \sum_{b \neq a} w_{ab} X_b^i X_a^i \Bigg) + \sum_{a,b} \lambda_{ab} |w_{ab}| \\ s.t. \quad & w_{ab} = w_{ba} \quad \forall a, b. \end{aligned} \tag{22}$$

The problem above can easily be optimized by any soft thresholding algorithms. Note that w_a is a scalar indicating the constant term for linear interactions between node a and the others.

We summarize our algorithm as follows.

Algorithm 2 Clustering Graphical Models

Input: $X_{n \times p}$
while $Q(\hat{w^t})$ does not converge **do**
  //E Step by Stochastic Variational Inference
  Compute γ, λ, ϕ by Algorithm 1
  //M Step by Lasso-like algorithm
  Compute ρ by Eq. 21
  Compute w^{t+1} by solving Eq. 20
  t ← t+1
end while

4 Experimental Results

In this section we present some experimental results of our method and its comparison with a heuristic approach, i.e.

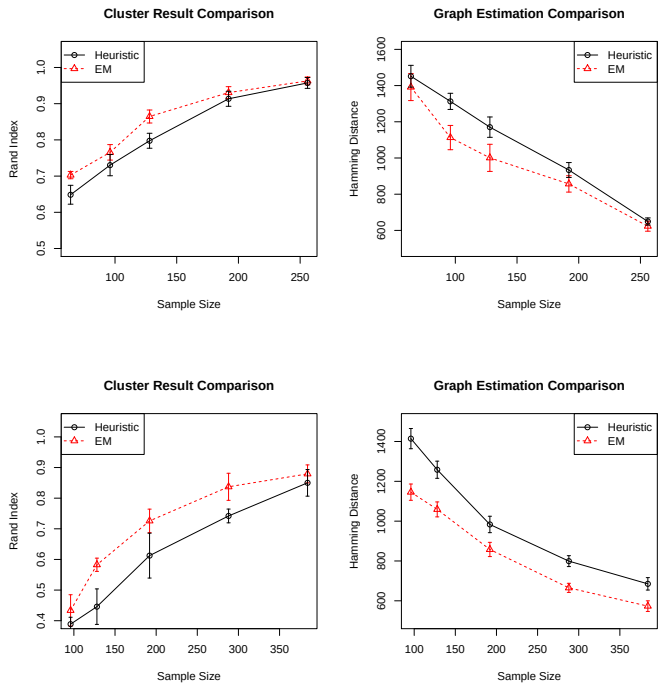

Figure 2: Performance comparison of our method and the heuristic approach for GGMs (Top) and PGMs (Bottom) in terms of Rand Index between clustering result (Left), and Hamming distance between graphs (Right). The standard deviation is computed based on five simulations for each sample size.

learn the graph by generalized linear model first, and then cluster the graph by MMSB, on synthetic data. Rand index and Hamming distance is applied for evaluation purpose. Rand index is a measure for the similarity of two clustering results. Rand index ranges from 0 to 1, with 1 indicating the exact match and 0 the worst. Hamming distance measures the distance between two graphs, with 0 indicating that two graphs are same. We further apply our method to a RNA data set to show that it can detect some biologically meaningful clusters.

4.1 Synthetic Data

Suppose we want to generate a network with p random variables and K clusters. We can use a simple version of MMSB model to generate the graph. In particular, we set β_k (the probability that there is an edge within block k, i.e. cluster strength) to 0.2 for $k = 1, \ldots, K$. We also assign γ_a to block k ($\gamma_{ak} = 1$) with probability $\frac{1}{K}$. Finally we compute r_{ab} according to the third step of our model for each pair of nodes a and b, and sample an edge between a and b by probability r_{ab}. Furthermore, we set ϵ to 0.02, p to 128, and K to 4. Given the graph, we can then sample the data correspondingly.

To evaluate performance of our model more precisely, two kinds of graphical models in the exponential family are used: Gaussian Graphical Models (GGMs) and Poisson Graphical Models (PGMs). We use the heuristic approach (i.e., estimate the graph first before clustering it) as the baseline. Note that the expectation of estimated π_{ak} for each node a can be treated as the probability that node a is in block k. To compare with the ground truth, we assign node a to bock l such that π_{al} is maximized over all π_{ak}, $k = 1, \ldots, K$.

For GGMs, we set the precision matrix element $\Omega_{ab} = 0.3$ if there is an edge between node a and node b, and 0 otherwise. To make sure Ω is positive definite, we set its diagonal to the absolute of the minimum eigenvalue of Ω plus 0.2. For PGMs, the conditional distribution for node a is

$$P(X_a|X_{-a}, w_a) = \exp(X_a \sum_{b \neq a} w_{ab} X_b - \log(X_a!) - A(w_a, X_{-a})), \quad (23)$$

where $\log(X_a!)$ is the base measure and $A(w_a, X_{-a})$ is log-normalizing constant. Some simple algebra can show that $A(w_a, X_{-a}) = \exp(\sum_{b \neq a} w_{ab} X_b)$ and $w_{ab} \leq 0$ for all a, b so that $A(w) < \infty$ [8]. Given the conditional distribution, Gibbs sampling is used to generate simulated data for PGMs.

We set $p = 128$ for both GGMs and PGMs. The simulation is conducted for each sample size N, where

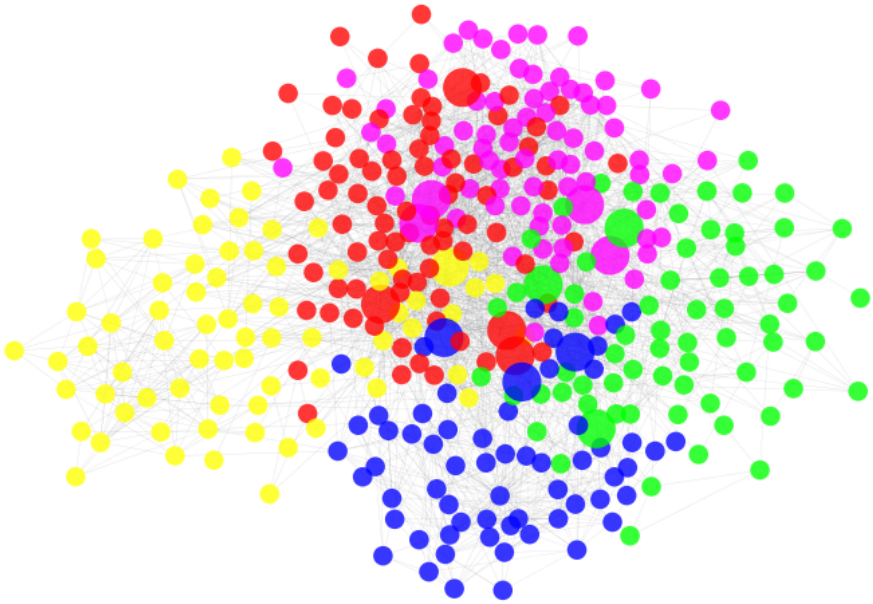

Figure 3: The estimated network from a set of breast cancer miRNA data. The five colors in the graph indicate different clusters, and bigger nodes imply the overlapping nodes. Most of the clusters are clear (Eg. the yellow, blue and green clusters).

$N \in \{64, 96, 128, 192, 256\}$, i.e. $\{0.5, 0.75, 1, 1.5, 2\} \times p$, for GGMs. For PGMs, we set sample size N to $\{96, 128, 196, 256, 384\}$, i.e. $\{0.75, 1, 1.5, 2, 3\} \cdot p$. The simulation results are illustrated in Figure 2. It shows that our algorithm outperforms the heuristic approach for nearly all sample sizes in terms of both clustering accuracy (Rand Index) and graph estimation accuracy (Hamming distance). As for running time, our EM algorithm is about two times slower than heuristic approach since we use warm start.

4.2 Real Data

To test the performance of our method, we apply it to a breast cancer microRNA (miRNA) data set from next generation sequencing data. The data set is obtained from Cancer Genome Atlas (TCGA) [21], and preprocessed according to the method in [2]. The final dataset has 416 variables and 452 samples. An EM Poisson graphical model with $\epsilon = 0.05$ is fitted to estimate and cluster the graph at the same time. Note that GGMs cannot be easily applied here because the data consist of counts. After removing cluster with size less than 5, the final resulting network with 374 miRNAs is illustrated in Figure 3, where the color indicates our clustering result. Bigger nodes indicates "overlapping" nodes that probably play the role of connecting two different blocks. Although we cannot evaluate our result based on Rand index or Hamming distance since the labels of cluster assignment as well as network are not available, there are still several interesting results from a biological perspective. For example, our algorithm identifies 8 overlapping nodes (i.e., miRNAs). Amongst them, it is known that HSA-MIR-146A[23], HSA-MIR-200B[4] and HSA-MIR-200C [4] play a very important role in identifying breast cancer gene targets. In contrast, other models such as GGMs or PGMs cannot identify such targets since only hard clustering is involved. Further, those overlapping node cannot be identified by simply selecting nodes with highest degree since their degrees are 10, 7, 7, 4, 8, 7, 7 and 5 respectively, while the 8-th highest degree of all nodes is 14.

Our clustering result is also consistent with some biological experimental results. For example, in the purple cluster, the non-coding miRNAs HSA-LET-7g, MIR-200C, HSA-MIR-181B-1 and HSA-MIR181B are all associated with Chemoresponse to S-1 in Colon Cancer [20]. In the red cluster, the HSA-LET-7A family members are a modulator of KLK6 protein expression that is independent of the KLK6 copy number status. Further, the miRNAs which have been identified to have no direct relationship with KLK6 copy number status, such as HSA-MIR-296 and HSA-MIR-296, do not apprear in the red cluster[6]. On the other hand, empirical approach fails to detect such information.

In this case, a node is called an overlapping node if it belongs to two blocks with probability larger than or equal to 0.4

5 Conclusions

We present a generative model that can simultaneously detect the overlapping block structure and estimate the graph by applying an EM algorithm with variational inference. Experimental results show that our method outperforms the heuristic approach on both synthetic and real data. In particular, our algorithm not only applies to Gaussian Graphical models, but also to all kinds of models belonging to general exponential families, such as Poisson distribution and multinomial distribution. Besides, our method can obtain a soft clustering result and detect overlapping nodes due to the application of MMSB.

In future we will work on the derivation and improvement of our method in high dimensional data (i.e. $n \ll p$) settings, where it shall be more effective and helpful. The reason is that with enough data, one can always estimate graph first, then cluster it. Furthermore, theoretical results also need to be established for our method, since we do not have consistent guarantee as of yet.

Acknowledgements

This work is supported by the Alfred P. Sloan Research Fellowship to JX and NSF Career Award NSF/CCF AF-1149811 to JX.

References

[1] Edoardo M Airoldi, David M Blei, Stephen E Fienberg, and Eric P Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9(1981-2014):3, 2008.

[2] Genevera I Allen and Zhandong Liu. A log-linear graphical model for inferring genetic networks from high-throughput sequencing data. In *Bioinformatics and Biomedicine (BIBM), 2012 IEEE International Conference on*, pages 1–6. IEEE, 2012.

[3] Christophe Ambroise, Julien Chiquet, Catherine Matias, et al. Inferring sparse gaussian graphical models with latent structure. *Electronic Journal of Statistics*, 3:205–238, 2009.

[4] Raffaele Baffa, Matteo Fassan, Stefano Volinia, Brian O'Hara, Chang-Gong Liu, Juan P Palazzo, Marina Gardiman, Massimo Rugge, Leonard G Gomella, Carlo M Croce, et al. Microrna expression profiling of human metastatic cancers identifies cancer gene targets. *The Journal of pathology*, 219(2):214–221, 2009.

[5] Onureena Banerjee, Laurent El Ghaoui, Alexandre d'Aspremont, and Georges Natsoulis. Convex optimization techniques for fitting sparse gaussian graphical models. In *Proceedings of the 23rd international conference on Machine learning*, pages 89–96. ACM, 2006.

[6] Jane Bayani, Uros Kuzmanov, Punit Saraon, William A Fung, Antoninus Soosaipillai, Jeremy A Squire, and Eleftherios P Diamandis. Copy number and expression alterations of mirnas in the ovarian cancer cell line ovcar-3: Impact on kallikrein 6 protein expression. *Clinical chemistry*, 59(1):296–305, 2013.

[7] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[8] Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 192–236, 1974.

[9] Aaron J Defazio. A convex formulation for learning scale-free networks via submodular relaxation tiberio s. caetano nicta/anu/university of sydney canberra and sydney, australia.

[10] Marcelo Fiori, Pablo Musé, and Guillermo Sapiro. Topology constraints in graphical models. In *NIPS*, pages 800–808, 2012.

[11] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[12] Prem Gopalan, Sean Gerrish, Michael Freedman, David M Blei, and David M Mimno. Scalable inference of overlapping communities. In *Advances in Neural Information Processing Systems*, pages 2258–2266, 2012.

[13] Prem K Gopalan and David M Blei. Efficient discovery of overlapping communities in massive networks. *Proceedings of the National Academy of Sciences*, 110(36):14534–14539, 2013.

[14] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[15] Jie Liu and David Page. Bayesian estimation of latently-grouped parameters in undirected graphical models. In *Advances in Neural Information Processing Systems*, pages 1232–1240, 2013.

[16] Qiang Liu and Alexander T Ihler. Learning scale free networks by reweighted l1 regularization. In *International Conference on Artificial Intelligence and Statistics*, pages 40–48, 2011.

[17] Benjamin M Marlin and Kevin P Murphy. Sparse gaussian graphical models with unknown block structure. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 705–712. ACM, 2009.

[18] Benjamin M Marlin, Mark Schmidt, and Kevin P Murphy. Group sparse priors for covariance estimation. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 383–392. AUAI Press, 2009.

[19] Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, pages 1436–1462, 2006.

[20] Go Nakajima, Kazuhiko Hayashi, Yaguang Xi, Kenji Kudo, Kazumi Uchida, Ken Takasaki, Masakazu Yamamoto, and Jingfang Ju. Non-coding micrornas hsa-let-7g and hsa-mir-181b are associated with chemoresponse to s-1 in colon cancer. *Cancer Genomics-Proteomics*, 3(5):317–324, 2006.

[21] Cancer Genome Atlas Network et al. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012.

[22] Konstantina Palla, David A Knowles, and Zoubin Ghahramani. A nonparametric variable clustering model. In *Advances in Neural Information Processing Systems*, pages 2996–3004, 2012.

[23] Chiara Pastrello, Jerry Polesel, Lara Della Puppa, Alessandra Viel, and Roberta Maestro. Association between hsa-mir-146a genotype and tumor age-of-onset in brca1/brca2-negative familial breast and ovarian cancer patients. *Carcinogenesis*, 31(12):2124–2126, 2010.

[24] Pradeep Ravikumar, Martin J Wainwright, John D Lafferty, et al. High-dimensional ising model selection using 1-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010.

[25] Roded Sharan, Igor Ulitsky, and Ron Shamir. Network-based prediction of protein function. *Molecular systems biology*, 3(1), 2007.

[26] Siqi Sun, Yuancheng Zhu, and Jinbo Xu. Adaptive variable clustering in gaussian graphical models. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, pages 931–939, 2014.

[27] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[28] Scott White and Padhraic Smyth. A spectral clustering approach to finding communities in graph. In *SDM*, volume 5, pages 76–84. SIAM, 2005.

[29] Eunho Yang, Pradeep D Ravikumar, Genevera I Allen, and Zhandong Liu. Graphical models via generalized linear models. In *NIPS*, volume 25, pages 1367–1375, 2012.

[30] Ming Yuan and Yi Lin. Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35, 2007.