Bayesian inference of exponential random graph models under measurement errors

Nan Lin
Department of Mathematics
Washington University in St. Louis

Guanshengrui Hao
Department of Mathematics
Washington University in St. Louis

April 14, 2015

Abstract

While the impact of measurement errors inherent in network data has been widely recognized, relatively little work has been done to solve the problem mainly due to the complex dependence nature of network data. In this paper, we propose a Bayesian inference framework for summary statistics of the true underlying network, based on the network observed with measurement errors. To the best of our knowledge, this paper is the first to deal with measurement errors in the network data analysis in a Bayesian framework. We construct a Gibbs sampler to iteratively draw underlying true networks and model parameters. We incorporate the exponential random graph model to serve as the likelihood function, and use the exchange algorithm to draw model parameters. Simulation results show that using our inference framework, the impact of measurement errors has been reduced significantly.

Key Words: Bayesian inference; measurement errors; exponential random graph models; exchange algorithm.
1 Introduction

During the past decade, network data have emerged explosively across different fields including biology and bioinformatics, computer science, physics, statistics, sociology, economics, etc. [Alon (2006), Goldenberg et al. (2010), Jackson et al. (2008), Kolaczyk (2009), Newman (2004)]. In all such settings, the structural relationship among the data instances must be taken into account. As a result, increasing attention has been received on network data analysis, and network modeling is one of the most active areas of research.

Typically, networks are constructed through certain measurements on raw data. For instance, biological networks (e.g., the gene regulatory network) are often constructed based on certain associations among experimental measurements of activity levels. Noise is thus brought in through inherent measurement errors and it has significant impacts on the structure of the networks. Such impacts have been explored in many theoretical studies [Balachandran et al. (2013), Holland and Leinhardt (1973)] as well as simulation results [Fujita et al. (2009), Wang et al. (2012)].

While the impact of measurement errors is widely recognized [Balachandran et al. (2013), Fujita et al. (2009), Holland and Leinhardt (1973), Wang et al. (2012)], most network analyses in practice still proceed as if there were no errors. The reason for that is mostly because, there are relatively few formal probabilistic analyses for characterizing the propagation of errors and statistical inference under measurement errors, due to the complex dependence nature of network data [Balachandran et al. (2014)]. Holland and Leinhardt (1973) suggested to develop robust data analytic techniques to minimize the effects of measurement errors. Balachandran et al. (2013) proposed a nonparametric denoising approach to construct an estimator through spectral decomposition and showed that the estimator can reduce impact caused by measurement errors for a certain group of problems. But the estimator loses the nature of network structure and cannot be used to calculate summary statistics directly. Balachandran et al. (2014) showed that under certain assumptions, the distribution of discrepancy in summary statistics for networks with and without measurement errors can be approximated by a Skellam distribution. But in real cases, some of the assumptions are strong and relatively unrealistic.
As remarked above, studies aiming to reduce the impact of measurement errors remain scarce. Unlike previous works, which directly estimate the network or summary statistics, this paper proposes a Bayesian inference framework. We use the exponential random graph model (ERGM) \cite{Robins2007, Wasserman1996} to serve as the likelihood and construct a Gibbs sampler, which allows us to draw samples of true underlying networks and model parameters, thus obtain estimates of both summary statistics and model parameters. The organization of this paper is as follows. Section 2 introduces the ERGM model in detail and sets up assumptions on measurement errors. In Section 3, we will formulate the Bayesian framework to make inference for the summary statistics and model parameters under consideration of measurement errors. In Section 4, we will apply our framework onto simulated networks and realistic networks, respectively, and show how it reduces the impact of measurement errors. Finally, there will be further discussion in Section 5. Derivations of main results are presented in the Appendix.

2 Background

2.1 Representation of a network

Typically, a network consists of a set of vertices and a set of edges. While each vertex represents an actor of interest, each edge represents the relationship between the two vertices it connects. For example, in a co-authorship network, the set of vertices represents the scientists, and two scientists are connected by an edge if they have coauthored a paper \cite{Newman2004}. Here we only consider simple networks, i.e., networks that are undirected, unweighted and have no self loops, i.e., edge connected at both ends to the same vertex.

Assume that the network $G = (V, E)$ we are interested in is one realization of a random graph $\mathcal{G}$, where $V$ denotes the set of vertices and $E$ denotes the set of edges. And let $n = |V|$ be the number of vertices and $m = |E|$ be the number of edges. Denote the $n \times n$ adjacency matrix of the random graph $\mathcal{G}$ by $W$, and denote that of the realization network $G$ by $W$. $W_{ij} = 1$ if the dyad $(i, j)$ is connected by an edge, and $W_{ij} = 0$ otherwise. Since no self loop is allowed, $W_{ii} = 0$ for all $i = 1, \ldots, n$. 
2.2 Exponential Random Graph model

This paper uses the ERGM model as the likelihood function in the Bayesian inference framework. Given the parameters $\theta = \{\theta_1, \cdots, \theta_K\}^\top$, the ERGM models the log-likelihood that random graph $G$ takes a certain configuration as a linear combination of a few functions of its adjacency matrix $W$ together with a constant which depends on $\theta$, i.e.,

$$\log P(W = W|\theta) = \sum_{i=1}^K \theta_i s_i(W) - c(\theta) = \theta^\top s(W) - c(\theta). \quad (1)$$

Those functions $s_i(W)$ are summary statistics, describing certain network characteristics or nodal attributes of interest, e.g., number of edges, number of triangles, graph diameter, etc.. Let $W$ be the space of all $2^{(n^2)}$ possible networks, then

$$\exp c(\theta) = \sum_{W \in W} \exp \left[ \theta^\top s(W) \right]. \quad (2)$$

Denote $\exp c(\theta)$ by $z(\theta)$, then (1) is thus

$$P(W = W|\theta) = \frac{\exp \left[ \theta^\top s(W) \right]}{z(\theta)}. \quad (3)$$

The model parameter $\theta$ can be interpreted as the increase of log-odds for forming an edge between a dyad, conditional on all other dyads remaining unchanged. More specifically, let $W_{ij}^c = W_{ij}^c$ denote that all dyads except $(i,j)$ are realized, and denote $W_{ij}^+ = \{W_{ij}^c \cup \{W_{ij} = 1\}\}$, $W_{ij}^- = \{W_{ij}^c \cup \{W_{ij} = 0\}\}$. Then

$$\logit \left[ P(W_{ij} = 1|W_{ij}^c = W_{ij}^c, \theta) \right] = \theta^\top \left[ s(W_{ij}^+) - s(W_{ij}^-) \right], \quad (4)$$

where $s(W_{ij}^+) - s(W_{ij}^-)$ is the change of summary statistics when $W_{ij}$ change from 0 to 1.

2.3 Measurement errors

When measurement errors are considered, we denote the adjacency matrix of the true realization of the random graph $G$ by $W_{true}$, and denote the network
we observed, which contains measurement errors, by $W_{\text{obs}}$. We assume the error terms on $W_{\text{true}}$ are additive and independent across different dyads. Denote the measurement error instance by $W_{\text{noise}}$, i.e.

$$W_{\text{obs}} = W_{\text{true}} + W_{\text{noise}}.$$  \hfill (5)

We assume that $W_{\text{noise}}$ satisfies

- $-W_{\text{noise}}^{ij} \sim \text{Bernoulli}(p)$, if $W_{\text{true}} = 1$,
- $W_{\text{noise}}^{ij} \sim \text{Bernoulli}(q)$, if $W_{\text{true}} = 0$,
- $W_{\text{noise}}^{ij} \perp W_{\text{noise}}^{kl}$ if $(i, j) \neq (k, l)$ or $(l, k)$,

where $p$ and $q$ are assumed to be known parameters. If we consider that the construction of the network is based on hypothesis testing on each dyad $(i, j)$, $p$ can be interpreted as the probability of Type-II error (false negative) for the test on $(i, j) \in E$, and $q$ the probability of Type-I error (false positive) for the test on $(i, j) \in E^c$.

### 3 Bayesian inference framework

The estimation of our model parameters $\theta$ is based on not only the observed network $W_{\text{obs}}$, but also noise parameters $p$ and $q$. Consider the Bayesian treatment of the problem, where a prior distribution $\pi(\theta)$ is placed on $\theta$ and our interest is the posterior distribution,

$$f(\theta|W_{\text{obs}}) \propto f(W_{\text{obs}}|\theta)\pi(\theta) = \int f(W_{\text{obs}}|W_{\text{true}}, \theta)P(W_{\text{true}}|\theta)W_{\text{true}}.$$  \hfill (6)

To draw samples from the posterior distribution $f(\theta|W_{\text{obs}})$, we need to solve the integration in (6), which involves enumerating all possible $2^n$ configurations of $W_{\text{true}} \in \mathcal{W}$. It becomes intractable as the number of vertices $n$ gets larger. Therefore, we cannot directly draw samples from the posterior distribution $f(\theta|W_{\text{obs}})$ to obtain an estimate.

Alternatively, consider the augmented posterior distribution $f(\theta, W_{\text{true}}|W_{\text{obs}})$. Through a Gibbs sampler, we can draw samples $\theta^t$ and $W_{\text{true}}^t$ from $f(\theta, W_{\text{true}}|W_{\text{obs}})$ iteratively, thus obtain not only a marginalized estimate of $\theta$ but also an estimate of $s(W_{\text{true}})$ based on $W_{\text{true}}^t$. 


To construct the Gibbs sampler, we need to compute two full conditional distributions, \( f(W_{\text{true}}|W_{\text{obs}}, \theta) \) and \( f(\theta|W_{\text{obs}}, W_{\text{true}}) \). The first one, \( f(W_{\text{true}}|W_{\text{obs}}, \theta) \), is a mixture of \( \binom{n}{2} \) independent Bernoulli distributions and the ERGM model \( P(W_{\text{true}}|\theta) \), given by

\[
f(W_{\text{true}}|W_{\text{obs}}, \theta) \propto \exp \left[ \left( \theta_1 + \log \frac{1-p}{q} \right) \sum_{i,j}^{W_{ij}^{\text{obs}}=1} W_{ij}^{\text{true}} + \left( \theta_1 + \log \frac{p}{1-q} \right) \sum_{i,j}^{W_{ij}^{\text{obs}}=0} W_{ij}^{\text{true}} \right] \exp \left[ \tilde{\theta}^\top \tilde{s}(W_{\text{true}}) \right], \tag{7}
\]

where \( p \) and \( q \) are noise parameters introduced in Section 2.3 and \( \theta_1 \) is the model parameter for the number of edges. \( \tilde{\theta} \) and \( \tilde{s} \) are abbreviated model parameter vector and summary statistics vector, respectively. Detailed derivation is included in the Appendix. For the second full conditional distribution \( f(\theta|W_{\text{obs}}, W_{\text{true}}) \), notice that given the true realization \( W_{\text{true}} \), the model parameter \( \theta \) is independent with the observed network \( W_{\text{obs}} \). Therefore, it can be simplified as \( f(\theta|W_{\text{true}}) \). With a prior distribution \( \pi(\theta) \) placed on \( \theta \),

\[
f(\theta|W_{\text{true}}) \propto P(W_{\text{true}}|\theta)\pi(\theta). \tag{8}
\]

Our main algorithm contains following steps:

**Algorithm 1**

1. Initialize \( W_{\text{true}}^0 \) and \( \theta^0 \);
2. Iteratively, in the \( t \)-th step,
   a. Draw \( W_{\text{true}}^{t+1} \) from \( f(\cdot|W_{\text{obs}}, \theta^t) \);
   b. Draw \( \theta^{t+1} \) from \( f(\cdot|W_{\text{true}}^{t+1}) \);
3. Stop when the chain converges.

Our framework involves two issues: (1) how to draw a network from distribution (7) in Step 2a Algorithm 1, as illustrated in Section 3.1; and (2) how to draw model parameters in an ERGM model in Step 2b Algorithm 1, with details in Section 3.2.
3.1 Updating $W_{true}$

To draw $W_{true}^{t+1}$ from the full conditional distribution $f(\cdot|W_{obs}, \theta)$, we use a Markov chain Monte Carlo (MCMC) algorithm. One naive implementation is:

**Algorithm 1-1a**

1. Start with some arbitrary network $W^0$;
2. Iteratively, in the $k$-th step,
   a. Randomly pick a dyad $(i^k, j^k)$,
   b. Propose a move to a new network $W^*$ constructed by
      \[
      W^*_{ij} = \begin{cases} 
      1 - W^k_{ij}, & \text{if } (i, j) = (i^k, j^k) \text{ or } (j^k, i^k), \\
      W^k_{ij}, & \text{otherwise}.
      \end{cases}
      \]
   c. Calculate the acceptance ratio
      \[
      r(W^k, W^*) = \frac{f(W^* | W_{obs}, \theta)}{f(W^k | W_{obs}, \theta)}.
      \]
   d. Accept the proposal move to $W^*$ with probability
      \[
      a(W^k, W^*) = \min(1, r(W^k, W^*)�)
      \]
3. Stop when the chain converges.

In many realistic cases, networks are very sparse [Morris et al. (2008)]. When using the naive implementation, high proportion of the proposed moves in step 2d Algorithm 1-1a would be to add an edge rather than remove one. Such moves would more likely be rejected when the full conditional distribution $f(\cdot|W_{obs}, \theta)$ is in favor of drawing sparse networks, thus affect the mixing of MCMC algorithm. Morris et al. (2008) suggest to use a TNT (tie / no tie) sampler, where in each step, an edge is added or removed with an equal probability. Combining the idea of TNT sampler, our algorithm works through following steps to draw $W_{true}^{t+1}$:

**Algorithm 1-1b**

1. Start with $W^0 = W^t$, where $W^t$ is the network in the $t$-th iteration in step 2 Algorithm 1;
2. Iteratively, in the $k$-th step,
   a. With an equal probability, do one of the two followings,
      i. Randomly pick a dyad $(i^k, j^k) \in E^c_k$, where $E^c_k$ is the set of non-edges for $W^k$,
      ii. Randomly pick a dyad $(i^k, j^k) \in E_k$, where $E_k$ is the set of edges for $W^k$.
   b. Propose a move to a new network $W^*$ constructed by
      \[
      W^*_{ij} = \begin{cases} 
      1 - W^k_{ij}, & \text{if } (i, j) = (i^k, j^k) \text{ or } (j^k, i^k), \\
      W^k_{ij}, & \text{otherwise.}
      \end{cases}
      \]
   c. Calculate the acceptance ratio
      \[
      r(W^k, W^*) = \frac{f(W^*|W_{obs}, \theta)q(W^k|W^*)}{f(W^k|W_{obs}, \theta)q(W^*|W^k)},
      \]
      where $q(W^*|W^k)$ is the probability to get $W^*$ based on $W^k$ and $q(W^k|W^*)$ is the probability to get $W^k$ based on $W^*$,
   d. Accept the proposal move to $W^*$ with probability
      \[
      a(W^k, W^*) = \min(1, r(W^k, W^*));
      \]
3. Stop when it converges.

3.2 Updating $\theta$

The procedure to draw the model parameters $\theta^{t+1}$ from the full conditional distribution $f(\cdot|W_{true}^{t+1})$ happens to be equivalent to estimate model parameters based on $W_{true}^{t+1}$. Two classical approaches are maximum pseudolikelihood estimation (MPLE) [Besag (1974), Strauss and Ikeda (1990)] and Monte Carlo maximum likelihood estimation (MCMLE) [Geyer and Thompson (1992)].

MPLE was first proposed in [Besag (1974)] and adapted to social network models in [Strauss and Ikeda (1990)]. It assumes the weak dependence between variables in the network so that the likelihood can be well approximated by pseudolikelihood functions. But the assumption itself is in general
inappropriate, and Friel et al. (2009) have proved that under certain cases, the MPLE becomes inefficient.

MCMLE was first introduced in [Geyer and Thompson (1992)] which aims to tackle statistical models with intractable normalizing constants. In an ERGM model, the intractable normalizing constant refers to $z(\theta)$ in (2). The MCMLE involves a procedure to sample networks from the ERGM model $P(W|\theta_0)$, where the initial parameters $\theta_0$ should ideally lie very close to the maximum likelihood estimator (MLE). The result of MCMLE is very sensitive to the choice of $\theta_0$. If $\theta_0$ lies within the degeneracy region [Handcock et al. (2003), Rinaldo et al. (2009)], the estimate would be very poor.

To tackle the problems caused by the inherent intractable normalizing constant together with the degeneracy behavior in ERGM, Caimo and Friel (2011) proposed a Bayesian framework using the exchange algorithm [Murray et al. (2012)]. To avoid the problem in the MCMLE algorithm caused by choices of $\theta_0$, exchange algorithm samples from an augmented distribution

$$ P(\theta^*, W^*, \theta^t|W_{true}^{t+1}) \propto P(W_{true}^{t+1}|\theta^*) \pi(\theta^t) P(W^*|\theta^*), \quad (9) $$

where $W_{true}^{t+1}$ and $\theta^t$ are the network and model parameters in the $t$-th iteration in step 2 Algorithm 1, $P(W^*|\theta^*)$ follows the same distribution as $P(W_{true}^{t+1}|\theta^t)$ and $\pi(\theta^t)$ is the prior for parameter $\theta^t$. Let $q(\theta^*|\theta^t)$ be an appropriate proposal distribution (e.g., a random walk centered at $\theta^t$) to draw $\theta^*$ based on $\theta^t$, then the algorithm can be written in following steps:

**Algorithm 1-2**

1. Draw $\theta^*$ from $q(\cdot|\theta^t)$;
2. Draw $W^*$ from $P(\cdot|\theta^*)$;
3. Accept the proposal move from $\theta^t$ to $\theta^*$ with probability

$$ a(\theta^t, \theta^*) = \min \left(1, \frac{P(W^*|\theta^*) \pi(\theta^t) q(\theta^*|\theta^t) P(W_{true}^{t+1}|\theta^*)}{P(W_{true}^{t+1}|\theta^*) \pi(\theta^t) q(\theta^*|\theta^t) P(W^*|\theta^*)} \right). \quad (10) $$

Notice that in (10), two normalizing constants $z(\theta^*)$ and $z(\theta^t)$ are involved in both the numerator and denominator, hence cancel out. Through Algorithm 1-2, we can draw samples from the augmented distribution $P(\theta^*, W^*, \theta^t|W_{true}^{t+1})$, thus obtain the marginalized estimate of parameters $\theta$. 


Table 1: Summary statistics in \(W_{\text{true}}\) and \(W_{\text{obs}}\).

<table>
<thead>
<tr>
<th>Summary statistics</th>
<th>(W_{\text{true}})</th>
<th>(W_{\text{obs}})</th>
</tr>
</thead>
<tbody>
<tr>
<td># of edges</td>
<td>96</td>
<td>120</td>
</tr>
<tr>
<td># of two stars</td>
<td>130</td>
<td>230</td>
</tr>
</tbody>
</table>

4 Simulation results

In our simulation, let the number of vertices in the network \(n = |V| = 100\). Choose the summary statistics of interest to be the number of edges and the number of two stars, i.e.,

\[
s_1(W) = \sum_{i<j} W_{ij},
\]

\[
s_2(W) = \sum_{i=1}^{n} \sum_{j,k \neq i,j<k} W_{ij}W_{ik}.
\]

The ERGM model in (2) is thus

\[
P(W = W | \theta) \propto \exp \left( \theta_1 \sum_{i<j} W_{ij} + \theta_2 \sum_{i=1}^{n} \sum_{j,k \neq i,j<k} W_{ij}W_{ik} \right).
\]

Set the model parameters to be \((\theta_1, \theta_2) = (-1, -1)\) and the noise parameters \(p\) and \(q\) introduced in Section 2.3 to be \(p = 0.01\), \(q = 0.005\). To get a glance of the impact of measurement errors under such setups, we can draw multiple networks from (13), add noise to each network we drew, and compare the discrepancies in the summary statistics before / after adding noise. The discrepancies are shown in Figure 1.

In our simulation, we draw one network from (13) with \((\theta_1, \theta_2) = (-1, -1)\) and treat it as the true underlying network \(W_{\text{true}}\). We then obtain a network by adding noise onto \(W_{\text{true}}\) and treat it as the observed network with measurement errors \(W_{\text{obs}}\). The summary statistics for \(W_{\text{true}}\) and \(W_{\text{obs}}\) are summarized in Table 1.

To implement the Gibbs sampler in Algorithm 1, we place a flat multivariate normal prior to the model parameters \(\theta = (\theta_1, \theta_2)^\top\),

\[
\pi(\theta) \sim \mathcal{N}(0, 15^2 I_2),
\]

\(10\)
Figure 1: Boxplots for the summary statistics in 12500 networks drawn from (13) before (left, labeled by “True”) / after (right, labeled by “With errors”) adding noise. (a) Boxplots for the number of edges. (b) Boxplots for the number of two stars.

Table 2: Posterior mean and standard deviation for each parameter, plus standard error of mean (ignoring the autocorrelation of the chain) and time-series standard error.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mean</th>
<th>SD</th>
<th>Naive SE</th>
<th>Time-series SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_1$</td>
<td>-1.3578</td>
<td>0.5393</td>
<td>0.004824</td>
<td>0.08961</td>
</tr>
<tr>
<td>$\theta_2$</td>
<td>-0.8046</td>
<td>0.1903</td>
<td>0.001702</td>
<td>0.03367</td>
</tr>
</tbody>
</table>

where $I_2$ is the identity matrix with dimension 2. In step 1 Algorithm 1-2, we use two independent random walks for $\theta_1^t$ and $\theta_2^t$ separately, i.e.,

$$q(\cdot | \theta_1^t) \sim N(0, \sigma_1^2), \quad (15)$$

$$q(\cdot | \theta_2^t) \sim N(0, \sigma_2^2), \quad (16)$$

where $\sigma_1$ and $\sigma_2$ are pre-tuned and set to be 0.25 and 0.0001, respectively.

We run 12,500 iterations in total, using the first half to tune hyperparameters and second half to make inference. The traceplots of model parameters drawn using Algorithm 1 are shown in Figure 2, and the estimations of model parameters and summary statistics based on the samples are summarized in Table 2 and Table 3, respectively.

We can use the estimate of $\theta$ based on $W_{true}$ as a benchmark and compare
Figure 2: Traceplots for the model parameters drawn using Algorithm 1. The first half (1 : 12500) has been discarded. (a) Traceplot for $\theta_1$, the parameter for number of edges. (b) Traceplot for $\theta_2$, the parameter for number of two stars.

Table 3: Posterior mean and standard deviation for each summary statistic, plus standard error of mean (ignoring the autocorrelation of the chain) and time-series standard error.

<table>
<thead>
<tr>
<th>Summary statistics</th>
<th>Mean</th>
<th>SD</th>
<th>Naive SE</th>
<th>Time-series SE</th>
</tr>
</thead>
<tbody>
<tr>
<td># of edges</td>
<td>96.47</td>
<td>5.487</td>
<td>0.04907</td>
<td>1.179</td>
</tr>
<tr>
<td># of two stars</td>
<td>131.67</td>
<td>18.541</td>
<td>0.16584</td>
<td>3.870</td>
</tr>
</tbody>
</table>
Table 4: *Comparison of the posterior mean and model parameters estimated based on $W_{true}$, $W_{obs}$.*

<table>
<thead>
<tr>
<th>Parameters</th>
<th>$W_{true}$</th>
<th>$W_{obs}$</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_1$</td>
<td>-1.5252</td>
<td>-1.8524</td>
<td>-1.3578</td>
</tr>
<tr>
<td>$\theta_2$</td>
<td>-0.7425</td>
<td>-0.4323</td>
<td>-0.8046</td>
</tr>
</tbody>
</table>

Figure 3: Boxplots for the summary statistics in 12500 networks drawn using Algorithm 1 (left, labeled by “Sampled”) and drawn from (13) before (middle, labeled by “True”) / after (right, labeled by “With errors”) adding noise. (a) Boxplots for the number of edges. (b) Boxplots for the number of two stars.

the estimations based on $W_{obs}$ and the posterior mean. The comparison is summarized in Table 4. The estimations based on $W_{true}$ and $W_{obs}$ are obtained using Caimo and Friel (2011)’s framework as mentioned in Section 3.2. We can also compare the summary statistics of networks drawn using Algorithm 1 with that of networks drawn from (13) with / without measurement errors. The comparison is shown in Figure 3.

5 Discussion

In the paper, we proposed a Bayesian framework to make inference for the summary statistics and model parameters of network data under consideration of measurement errors. A Gibbs sampler is constructed, which consists of an MCMC algorithm to draw true underlying networks and a Bayesian
approach [Caimo and Friel (2011)] to draw model parameters. The full conditional distribution used to draw the underlying true networks is derived. Based on the samples we draw, estimates of both summary statistics and model parameters are obtained.

There are a few concerns about our framework. Summary statistics of networks drawn from the ERGM models are sensitive to natural model parameters [Rinaldo et al. (2009)], which can affect the mixing of the algorithm. Chatterjee et al. (2013) and Rinaldo et al. (2009) suggested to reparameterize the model, and for certain models, e.g. two-star model, a few reparameterization methods have been introduced [Mukherjee (2013), Park and Newman (2004)]. On the other hand, since our algorithm contains multiple MCMC procedures, computational cost can be high, especially when the network is large and summary statistics are costly to compute. Furthermore, in real applications, measurement errors may not be homogenous, or not even known to us. How to combine heterogeneous or unknown measurement errors remains a future interest.

References


Chatterjee, S., P. Diaconis, et al. (2013). Estimating and understanding


**Appendix**

In this section, we would derive the full conditional distribution $f(W_{\text{true}}|W_{\text{obs}}, \theta)$ in (7). Recall in Section 2.3, we assume that

\[
P(W_{\text{obs}}^{ij} = 0|W_{\text{true}}^{ij} = 1) = p \quad (17)
\]

\[
P(W_{\text{obs}}^{ij} = 1|W_{\text{true}}^{ij} = 0) = q \quad (18)
\]

Therefore, the conditional distribution $f(W_{\text{obs}}|W_{\text{true}})$ can be expressed as

\[
f(W_{\text{obs}}|W_{\text{true}}) = q^{N^+} (1-q)^{M_0-N^+} p^{N^-} (1-p)^{M_1-N^-}
\]

\[
= \exp \left[ N^+ \log q + (M_0 - N^+) \log(1-q) + N^- \log p + (M_1 - N^-) \log(1-p) \right] \quad (19)
\]
where $M_0$ and $M_1$ denote the number of non-edges and edges in $W_{true}$ respectively, and $N^+$ and $N^-$ denote the number of $+1$ and $-1$'s in $W_{noise} = W_{obs} - W_{true}$, respectively. In other words, $W_{ij}^{noise} = +1$ means the dyad $(i, j)$ is non-edge in $W_{true}$ but edge in $W_{obs}$, while $W_{ij}^{noise} = -1$ means it is edge in $W_{true}$ but non-edge in $W_{obs}$. Meanwhile, we can interpret $M_0 - N^+$ as the number of dyads which are non-edges in both $W_{true}$ and $W_{obs}$, while on the other hand $M_1 - N^-$ as the number of dyads which are edges in both. In this way, we can reformulate (19) as

\[
\begin{align*}
    f(W_{obs}|W_{true}) &= \exp \left[ \sum_{W_{ij}^{obs} = 0} W_{ij}^{true} \log p + \sum_{W_{ij}^{obs} = 0} (1 - W_{ij}^{true}) \log (1 - q) \right] \\
    &\quad \times \exp \left[ \sum_{W_{ij}^{obs} = 1} (1 - W_{ij}^{true}) \log q + \sum_{W_{ij}^{obs} = 1} W_{ij}^{true} \log (1 - p) \right] \\
    &\propto \exp \left( \sum_{W_{ij}^{obs} = 0} W_{ij}^{true} \log \frac{p}{1 - q} \right) + \exp \left( \sum_{W_{ij}^{obs} = 1} W_{ij}^{true} \log \frac{1 - p}{q} \right)
\end{align*}
\]

For simplicity, assume the ERGM model only contains one summary statistic, the number of edges $s(W) = \sum_{i<j} W_{ij}$. Then the likelihood function $f(W_{true}|\theta)$ can be written as

\[
f(W_{true}|\theta) \propto \exp \left( \theta \sum_{i<j} W_{ij}^{true} \right) \tag{21}
\]

The dyad set can be divided into two subsets based on $W_{obs}$: $E_{obs} = \{(i, j) : W_{ij}^{obs} = 1\}$ and $E_{obs}^c = \{(i, j) : W_{ij}^{obs} = 0\}$, and (21) can be rewritten based on this division

\[
f(W_{true}|\theta) \propto \exp \left( \theta \sum_{W_{ij}^{obs} = 0} W_{ij}^{true} + \theta \sum_{W_{ij}^{obs} = 1} W_{ij}^{true} \right) \tag{22}
\]
Utilizing the Bayes formula, together with (20) and (22), we have the full conditional distribution $f(W_{true}|W_{obs}, \theta)$ expressed by

$$f(W_{true}|W_{obs}, \theta) \propto f(W_{obs}|W_{true}, \theta) f(W_{true}|\theta)$$

$$= f(W_{obs}|W_{true}) f(W_{true}|\theta)$$

$$\propto \exp \left( \sum_{W_{ij}^{true}=0} W_{ij}^{true} \log \frac{p}{1-q} \right) + \exp \left( \sum_{W_{ij}^{true}=1} W_{ij}^{true} \log \frac{1-p}{q} \right)$$

$$\exp \left( \theta \sum_{W_{obs}^{true}=0} W_{ij}^{true} + \theta \sum_{W_{obs}^{true}=1} W_{ij}^{true} \right)$$

$$= \exp \left[ \left( \theta + \log \frac{p}{1-q} \right) \sum_{W_{obs}^{true}=0} W_{ij}^{true} + \left( \theta + \log \frac{1-p}{q} \right) \sum_{W_{obs}^{true}=1} W_{ij}^{true} \right]$$

which is just (7) when $s(W)$ is the number of edges. For models with other summary statistics, we can see through the derivation above, other summary statistics will not be affected. Therefore, we obtain (7).