Notes on Stochastic Block Models

CMSC828O

As a reminder, we are fitting a model where the probability of the presence of edge $i\, j$ between nodes i and j in a network is based on class membership of i in class q and j in class r:

$$\log \frac{P(Y_{ij}=1|Y_{(ij)}=y_{(ij)})}{P(Y_{ij}=0|Y_{(ij)}=y_{(ij)})} = \theta_{qr}$$

The parameters to estimate are θ_{qr} for every pair of classes q and r. We will discuss two methods to estimate these parameters (a) maximizing likelihood using variational EM (https://arxiv.org/pdf/1011.1813.pdf) and (b) sampling using MCMC (https://arxiv.org/pdf/1310.4378.pdf).

Both cases are based on rewriting the model above in terms of latent (unobserved) class assignment variables. For instance, $z_{iq}=1$ if vertex i is assigned to class q and 0 otherwise. The number of classes Q is assumed given. With that we can write the likelihood of a set of parameters $\theta=\{\theta_{qr}\}$ given observed adjacency matrix y and assignments z as

$$\mathcal{L}(\theta,\alpha;y,z) = \sum_i \sum_q z_{iq} \log \alpha_q + \frac{1}{2} \sum_{i \neq j} \sum_{q,r} z_{iq} z_{jr} b(y_{ij};\theta_{qr})$$

with $b(y_{ij};\theta_{qr}) = y_{ij}\log\theta_{qr} + (1-y_{ij})\log(1-\theta_{qr})$. Values α_q are prior (do not depend on specific nodes) probabilities of assignment to class q. We will estimate these as well.

In EM we maximize this likelihood iteratively plugging in $E[z_{iq}|y,\theta,\alpha] = P(z_{iq}=1|y,\theta,\alpha)$, in MCMC we sample assignments z_{iq} from $P(z_{iq}=1|y,\theta,\alpha)$.

Variational EM

The standard EM

The usual EM algorithm takes the following steps:

1. Initialize parameters θ, α

2. Repeat until convergence:

a. E-step: Compute $\gamma_{iq} = E[z_{iq}|y,\theta,\alpha] = P(z_{iq}=1|y,\theta,\alpha)$

b. M-step: Estimate θ, α by maximizing $\mathcal{L}(\theta,\alpha;y,\gamma)$.

The solution to the M-step has a closed form:

$$\alpha_q = \frac{1}{n}\sum_i \gamma_{iq}$$

with n the number of nodes in the graph.

$$\theta_{qr} = \frac{\sum_{i \neq j} \gamma_{iq}\gamma_{jr}}{\sum_{s,t}\sum_{u \neq v} \gamma_{us}\gamma_{vt}}$$

The E-step presents an issue. To compute $P(z_{iq} = 1|y, \theta, \alpha)$ we would employ Bayes' rule:

$$P(z_{iq} = 1|y, \theta, \alpha)) = \frac{P(Y = y|z_{iq} = 1, \theta, \alpha)}{P(Y = y|z_{iq} = 1, \theta, \alpha) + P(Y = y|z_{iq} = 0, \theta, \alpha)}$$

However, the probability model $P(Y = y|z_{iq} = 1, \theta, \alpha)$ induces a dependence between settings of z. This means that in order to perform the E-step we need to reason about the joint distribution $P(z|y, \theta, \alpha)$ which leads to an inefficient algorithm. Here is where the variational trick comes in.

The variational trick

We introduce new parameters τ_{iq} to define a probability distribution $R_Y(z|\tau) = \prod_i h(z_i, \tau_i)$ with h a Multinomial distribution over classes with parameters $\tau_i = [\tau_{i1}, \ldots, \tau_{iQ}]$ used to approximate $P(z|y, \theta, \alpha)$.

Iterations are now to find estimates τ to those that make $R_Y(z|\tau)$ best approximate $P(z|y, \theta, \alpha)$ and then estimate θ and α as before, but now using parameters τ.

Finding the optimal τ does not have a closed-form, but you can show that the optimal τ satisfy this fixed-point relation

$$\tau_{iq} \propto \alpha_q \prod_{j \neq i} \prod_{l} \left[\exp\{b(y_i j, \theta_{ql})\} \exp\{b(y_{ij}, \theta_{lq})\}\right]^{\tau_{jl}}$$

Based on this observation, the E-step is replaced by iterating over the fixed point relation until convergence to obtain τ.

The M-step estimates are obtained by the same equations as above, replacing γ_{iq} with τ_{iq}.

Checking convergence

Convergence can be determined by checking convergence of $\mathcal{L}(\theta, \alpha; y, \tilde{z})$ where $\tilde{z}$ is the prediction of z given by the current model. In Variational EM this would be given by the highest probability assignment determined by τ.

Selecting the number of classes

To select the number of classes Q, you can use the ICL criterion by selecting the value Q that maximizes

$$\ell(\theta, \alpha; y, \tilde{z}) - \frac{1}{2}\left\{\frac{Q(Q+1)}{2} \log\left[n(n-1)\right] - (Q-1)\log(n)\right\}$$

where $\ell(\theta, \alpha; y, \tilde{z})$ is the value of $\mathcal{L}(\theta, \alpha; y, \tilde{z})$ after convergence.

MCMC

In MCMC, instead of maximizing likelihood we sample from the probability model we have just defined, and derive estimates empirically from the samples we obtain. We operate on an equivalent formulation of the problem.

First, we ignore prior probabilities α_q. Second, we note that the probability of graphs is strictly determined by edge counts $e_{qr} = \sum_{i \neq j} z_{iq} z_{jr}$ and $n_q = \sum_i z_{iq}$. These two points lead to the observation that $P(Y|z) = 1/\Theta(\{e_{qr}\}, \{n_q\})$ with $S(\{e_{qr}\}, \{n_r\}) = \log \Theta(\{e_{qr}\}, \{n_q\})$ given by

$$S(\{e_{qr}\}, \{n_r\}) = \frac{1}{2} \sum_{q,r} n_q n_r H_z \left(\frac{e_{qr}}{n_q n_r} \right)$$

with $H_z(x) = -x \log x - (1-x) \log x$. This suggests that instead of maximizing $\mathcal{L}$ as we did before, we get our model from minimizing S (this is the entropy of the same distribution btw). We will use sampling in such a way that samples are accepted so they improve S.

The sampling procedure

The general algorithm is as follows (with parameters $\epsilon > 0$ and $\beta > 0$):

First, randomly sample assignment q for each node i.

Then iterate K times (after enough iterations to mix):

a. For each node i move assignment q to r as follows:

(a.i) randomly select neighbor j of i and denote the current assignment of j as t

(a.ii) choose r uniformly at random (from $1, \ldots, Q$), accept with probability $R_t = \epsilon Q/(e_t + \epsilon Q)$, with e_t the total number of edges involving nodes in class t

(a.iii) if r is rejected, choose any edge $u \sim v$ with node u in class t, and set r to the class of node v

b. Decide if you accept move $q \to r$ as follows

(b.i) compute $x^i_{q \to r} = \sum_t p^i_t p(q \to r|t)$ where p^i_t is the fraction of neighbors of node i in class t and

$$p(q \to r|t) = \frac{e_{tr} + \epsilon}{e_t + \epsilon Q} = (1 - R_t) e_{tr}/e_t + R_t/Q$$

with R_t as before with values e_{tr} and e_t computed **after the move**

(b.ii) compute $y^i_{q \to r} = \sum_t p^i_t p(r \to q|t)$ with values e_{tq} and e_t computed **before the move**

(b.iii) compute $\delta^i_{q \to r} = S(\{e^{\text{new}}_{qr}\}, \{n^{\text{new}}_q\}) - S(\{e^{\text{old}}_{qr}\}, \{n^{\text{old}}_q\})$ where e^{new}_{qr} and n^{new}_q are computed **after the move** and e^{old}_{qr} and n^{old}_q are computed **before the move**. Notice that only terms involving the classes for i and its neighbors change so computing δ involves only computing the difference for those terms

(b.iv) accept moving node i from $q \to r$ with probability

$$\min \left\{ e^{-\beta \delta^i_{q \to r}} \times \frac{x^i_{q \to r}}{y^i_{q \to r}}, 1 \right\}$$

Deriving estimates

We can derive estimates for θ_{qr} and γ_q from our samples as follows:

- After each iteration k over the nodes of the graph, we can estimate $\theta^k_{qr} = e^k_{qr}/(n^k_q n^k_r)$ where e^k and n^k are calculated after all nodes have been (potentially) moved. To get an estimate of θ_{qr} we use the average of the θ^k_{qr}.
- Likewise, after each iteration k we produce an assignment for node i. We set γ_{iq} to be the proportion of the K iterations in which node i was assigned to class q.

Model selection

Like Variational EM we can use the ICL criterion to determine the number of classes Q. In that case we use the sampling estimates to compute $\ell(\theta; y, \tilde{z})$