Fitting Social Network Models Using the Varying Truncation Stochastic Approximation MCMC Algorithm

May. 17, 2012

1This talk is based on a joint work with Dr. Ick Hoon Jin
Abstract

The exponential random graph model (ERGM) plays a major role for social network analysis. However, parameter estimation for the ERGM is a hard problem due to the intractability of its normalizing constant and the model degeneracy. The existing algorithms, such as MCMLE and stochastic approximation, often fail for this problem in the presence of model degeneracy. In this talk, we introduce the varying truncation stochastic approximation Markov chain Monte Carlo (SAMCMC) algorithm to tackle this problem. The varying truncation mechanism enables the algorithm to choose an appropriate starting point and an appropriate gain factor sequence and thus to produce a reasonable parameter estimate for the ERGM even in the presence of model degeneracy. The numerical results indicate that the varying truncation SAMCMC algorithm can significantly outperform the existing algorithms: For degenerate ERGMs, MCMLE and stochastic approximation often fail to produce any reasonable parameter estimates, while SAMCMC can do; for non-degenerate ERGMs, SAMCMC can work as well as or better than MCMLE and stochastic approximation.
Social Network

- The social network is a social structure made of actors (individuals, organizations, etc.), which are interconnected by certain relationship, such as friendship, common interest, financial exchange, etc.

- The network can be represented in a graph with a node for each actor and an edge for each relation between a pair of actors. This graph representation can provide insight into organizational structures, social behavior patterns, and a variety of other social phenomena.

- The technique of social network analysis has been applied to many other fields, such as biology, politics, and Internet studies (e.g., Facebook).
Examples of Social Networks

- **Florentine business network**: a set of business ties, such as loans, credits and joint partnerships, among 16 families who were locked in a struggle for political control of the city of Florence around 1430.

- **Kapferer tailor shop network**: worker interactions in a tailor shop in Zambia.

- **Lazega lawyer network**: It comes from a network study of corporate law partnership that was carried out in a Northeastern US corporate law firm, referred to as SG & R 1988-1991 in New England. The network represents friendships among 36 partners of this firm.
Examples of Social Networks

- **Karate network:** friendships among 34 members of a karate club at a US university in 1970s.
- **High school student friendship network:** It comes from a survey of high school students friendship.
Examples of Social Networks

Figure: Visualization of example networks: (a) Florentine business network; (b) Kapferer tailor shop network; (c) Lazega lawyer network; (d) Zachary’s Karate network.
Models of Social Networks

- **Bernoulli random graph model**: edges are considered independent of each other.
- **$p_1$ model**: dyads (a pair of nodes) are assumed independent.
- **Markov random graph model**: Each pair of edges is conditionally independent given the rest of the graph.
- **Exponential random graph model**: A generalization of the Markov random graph model, and it provides a flexible way to model the complex dependence structure of network graphs.
The likelihood function of the model is given by

\[ f(y|\theta) = \frac{1}{\kappa(\theta)} \exp \left\{ \sum_{i \in A} \theta_i S_i(y) \right\}, \]  

(1)

where \( S_i(y) \) denotes a statistic, \( \theta_i \) is the corresponding parameter, \( A \) specifies the set of statistics considered in the model, and \( \kappa(\theta) \) is the normalizing constant.

First Difficulty: \( \kappa(\theta) \) is intractable!
ERGM: Basic Markov Statistics

- **edge count**: count of edges.
- **k2-star**: count of 2-stars (if one node connects to other two nodes, it is called a 2-star).
- **k3-star**: count of 3-stars (if one node connects to other three nodes, it is called a 3-star).
- **triangle count**: count of triangles (If node 'a' connects to node 'b', node 'b' connects to node 'c', and node 'c' connects to node 'a' simultaneously, then the nodes 'a', 'b' and 'c' form a triangle).
Let $D_i(y)$ denote the number of nodes with degree $i$. The geometrically weighted degree (GWD) statistic is defined by

$$u(y|\tau) = e^\tau \sum_{i=1}^{n-2} \left\{ 1 - \left( 1 - e^{-\tau} \right)^i \right\} D_i(y),$$

where the additional parameter $\tau$ specifies the decreasing rate of the weights put on the higher order terms.
ERGM: Shared Partnership

- $EP_k(y)$: the number of unordered pairs $(i, j)$ for which $i$ and $j$ have exactly $k$ common neighbors and $Y_{ij} = 1$.
- $DP_k(y)$: the number of unordered pairs $(i, j)$ for which $i$ and $j$ have exactly $k$ common neighbors regardless of the value of $Y_{ij}$.

The geometrically weighted edgewise shared partnership (GWESP) statistic and geometrically weighted dyadwise shared partnership (GWDSP) statistic are defined, respectively, by

\[
\begin{align*}
\nu(y|\tau) &= e^{\tau} \sum_{i=1}^{n-2} \left\{ 1 - \left( 1 - e^{-\tau} \right)^i \right\} EP_i(y), \\
\omega(y|\tau) &= e^{\tau} \sum_{i=1}^{n-2} \left\{ 1 - \left( 1 - e^{-\tau} \right)^i \right\} DP_i(y).
\end{align*}
\]
ERGMs suffer from two difficulties in parameter estimation:

- Intractability of the normalizing constant
- Model degeneracy
Intractability of the normalizing constant

- Monte Carlo maximum likelihood estimation (MCMLE) (Geyer and Thompson, 1992)
- Stochastic approximation (Snijder, 2002)
The basic idea is to approximate the normalizing constant $\kappa(\theta)$ using Monte Carlo samples. Let $\theta^{(0)}$ denote an initial estimate of $\theta$, and let $z_1, \ldots, z_m$ denote a set of auxiliary social networks simulated from $f(z|\theta^{(0)})$ using MCMC. Then

$$\log f_m(y_{obs}|\theta) = \sum_{a \in A} \theta_a s_a(y_{obs}) - \log \left( \frac{1}{m} \sum_{i=1}^{m} \exp \left\{ \sum_{a \in A} \theta_a S_a(z_i) - \sum_{a \in A} \theta_a^{(0)} S_a(z_i) \right\} \right),$$

converges to $\log f(y_{obs}|\theta)$ (up to a constant) as $m \to \infty$, where $y_{obs}$ denotes the observed social network.

It is known that the performance of this method depends on the choice of $\theta^{(0)}$. If $\theta^{(0)}$ is near the MLE, it can produce a good estimate of $\theta$. 
Stochastic Approximation

The theory of exponential family implies that maximizing the likelihood function (1) is equivalent to solving the system of equations

\[ E_\theta (S(Y)) = S(y_{obs}). \] (6)

A major shortcoming of this method (described below) is its inefficiency in generating independent network samples. The number of iteration steps for generating each sample \( y_{k+1} \) is in the order of \( 100n^2 \), where \( n \) is the number of nodes of the network.
Stochastic Approximation (continuation)

(a) (Independence network generation) Generate an independent sample \( y^{(k+1)} \) from the distribution \( f(y|\theta^{(k)}) \): Starting with a random graph in which each arc variable \( Y_{ij} \) is determined independently with a probability 0.5 for the values 0 and 1; and then updating the random graph using the Gibbs sampler or the MH algorithm.

(b) (Parameter estimate update) Set

\[
\theta^{(k+1)} = \theta^{(k)} - a_k D^{-1} (U(y_{k+1}, \tilde{y}_{k+1}) - S(y_{obs})),
\]

where \( \{a_k\} \) denotes a positive sequence converging to 0, \( D \) denotes a pre-estimated covariance matrix of \( S(Y) \) at the initial estimate \( \theta^{(1)} \), \( \tilde{y}_{k+1} = 1 - y_{k+1} \) denotes the complementary network of \( y_{k+1} \) (with each cell of the adjacency matrix of \( y_{k+1} \) being switched from 0 to 1 and vice versa),

\[
U(y_{k+1}, \tilde{y}_{k+1}) = P(\tilde{y}_{k+1}|y_{k+1})S(\tilde{y}_{k+1}) + (1 - P(\tilde{y}_{k+1}|y_{k+1}))S(y_{k+1}),
\]

and \( P(\tilde{y}_{k+1}|y_{k+1}) \) denotes the MH acceptance probability of the transition from \( y_{k+1} \) to \( \tilde{y}_{k+1} \).
Model Degeneracy

It refers to the phenomenon that for some configurations of $\theta$, ERGM lumps all or most of its probability mass on just one or a few possible graphs. In most cases of degeneracy, disproportionate probability mass is placed either on the complete (fully connected) or empty (entirely unconnected) networks.

The models with basic Markov statistics often suffer from the model degeneracy problem. When the observed network is fitted by such a model, MCMLE and SAA may produce a degenerate estimate of $\theta$ (i.e., the estimate falls in a degeneracy region) if the starting value is in or close to a degeneracy region. In this case, the resulting model will not provide a good fitting to the network.
Figure: Visualization of the degeneracy (black) and non-degeneracy (white) regions of an ERGM. The cycle (o), plus (+) and triangle (Δ) indicate the estimates produced by varying truncation SAMCMC, SAA and MCMLE, respectively.
Basic setup

(C1) Set

\[ a_k = C_a \left( \frac{k_0}{\max(k_0, k)} \right)^\eta, \quad b_k = C_b \left( \frac{k_0}{\max(k_0, k)} \right)^\xi, \quad (8) \]

for some constants \( k_0 > 1, \eta \in (1/2, 1), \xi \in (1/2, \eta), C_a > 0, \) and \( C_b > 0. \)

(C2) Let \( \{\mathcal{K}_s, s \geq 0\} \) denote a sequence of compact sets of \( \Theta, \)

\[ \bigcup_{s \geq 0} \mathcal{K}_s = \Theta, \quad \text{and} \quad \mathcal{K}_s \subset \text{int}(\mathcal{K}_{s+1}), \quad (9) \]

where \( \text{int}(A) \) denotes the interior of set \( A. \) In addition, there exist constants \( l_0 \) and \( l_1 \) such that \( l_0 > l_1, \mathcal{K}_0 \subset \{\theta \in \Theta : l(\theta|\mathbf{y}_{\text{obs}}) > l_0\}, \) and \( \{\theta \in \Theta : l(\theta|\mathbf{y}_{\text{obs}}) \geq l_1\} \) is compact, where \( l(\theta|\mathbf{y}_{\text{obs}}) \) denotes the log-likelihood function of the model under consideration.
Basic setup (continuation)

- Let $\mathcal{X}$ denote a space of social networks. Let $\mathcal{X}_0$ denote a subset of $\mathcal{X}$, and let $T : \mathcal{X} \times \Theta \rightarrow \mathcal{X}_0 \times \mathcal{K}_0$ be a measurable function which maps a point $(y, \theta)$ in $\mathcal{X} \times \Theta$ to a random point in the initial region $\mathcal{X}_0 \times \mathcal{K}_0$; that is, $T$ defines a re-initialization mechanism, re-initializing the simulation in the set $\mathcal{X}_0 \times \mathcal{K}_0$. For ERGMs, the sample space $\mathcal{X}$ is finite, we set $\mathcal{X}_0 = \mathcal{X}$; that is, each run restarts or is re-initialized with a random configuration of the network.

- Let $\sigma_k$ denote the number of re-initializations performed until iteration $k$, and $\sigma_0 = 0$. 
SAMCMC Algorithm

(a) Draw an auxiliary social network $y_{k+1}$ from the distribution $f(y|\theta^{(k)})$ using the Gibbs sampler, starting with the network $y_k$ and iterating for $m$ sweeps.

(b) Set $\theta^{(k+\frac{1}{2})} = \theta^{(k)} + a_k \left( S(y_{k+1}) - S(y_{obs}) \right)$.

(c) If $||\theta^{(k+\frac{1}{2})} - \theta^{(k)}|| \leq b_k$ and $\theta^{(k+\frac{1}{2})} \in \mathcal{K}_{\sigma_k}$, where $||z||$ denotes the Euclidean norm of the vector $z$, then set $\sigma_{k+1} = \sigma_k$ and $(y_{k+1}, \theta^{(k+1)}) = (y_{k+1}, \theta^{(k+\frac{1}{2})})$; otherwise set $\sigma_{k+1} = \sigma_k + 1$ and $(y_{k+1}, \theta^{(k+1)}) = \mathcal{T}(y_k, \theta^{(k)})$. 

Faming Liang
SAMCMC Algorithm: Trajectory averaging

The $\theta$ can be estimated by the trajectory averaging estimator

$$\bar{\theta}_n = \frac{1}{n} \sum_{k=1}^{n} \theta^{(k)}. \quad (10)$$

In practice, to reduce the variation of the estimate, one can use

$$\bar{\theta}(n_0, n) = \frac{1}{n - n_0} \sum_{k=n_0+1}^{n} \theta^{(k)}, \quad (11)$$

where $n_0$ denotes the number of burn-in iterations and it is usually set to a value at which the last truncation occurs.
Assume that the conditions \((C_1), (C_2)\) hold and the proposal satisfies the local positive condition (defined below). Then, as \(n \to \infty\),

1. **(Convergence)** \(\theta^{(n)} \to \theta^*\) almost surely, where \(\theta^*\) denotes a solution of equation (6).

2. **(Asymptotic Normality)**

\[
\sqrt{n}(\bar{\theta}_n - \theta^*) \to N(0, \Gamma),
\]

where \(\Gamma\) is a positive definite matrix independent of the sequences \(\{a_k\}\) and \(\{b_k\}\).

3. **(Asymptotic Efficiency)** \(\bar{\theta}_n\) is asymptotically efficient; that is, \(\Gamma\) is the smallest possible limit covariance matrix that an estimator based on a stochastic approximation algorithm can achieve.
SAMCMC Algorithm: Local positive condition

\((C_3)\) (Local positive) For every \(y \in \mathcal{X}\), there exist positive \(\epsilon_1\) and \(\epsilon_2\) such that

\[
\|z - y\| \leq \epsilon_1 \implies q(z|y) \geq \epsilon_2,
\]

(12)

where \(q(z|y)\) denotes the proposal distribution conditioned on the current sample \(y\).

The Gibbs sampler satisfies the condition \(C_3\) by noting that it is a special case of the MH algorithm and that only a single arc variable is updated at each step.
Florentine Business Network

Table: Parameter estimates for the Florentine business network, which are calculated by averaging over five independent runs with the standard deviations (Monte Carlo error) given in the parentheses.

<table>
<thead>
<tr>
<th>Method</th>
<th>Edge Count($\theta_1$)</th>
<th>K2-Star($\theta_2$)</th>
<th>CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAMCMC</td>
<td>-2.733 (4.2 × 10^{-4})</td>
<td>0.198 (9.0 × 10^{-5})</td>
<td>3.2s</td>
</tr>
<tr>
<td>MCMLE</td>
<td>-3.191 (2.6 × 10^{-1})</td>
<td>0.412 (1.2 × 10^{-1})</td>
<td>78.1s</td>
</tr>
<tr>
<td>SAA</td>
<td>-2.842 (7.7 × 10^{-3})</td>
<td>0.283 (3.9 × 10^{-2})</td>
<td>370.4s</td>
</tr>
</tbody>
</table>
Florentine Business Network: Goodness of fit

Figure: Row 1: SAMCMC; Row 2: MCMLE; Row 3: SAA. The solid line shows the observed network statistics, and the box-plots represent the distributions of simulated network statistics.
Florentine Business Network: Effect of $m$

Figure: Trajectories of $\theta$ produced by SAMCMC for the Florentine business network with different values of $m$: 1, 5 and 10.
Kapferer Tailor Shop Network

Table: Estimates of $\theta$ produced by SAMCMC, MCMLE and SAA for the Kapferer tailor shop network (based on 5 runs). * The MCMLE estimates are calculated based on 4 runs only, as it failed to produce an estimate in one run.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Edge Count</th>
<th>K2-star</th>
<th>GWESP</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAMCMC</td>
<td>-4.056</td>
<td>0.038</td>
<td>0.962</td>
<td>4.3m</td>
</tr>
<tr>
<td></td>
<td>(5.7 $\times$ 10$^{-3}$)</td>
<td>(2.4 $\times$ 10$^{-4}$)</td>
<td>(1.0 $\times$ 10$^{-3}$)</td>
<td></td>
</tr>
<tr>
<td>MCMLE*</td>
<td>-4.256</td>
<td>0.089</td>
<td>0.542</td>
<td>204.8m</td>
</tr>
<tr>
<td></td>
<td>(1.8 $\times$ 10$^{-5}$)</td>
<td>(1.2 $\times$ 10$^{-3}$)</td>
<td>(3.4 $\times$ 10$^{-5}$)</td>
<td></td>
</tr>
<tr>
<td>SAA</td>
<td>-3.927</td>
<td>0.068</td>
<td>0.571</td>
<td>99.9m</td>
</tr>
<tr>
<td></td>
<td>(3.8 $\times$ 10$^{-3}$)</td>
<td>(8.4 $\times$ 10$^{-5}$)</td>
<td>(5.7 $\times$ 10$^{-4}$)</td>
<td></td>
</tr>
</tbody>
</table>
Kapferer Tailor Shop Network: Goodness of fit

Figure: Row 1: SAMCMC; Row 2: MCMLE; Row 3: SAA.
Table: Estimates of $\theta$ produced by SAMCMC for Kapferer tailor Shop network, which are averaged over five independent runs with different starting points. The number in the parentheses represents the standard deviation (Monte Carlo error) of the estimate.

<table>
<thead>
<tr>
<th>Starting Point</th>
<th>Edge Count</th>
<th>K2-star</th>
<th>GWESP</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-20, 0, 17)</td>
<td>-4.015 (4.4e-3)</td>
<td>0.040 (6.9e-5)</td>
<td>0.921 (2.5e-3)</td>
</tr>
<tr>
<td>(-350, 0, 350)</td>
<td>-3.886 (3.9e-4)</td>
<td>0.028 (3.9e-5)</td>
<td>0.954 (3.5e-4)</td>
</tr>
</tbody>
</table>
### Lazega Lawyer Network

<table>
<thead>
<tr>
<th></th>
<th>SAMCMC</th>
<th>MCMLE</th>
<th>SAA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edge Counts</strong></td>
<td>-6.507(9.7e-4)</td>
<td>-6.442(7.1e-3)</td>
<td>-6.503(2.6e-2)</td>
</tr>
<tr>
<td><strong>Main Effect</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seniority</td>
<td>0.852(4.3e-4)</td>
<td>0.874(5.5e-3)</td>
<td>0.820(2.7e-2)</td>
</tr>
<tr>
<td>Practice</td>
<td>0.410(2.0e-4)</td>
<td>0.447(5.5e-3)</td>
<td>0.393(1.7e-2)</td>
</tr>
<tr>
<td><strong>Homophily Effect</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Practice</td>
<td>0.760(2.7e-4)</td>
<td>0.731(5.8e-3)</td>
<td>0.733(2.0e-2)</td>
</tr>
<tr>
<td>Sex</td>
<td>0.703(4.0e-4)</td>
<td>0.668(9.6e-3)</td>
<td>0.676(2.0e-2)</td>
</tr>
<tr>
<td>Location</td>
<td>1.145(2.8e-4)</td>
<td>1.168(9.7e-3)</td>
<td>1.111(2.8e-2)</td>
</tr>
<tr>
<td>GWESP</td>
<td>0.898(1.3e-4)</td>
<td>0.908(1.4e-2)</td>
<td>0.858(4.6e-2)</td>
</tr>
<tr>
<td><strong>time</strong></td>
<td>2.7m</td>
<td>8.0m</td>
<td>60.2m</td>
</tr>
</tbody>
</table>
Lazega Lawyer Network: Goodness of fit

Figure: Row 1: SAMCMC; Row 2: MCMLE; Row 3: SAA.
Zachary Karate Network

Table: Parameter estimates produced by SAMCMC, MCMLE and SAA for the Karate network, whose standard deviations (Monte Carlo error) are shown in the parentheses.

<table>
<thead>
<tr>
<th>Method</th>
<th>Edge Count</th>
<th>GWD</th>
<th>GWESP</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAMCMC</td>
<td>-3.730 (6.5e-4)</td>
<td>3.725 (5.0e-3)</td>
<td>1.303 (3.5e-4)</td>
<td>2.5m</td>
</tr>
<tr>
<td>MCMLE</td>
<td>-2.909 (5.3e-2)</td>
<td>7.901 (3.5e-3)</td>
<td>0.361 (7.7e-2)</td>
<td>2.9m</td>
</tr>
<tr>
<td>SAA</td>
<td>-3.637 (3.5e-2)</td>
<td>3.584 (5.1e-2)</td>
<td>1.224 (6.1e-2)</td>
<td>22.2m</td>
</tr>
</tbody>
</table>
Zachary Karate Network: Goodness of fit

Figure: Row 1: SAMCMC; Row 2: MCMLE; Row 3: SAA.
High school students friendship Network

Figure: A large network example: High school students friendship network.
High school students friendship Network: Goodness of fit

Faming Liang

Fitting Social Network Models Using the Varying Truncation SAMC MCMC
Conclusion

- Compared to MCMLE and SAA, SAMCMC can produce a reasonable parameter estimate for ERGMs even in presence of model degeneracy.
- The strength of SAMCMC comes from its varying truncation mechanism, which enables SAMCMC to jump out of the degeneracy region through re-initialization.
- MCMLE and SAA do not possess the re-initialization mechanism and tend to converge to a solution near the starting point, so they often fail to produce any reasonable parameter estimates for ERGMs in the presence of degeneracy.
The varying truncation mechanism looks like a random restarting mechanism, but they are different!

- For varying truncation, when the system is re-initialized, the gain factor sequence will start with a smaller value; that is, the solution will be searched in a more and more careful manner as the number of truncations increases.

- As seen from our plot, the MLE may lie at the boundary of the degeneracy and non-degeneracy regions. Even starting with a good point located in the non-degeneracy region and near the MLE, if the gain factor is too large, the system may still move into the degeneracy region. This explains why SAMCMC almost always needs to be re-initialized for a degenerate model. Sometimes, the re-initialization is not due to a bad starting point, but an inappropriate gain factor sequence. Therefore, a totally random restarting algorithm should not work better than SAMCMC.
Acknowledgment

- NSF
- KAUST