Message-Passing Algorithms for Network Analysis

Cristopher Moore (UNM / Santa Fe Institute)
joint work with
Lenka Zdeborová, Florent Krzakała, Aurelien Decelle (CNRS)
Xiaoran Yan, Yaojia Zhu (UNM)
Mark Newman (Michigan)
and a brief mention of joint work with
Varsha Dani (UNM)
Anna Olson (Chicago)
Functional communities

*assortative* communities: vertices connect with others of the same type

but food webs, word adjacency networks, even some social networks have a more general kind of structure

a *functional* community or “module” is a set of vertices that connect to the rest of the network in similar ways

how do we find them? modularity, spectral approaches...

statistical inference: choose a class of generative models, and find the one most likely to generate the data
The stochastic block model

Each vertex $i$ has a type $t_i \in \{1, \ldots, k\}$, with prior distribution $q_1, \ldots, q_k$

$k \times k$ matrix $p$

If $t_i = r$ and $t_j = s$, there is an edge $i \rightarrow j$ with probability $p_{rs}$

$p$ is not necessarily symmetric

We don’t assume that $p_{rr} > p_{rs}$

Given $G$, we want to infer the type assignment $t : V \rightarrow \{1, \ldots, k\}$ and the matrix $p$

How do we get off the ground?
The likelihood

the probability of $G$ given the types $t$ and parameters $\theta=(p,q)$ is

$$P(G \mid t, \theta) = \prod_{(i,j) \in E} p_{t_i, t_j} \prod_{(i,j) \notin E} (1 - p_{t_i, t_j})$$

so the probability of $t$ given $G$ is

$$P(t \mid G, \theta) = \frac{P(t \mid \theta) P(G \mid t, \theta)}{\sum_{t' \in \{1, \ldots, k\}^n} P(G \mid t', \theta)}$$

$$\propto \prod_{i \in V} q_{t_i} \prod_{(i,j) \in E} p_{t_i, t_j} \prod_{(i,j) \notin E} (1 - p_{t_i, t_j})$$

call this the Gibbs distribution on $t$. How do we maximize it, or sample from it?
Maximizing the likelihood

single-site heat-bath dynamics: choose a random vertex and update its type

if we like, we can jointly maximize $P(G|t,\theta)$ as a function of $t$ and $p$ by setting

$$p_{rs} = \frac{e_{rs}}{n_r n_s}, \quad q_r = \frac{n_r}{n}$$

this works reasonably well on small networks...
I record that I was born on a Friday
Maximizing the likelihood

single-site heat-bath dynamics: choose a random vertex and update its type
if we like, we can jointly maximize $P(G|t,p)$ as a function of $t$ and $p$ by setting

$$p_{rs} = \frac{e_{rs}}{n_r n_s}, \quad q_r = \frac{n_r}{n}$$

this works reasonably well on small networks... but it isn’t really what we want
the probability of $\theta$ given $G$ is a proportional to a partition function

$$P(G|\theta) = \sum_{t \in \{1,\ldots,k\}^n} P(G|t,\theta)$$

and $-\log P(G|\theta)$ is a free energy, not a ground state energy
Maximizing the free energy

A several-line derivation shows that

$$\nabla_\theta \log P(G | \theta) = \sum_t P(t | G, \theta) \nabla_\theta \log P(t, G | \theta).$$

Expectation-maximization (EM): given the current estimate $\hat{\theta}$, find the new $\theta$ that maximizes the expected log-likelihood

$$\sum_t P(t | G, \hat{\theta}) \log P(t, G | \theta)$$

Then set $\hat{\theta} = \theta$ and iterate

But how to compute this expectation?
Marginals

to do the maximization, we don’t need the entire Gibbs distribution

suppose we can estimate the one- and two-point marginals

\[ \mu^i_r = \Pr[t_i = r] , \quad \mu^{ij}_{rs} = \Pr[t_i = r \text{ and } t_j = s] \]

then the expected log-likelihood is maximized by \( \theta = (p, q) \) where

\[ q_r = \frac{\sum_i \mu^i_r}{n} , \quad p_{rs} = \frac{\sum_{(i,j) \in E} \mu^{ij}_{rs}}{\sum_{i,j} \mu^{ij}_{rs}} \]
Belief propagation (a.k.a. the cavity method)

each vertex $i$ sends a “message” to each of its neighbors $j$, giving $i$’s marginal distribution based on its other neighbors $k$

denote this message $\mu_{r \rightarrow j}^{i} = \text{estimate of } \Pr[t_i = r] \text{ if } j \text{ were absent}$

how do we update it?
Belief propagation (a.k.a. the cavity method)

\[ \mu_{i \rightarrow j}^{s \rightarrow j} = \frac{1}{Z_{i \rightarrow j}} q_s \prod_{k \neq j} \sum_{i \in E} \mu_{r \rightarrow i}^{k \rightarrow i} p_{rs} \times \prod_{r} \sum_{k \neq j} \mu_{r \rightarrow i}^{k \rightarrow i} (1 - p_{rs}) \]

BP on a complete graph — takes \( O(n^2) \) time to update

can simplify by assuming that \( \mu_{r \rightarrow i}^{k \rightarrow i} = \mu_{r}^{k} \) for all non-neighbors \( i \)

each vertex \( k \) applies an “external field” \( \sum_r \mu_{r}^{k} (1 - p_{rs}) \) to all vertices of type \( s \)
Belief propagation (a.k.a. the cavity method)

\[ \mu_{s \rightarrow j} = \frac{1}{Z_{i \rightarrow j}} q_s \prod_{k \neq j} \sum_{(i,k) \in E} \mu_{k \rightarrow i} p_{rs} \times \prod_{k \neq j} \sum_{(i,k) \notin E} \mu_{k \rightarrow i} \frac{(1 - p_{rs})}{\mu_r(1 - p_{rs})} \]

each update now takes \( O(n+m) \) time

update until the messages reach a fixed point
From expectation to maximization

after the messages $\mu_{r}^{k\rightarrow i}$ reach a fixed point,

the two-point BP marginals are

$$\mu_{rs}^{ij} \propto \mu_{r}^{i\rightarrow j} \mu_{s}^{j\rightarrow i} \times \begin{cases} p_{rs} & (i, j) \in E \\ 1 - p_{rs} & (i, j) \notin E \end{cases}$$

and we update $\theta=(\rho, q)$ to

$$q_{r} = \frac{\sum_{i} \mu_{r}^{i}}{n}, \quad p_{rs} = \frac{\sum_{(i, j) \in E} \mu_{st}^{ij}}{\sum_{i, j} \mu_{st}^{ij}}$$

EM: alternate expectation (through BP) and maximization to find $\theta$ and $\mu$
The Bethe free energy

Bayes’ rule implies

$$\log P(G | \theta) = \sum_t P(t | G, \theta) \ln P(G | t, \theta) - \sum_t P(t | G, \theta) \ln P(t | G, \theta)$$

physically, the free energy has an energy and entropy term, $F = U - TS$

the average energy $U = \mathbb{E}[-\log P(G | t, \theta)]$ is a function of the marginals $\mu_{rs}^{ij}$

to compute the entropy $S$ of the Gibbs distribution, we assume an approximate joint distribution based on the marginals for which we can compute $S$ exactly,

$$P_{\text{Bethe}}(t | G, \theta) = \frac{\prod_{ij} \mu_{t_i, t_j}^{ij}}{\prod_i (\mu_i^i)^{d_i-1}}$$

yields a surprisingly good approximation of $F$, even on finite graphs with loops; can compare with exact calculations and MCMC calorimetry
Performance on large synthetic networks

fix $\theta=(p,q)$ and take $n=10^5$ or so

choose type assignment $t$ randomly according to $q$

generate edges randomly according to $p$

run the algorithm — how well does it do? given $\theta$, does it find the right $t$?
can it find $\theta$ using EM?

given the marginals $\mu^{k\rightarrow i}$, guess that $t_i = \arg\max_r \mu^{k\rightarrow i}_r$

note: not the ground state!

define the overlap as the fraction of vertices labeled correctly...

...minus the size of the largest group, and normalized
Sparse benchmarks

set $q_i = 1/k$ for all $i$

let $p_{ij} = c_{ij}/n$ where $c_{ij} = c_{in}$ if $i=j$ and $c_{out}$ if $i \neq j$

vary the ratio

$$\varepsilon = \frac{c_{out}}{c_{in}}$$

while keeping the average degree $c = c_{in}/k + (1-1/k)c_{out}$ fixed

if $\varepsilon$ is too close to 1, BP converges to the uniform fixed point

$$\mu^i_{r \rightarrow j} = \frac{1}{k}$$
A phase transition from detectable to undetectable communities

\[ \text{overlap} = \frac{c_{\text{out}}}{c_{\text{in}}} \]

Fig. 1 represents two examples where the overlap is positive and the factorized fixed point is not the optimal assignment for each group.

Fig. 2 shows the results obtained by the cavity method and the overlap computed with MCMC for size \( N=100k \), \( N=70k \), and \( N=128 \) graphs. In particular, Fig. 1(b) shows the case of \( q=4 \) and \( c=16 \). We will first study the result given by the cavity method in the thermodynamic limit in the case when the parameters belong to the Erdős-Rényi random graph, and vary the ratio \( \epsilon = c_{\text{out}}/c_{\text{in}} \) from an almost perfect separation of groups to an undetectable one based purely on the structure of the graph. For two groups of the same size, average degree 128 results from the full BP (22) and MCMC for smaller graphs with \( N=500k \), BP \( N=70k \), MC, and \( N=128 \), full BP with \( q=4 \), and \( c=16 \).
This gives the following stability criterion, so let us first investigate the influence of the perturbation of the same behavior, but BP obtains the marginals much more quickly.

When we observe in (40) that the matrix $\lambda$, the factorized fixed point is then unstable, and communities are easily detectable. On the other hand, if the stability condition (43) is known in the literature on sparse reconstruction, or robust reconstruction in information science as the Kesten-Stigum condition [39], for census reconstruction [25], or robust reconstruction to the other hand, if the variance, however, we have

The equilibrium time of Gibbs sampling (MCMC) has quality

Now let us consider the influence from all the leaves. The mean value of the perturbation on the leaf $\epsilon$ can then be written as

For $c=16$ and $q=4$, the convergence time diverges at the critical point $\epsilon_c$, and communities are easily detectable.

The transfer matrix $T$ does not depend on the index $q$'s largest eigenvalue $\lambda$.

This includes the non-factorized fixed point, which is stable. On the leaf $\epsilon$ leaves as $\epsilon$ moves up the tree.

The equation time $T$ is a constant number of iterations.
The free energy landscape
Which kind of community do you want?
Which kind of community do you want?
Similarly for parameter space, very close to those in (50): with such a small equilibration time, that MCMC is essential.

For comparison, we also performed learning using MCMC for the "leaders" on the one hand and "students/followers" on the other group. Of course, this second division is not wrong; rather, it highlights the tendency of the high-degree nodes in one group, including both the president and the director, to form a more cohesive block. Depending on the initial parameters, we can expect that the BP algorithm may converge to a fixed point that is nearly the same as the (in this case exact) MCMC. This suggests that our BP learning algorithm is useful and robust.
Degree-corrected block models

the “vanilla” block model expects vertices of the same type to have roughly the same degree

a random multigraph [Karrer & Newman, 2010]

each vertex $i$ has an expected degree $d_i$

analogous to $p_{ij}$, a $k \times k$ matrix $w_{ij}$

for each pair $i, j$ with $t_i=r$ and $t_j=s$, the number of edges between them is

$$m_{ij} = \text{Poi}(d_i d_j w_{rs})$$

now the degrees are parameters, not data to be explained

can again write down the BP/EM algorithm
Blogs: vanilla block model

In Fig. 1 and 2, we compare the results of applying two block models with different priors. The uncorrected model (a) incorrectly splits the network into two groups, one of high-degree vertices and another of low. The degree-corrected model (b), on the other hand, accurately assigns the vertices to the known communities, except for one vertex that is misidentified on the boundary of the two groups (as well as by other commonly used algorithms).

The failure of the uncorrected model in this context is because it does not take the degree sequence into account. The a priori probability of an edge between two vertices varies as the product of their degrees, a variation that can be fit by the uncorrected blockmodel if we divide the network into high- and low-degree groups. However, with only one set of groups to assign, we must choose between this fit and the true community structure. In this case, the division into high and low degrees gives the higher likelihood, so the algorithm returns this division. In contrast, the degree-corrected blockmodel already includes the variation of edge probability with degree in its likelihood, freeing up the block structure for better fitting to the communities.

For $K=3$, the ordinary stochastic blockmodel will, for sufficiently heterogeneous degrees, be biased towards splitting into three groups by degree—high, medium, and low—and similarly for higher values of $K$. It is possible that the true community structure corresponds entirely or mainly to groups of high and low degree, but we want our model to find this structure if it is statistically surprising once we know about the degree sequence, and this is precisely what the corrected model does.

As a second real-world example, we show in Fig. 2 an application to a network of political blogs assembled by Adamic and Glance (2010). The network is composed of blogs about US politics and the web links between them, captured on a single day in 2005. The blogs have known political leanings and were labeled as either liberal or conservative. We consider the network in undirected form and examine only the largest connected component, which has 1222 vertices. Figure 2 shows that, as with the karate club, the uncorrected model splits the vertices into high- and low-degree groups, while the degree-corrected model finds a split more aligned with the political division of the network. While not matching the known labeling exactly, the split generated by the degree-corrected model has a normalized mutual information of 0.72 with the labeling of Adamic and Glance, compared with 0.0001 for the uncorrected model.

As a check, we also verified that the group assignments found by the heuristic have a higher objective score than the known group assignments, and that using the known assignments as the initial condition for the optimization recovers the same group assignments as found with random initial conditions.

B. Generation of synthetic networks

We turn now to synthetic networks. The networks we use are generated from the degree-corrected model with parameters $K=2$. See Karrer & Newman (2010) for a detailed description of these models.
Blogs: degree-corrected block model

Fig. 2 shows that, as with the karate club, the uncorrected stochastic blockmodel splits the vertices into high- and low-degree groups, while the degree-corrected model finds a split more aligned with the political division of the network. While not matching the known labeling exactly, the split generated by the degree-corrected model has a normalized mutual information of 0.72 with the labeling of Adamic and Glance, compared with 0.0001 for the uncorrected model.

(Karrer & Newman, 2010)
Strengths and weaknesses

degree-corrected models don’t mind inhomogeneous degree distributions
but they also can’t use the degrees to help them label the nodes
on some networks, they perform worse than the vanilla model

yet another model: first generate vertex degrees $d_i$ according to some
distribution whose parameters depend on $t_i$ (e.g. power law)
then generate edges according to the degree-corrected model

for some networks (e.g. large word networks) works better than either vanilla or
degree-corrected model
Degree-generated model

out–degree distribution (Brown words network)

- blue line: adjective
- red line: noun

$p(d)$

degree (d)
Degree-generated model

in-degree distribution (Brown words network)

$p(d)$

degree (d)
Colorings with permutations

Threshold conjecture for $k$-colorability:

$$\lim_{n \to \infty} \Pr[G(n, p = d/n) \text{ is } k\text{-colorable}] = \begin{cases} 1 & \text{if } d < d_k \\ 0 & \text{if } d > d_k \end{cases}$$

Achlioptas and Naor determined $d_k$ to within $O(\log k)$ (and determined $k$ as a function of $d$ to two integers)

Conjecture: the threshold $d_k$ stays the same if we put a random permutation $\pi \in S_k$ on each edge, and demand that $c(u) \neq \pi(c(v))$ instead of $c(u) \neq c(v)$

Justification: correlation decay, reconstruction, survey propagation

Second moment calculations are much easier, letting us bound $d_c$ within an additive constant [Dani, Moore, Olson]: for any $\varepsilon$ and sufficiently large $k$,

$$2k \ln k - \ln k - 2 - \varepsilon \leq d_k \leq 2k \ln k - \ln k - 1 + \varepsilon$$
Shameless Plug

This book rocks! You somehow manage to combine the fun of a popular book with the intellectual heft of a textbook.
— Scott Aaronson

A treasure trove of information on algorithms and complexity, presented in the most delightful way.
— Vijay Vazirani

A creative, insightful, and accessible introduction to the theory of computing, written with a keen eye toward the frontiers of the field and a vivid enthusiasm for the subject matter.
— Jon Kleinberg

Oxford University Press, 2011
Acknowledgments

and the McDonnell Foundation