

Exponential Random Graph Models for Network Data

David Hunter

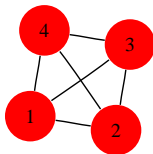
Department of Statistics
Penn State University

Student Organized Seminar, March 17, 2006

Outline

- 1 **Statistical Models for Networks**
 - What's a network?
 - What's an ERGM?
- 2 **Difficulties of fitting the ERGM**
 - Why MLE is difficult
 - Change Statistics and Maximum Pseudolikelihood
- 3 **Favored Approach: Approximate MLE via MCMC**
 - Law of Large Numbers to the Rescue
 - Obtaining samples via MCMC
 - A Numerical Example

What is a network?

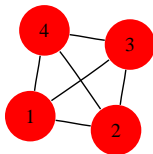


Definition

A representation of "relational data" in the form of a mathematical graph: A set of nodes along with a set of edges connecting some pairs of nodes.

- Edges can have directions and/or values. . .
. . . but for now, we'll assume undirected, binary (either on or off) edges.
- Notation: A symmetric matrix x of 0's and 1's.

What is a network?



Definition

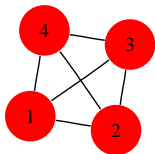
A representation of "relational data" in the form of a mathematical graph: A set of nodes along with a set of edges connecting some pairs of nodes.

- Edges can have directions and/or values. . .

. . . but for now, we'll assume undirected, binary (either on or off) edges.

- Notation: A symmetric matrix x of 0's and 1's.

What is a network?

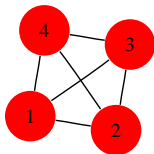


Definition

A representation of "relational data" in the form of a mathematical graph: A set of nodes along with a set of edges connecting some pairs of nodes.

- Edges can have directions and/or values. . .
. . . but for now, we'll assume undirected, binary (either on or off) edges.
- Notation: A symmetric matrix x of 0's and 1's.

What is a network?



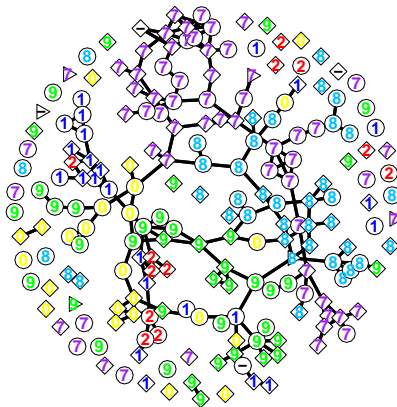
Definition

A representation of "relational data" in the form of a mathematical graph: A set of nodes along with a set of edges connecting some pairs of nodes.

- Edges can have directions and/or values. . .
. . . but for now, we'll assume undirected, binary (either on or off) edges.
- Notation: A symmetric matrix x of 0's and 1's.

Example Network: High School Friendship Data

School 10: 205 Students



- An edge indicates a mutual friendship.
- Colored labels give grade level, 7 through 12.
- Circles = female, squares = male, triangles = unknown.

Why study networks?

Many applications, including

- Epidemiology: Dynamics of disease spread
- Business: Viral marketing, word of mouth
- Telecommunications: WWW connectivity, phone calls
- Counterterrorism: Robustifying/attacking networks
- Political Science: Coalition formation dynamics
- ...

(It's a little embarrassing even to make such a list, because so many important items are missing.)

Why study networks?

Many applications, including

- Epidemiology: Dynamics of disease spread
- Business: Viral marketing, word of mouth
- Telecommunications: WWW connectivity, phone calls
- Counterterrorism: Robustifying/attacking networks
- Political Science: Coalition formation dynamics
- ...

(It's a little embarrassing even to make such a list, because so many important items are missing.)

What is an ERGM?

Exponential Random Graph Model (ERGM)

$$P_{\theta}(X = x) \propto \exp\{\theta^t s(x)\}$$

or

$$P_{\theta}(X = x) = \frac{\exp\{\theta^t s(x)\}}{c(\theta)},$$

where

- X is a random network on n nodes (a matrix of 0's and 1's)
- θ is a vector of parameters
- $s(x)$ is a known vector of graph statistics on x

Whence the name ERGM?

Exponential Family

Whenever the density of a random variable may be written

$$f(x) \propto \exp\{\theta^t s(x)\},$$

the family of all such random variables (for all possible θ) is called an **exponential family**.

- Since the random graphs in our model form an exponential family, we call the model an **exponential random graph model**.
- “ERGM” is easier to pronounce than “EFRGM”!

Whence the name ERGM?

Exponential Family

Whenever the density of a random variable may be written

$$f(x) \propto \exp\{\theta^t s(x)\},$$

the family of all such random variables (for all possible θ) is called an **exponential family**.

- Since the random graphs in our model form an exponential family, we call the model an **exponential random graph model**.
- “ERGM” is easier to pronounce than “EFRGM”!

Maximum Likelihood Estimation

The model:

$$P_{\theta}(X = x) = \frac{\exp\{\theta^t s(x)\}}{c(\theta)}, \text{ where } s(x^{\text{obs}}) = 0$$

- It follows that $c(\theta)$ is a normalizing “constant”:

$$c(\theta) = \sum_{\substack{\text{all possible} \\ \text{graphs } y}} \exp\{\theta^t s(y)\}.$$

- Replacing $s(x)$ by $s(x) - s(x^{\text{obs}})$ leaves $P_{\theta}(X = x)$ unchanged; thus, we “recenter” $s(x)$ so that $s(x^{\text{obs}}) = 0$.

Maximum Likelihood Estimation

The model:

$$P_{\theta}(X = x) = \frac{\exp\{\theta^t s(x)\}}{c(\theta)}, \text{ where } s(x^{\text{obs}}) = 0$$

- It follows that $c(\theta)$ is a normalizing “constant”:

$$c(\theta) = \sum_{\substack{\text{all possible} \\ \text{graphs } y}} \exp\{\theta^t s(y)\}.$$

- Replacing $s(x)$ by $s(x) - s(x^{\text{obs}})$ leaves $P_{\theta}(X = x)$ unchanged; thus, we “recenter” $s(x)$ so that $s(x^{\text{obs}}) = 0$.

Maximum Likelihood Estimation

The model:

$$P_{\theta}(X = x) = \frac{\exp\{\theta^t s(x)\}}{c(\theta)}, \text{ where } s(x^{\text{obs}}) = 0$$

- It follows that $c(\theta)$ is a normalizing “constant”:

$$c(\theta) = \sum_{\substack{\text{all possible} \\ \text{graphs } y}} \exp\{\theta^t s(y)\}.$$

- Replacing $s(x)$ by $s(x) - s(x^{\text{obs}})$ leaves $P_{\theta}(X = x)$ unchanged; thus, we “recenter” $s(x)$ so that $s(x^{\text{obs}}) = 0$.

Why it is difficult to find the MLE

The model:

$$P_{\theta}(X = x) = \frac{\exp\{\theta^t s(x)\}}{c(\theta)}, \text{ where } s(x^{\text{obs}}) = 0$$

- The loglikelihood function is

$$\ell(\theta) = -\log c(\theta) = \log \sum_{\substack{\text{all possible} \\ \text{graphs } y}} \exp\{\theta^t s(y)\}.$$

- Merely evaluating (let alone maximizing) $\ell(\theta)$ is somewhat computationally burdensome. . .

Why it is difficult to find the MLE

The model:

$$P_{\theta}(X = x) = \frac{\exp\{\theta^t s(x)\}}{c(\theta)}, \text{ where } s(x^{\text{obs}}) = 0$$

- The loglikelihood function is

$$\ell(\theta) = -\log c(\theta) = \log \sum_{\substack{\text{all possible} \\ \text{graphs } y}} \exp\{\theta^t s(y)\}.$$

- Merely evaluating (let alone maximizing) $\ell(\theta)$ is somewhat computationally burdensome. . .

Why it is difficult to find the MLE

The model:

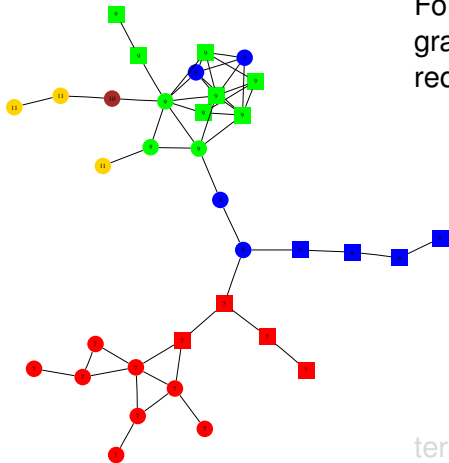
$$P_{\theta}(X = x) = \frac{\exp\{\theta^t s(x)\}}{c(\theta)}, \text{ where } s(x^{\text{obs}}) = 0$$

- The loglikelihood function is

$$\ell(\theta) = -\log c(\theta) = \log \sum_{\substack{\text{all possible} \\ \text{graphs } y}} \exp\{\theta^t s(y)\}.$$

- Merely evaluating (let alone maximizing) $\ell(\theta)$ is somewhat computationally burdensome...

How burdensome?

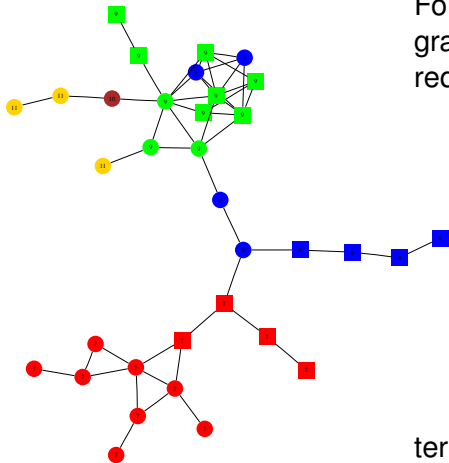


For this undirected, 34-node graph, computing $c(\theta)$ directly requires summation of

7,547,924,849,643,082,704,483,
 109,161,976,537,781,833,842,
 440,832,880,856,752,412,600,
 491,248,324,784,297,704,172,
 253,450,355,317,535,082,936,
 750,061,527,689,799,541,169,
 259,849,585,265,122,868,502,
 865,392,087,298,790,653,952

terms.

How burdensome?



For this undirected, 34-node graph, computing $c(\theta)$ directly requires summation of

7,547,924,849,643,082,704,483,
 109,161,976,537,781,833,842,
 440,832,880,856,752,412,600,
 491,248,324,784,297,704,172,
 253,450,355,317,535,082,936,
 750,061,527,689,799,541,169,
 259,849,585,265,122,868,502,
 865,392,087,298,790,653,952

terms.

Conditional log-odds of an edge

Notation: For a network x and a pair (i, j) of nodes,

- $x_{ij} = 0$ or 1 , depending on whether there is an edge
- x_{ij}^c denotes the status of all pairs in x other than (i, j)
- x_{ij}^+ denotes the same network as x but with $x_{ij} = 1$
- x_{ij}^- denotes the same network as x but with $x_{ij} = 0$

Conditional log-odds of an edge

Notation: For a network x and a pair (i, j) of nodes,

- $x_{ij} = 0$ or 1 , depending on whether there is an edge
- x_{ij}^c denotes the status of all pairs in x other than (i, j)
- x_{ij}^+ denotes the same network as x but with $x_{ij} = 1$
- x_{ij}^- denotes the same network as x but with $x_{ij} = 0$

Conditional on $X_{ij}^c = x_{ij}^c$, X has only two possible states, depending on whether $X_{ij} = 0$ or $X_{ij} = 1$.

Conditional log-odds of an edge

Notation: For a network x and a pair (i, j) of nodes,

- $x_{ij} = 0$ or 1 , depending on whether there is an edge
- x_{ij}^c denotes the status of all pairs in x other than (i, j)
- x_{ij}^+ denotes the same network as x but with $x_{ij} = 1$
- x_{ij}^- denotes the same network as x but with $x_{ij} = 0$

Conditional on $X_{ij}^c = x_{ij}^c$, X has only two possible states, depending on whether $X_{ij} = 0$ or $X_{ij} = 1$.

Let's calculate the ratio of the two respective probabilities:

Conditional log-odds of an edge

Notation: For a network x and a pair (i, j) of nodes,

- $x_{ij} = 0$ or 1 , depending on whether there is an edge
- x_{ij}^c denotes the status of all pairs in x other than (i, j)
- x_{ij}^+ denotes the same network as x but with $x_{ij} = 1$
- x_{ij}^- denotes the same network as x but with $x_{ij} = 0$

$$\frac{P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)}{P(X_{ij} = 0 | X_{ij}^c = x_{ij}^c)} = \frac{\exp\{\theta^t s(x_{ij}^+)\}}{\exp\{\theta^t s(x_{ij}^-)\}}$$

Conditional log-odds of an edge

Notation: For a network x and a pair (i, j) of nodes,

- $x_{ij} = 0$ or 1 , depending on whether there is an edge
- x_{ij}^c denotes the status of all pairs in x other than (i, j)
- x_{ij}^+ denotes the same network as x but with $x_{ij} = 1$
- x_{ij}^- denotes the same network as x but with $x_{ij} = 0$

$$\frac{P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)}{P(X_{ij} = 0 | X_{ij}^c = x_{ij}^c)} = \exp\{\theta^t [s(x_{ij}^+) - s(x_{ij}^-)]\}$$

Conditional log-odds of an edge

Notation: For a network x and a pair (i, j) of nodes,

- $x_{ij} = 0$ or 1 , depending on whether there is an edge
- x_{ij}^c denotes the status of all pairs in x other than (i, j)
- x_{ij}^+ denotes the same network as x but with $x_{ij} = 1$
- x_{ij}^- denotes the same network as x but with $x_{ij} = 0$

$$\log \frac{P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)}{P(X_{ij} = 0 | X_{ij}^c = x_{ij}^c)} = \theta^t [s(x_{ij}^+) - s(x_{ij}^-)]$$

Conditional log-odds of an edge

Notation: For a network x and a pair (i, j) of nodes,

- $\Delta(s(x))_{ij}$ denotes the vector of change statistics,

$$\Delta(s(x))_{ij} = s(x_{ij}^+) - s(x_{ij}^-).$$

So $\Delta(s(x))_{ij}$ is the conditional log-odds of edge (i, j) .

$$\log \frac{P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)}{P(X_{ij} = 0 | X_{ij}^c = x_{ij}^c)} = \theta^t \Delta(s(x))_{ij}$$

Maximum Pseudolikelihood: Alternative to MLE?

- What if we approximate the marginal $P(X_{ij} = 1)$ by the conditional $P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)$?
- Then the X_{ij} are independent with

$$\log \frac{P(X_{ij} = 1)}{P(X_{ij} = 0)} = \theta^t \Delta(s(x^{\text{obs}}))_{ij},$$

so we obtain $\hat{\theta}$ using simple logistic regression.

- Result: The **maximum pseudolikelihood estimate**.
- Unfortunately, little is known about the quality of MPL estimates.

Maximum Pseudolikelihood: Alternative to MLE?

- What if we approximate the marginal $P(X_{ij} = 1)$ by the conditional $P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)$?
- Then the X_{ij} are independent with

$$\log \frac{P(X_{ij} = 1)}{P(X_{ij} = 0)} = \theta^t \Delta(s(x^{\text{obs}}))_{ij},$$

so we obtain $\hat{\theta}$ using simple logistic regression.

- Result: The **maximum pseudolikelihood estimate**.
- Unfortunately, little is known about the quality of MPL estimates.

Maximum Pseudolikelihood: Alternative to MLE?

- What if we approximate the marginal $P(X_{ij} = 1)$ by the conditional $P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)$?
- Then the X_{ij} are independent with

$$\log \frac{P(X_{ij} = 1)}{P(X_{ij} = 0)} = \theta^t \Delta(s(x^{\text{obs}}))_{ij},$$

so we obtain $\hat{\theta}$ using simple logistic regression.

- Result: The **maximum pseudolikelihood estimate**.
- Unfortunately, little is known about the quality of MPL estimates.

Maximum Pseudolikelihood: Alternative to MLE?

- What if we approximate the marginal $P(X_{ij} = 1)$ by the conditional $P(X_{ij} = 1 | X_{ij}^c = x_{ij}^c)$?
- Then the X_{ij} are independent with

$$\log \frac{P(X_{ij} = 1)}{P(X_{ij} = 0)} = \theta^t \Delta(s(x^{\text{obs}}))_{ij},$$

so we obtain $\hat{\theta}$ using simple logistic regression.

- Result: The **maximum pseudolikelihood estimate**.
- Unfortunately, little is known about the quality of MPL estimates.

MLE Revisited

- Remember, $c(\theta)$ is *really* hard to compute.
- However, suppose we fix θ_0 . A bit of algebra shows that

$$E_{\theta_0} [\exp \{(\theta - \theta_0)^t s(X)\}] = \frac{c(\theta)}{c(\theta_0)}.$$

- Thus, $c(\theta)/c(\theta_0)$ is the expectation of a function of a random network, where the random behavior is governed by the known constant θ_0 .

MLE Revisited

- Remember, $c(\theta)$ is *really* hard to compute.
- However, suppose we fix θ_0 . A bit of algebra shows that

$$E_{\theta_0} [\exp \{(\theta - \theta_0)^t s(X)\}] = \frac{c(\theta)}{c(\theta_0)}.$$

- Thus, $c(\theta)/c(\theta_0)$ is the expectation of a function of a random network, where the random behavior is governed by the known constant θ_0 .

MLE Revisited

- Remember, $c(\theta)$ is *really* hard to compute.
- However, suppose we fix θ_0 . A bit of algebra shows that

$$E_{\theta_0} [\exp \{(\theta - \theta_0)^t s(X)\}] = \frac{c(\theta)}{c(\theta_0)}.$$

- Thus, $c(\theta)/c(\theta_0)$ is the expectation of a function of a random network, where the random behavior is governed by the known constant θ_0 .

Law of Large Numbers to the Rescue!

The LOLN suggests that we approximate an unknown population mean by a sample mean.

Thus,

$$\begin{aligned}c(\theta)/c(\theta_0) &= E_{\theta_0} \left(\exp \{ (\theta - \theta_0)^t s(X) \} \right) \\ &\approx \frac{1}{M} \sum_{i=1}^M \exp \{ (\theta - \theta_0)^t s(X^{(i)}) \},\end{aligned}$$

where $X^{(1)}, X^{(2)}, \dots, X^{(M)}$ is a random sample of networks from the distribution defined by the ERGM with parameter θ_0 .

Law of Large Numbers to the Rescue!

The LOLN suggests that we approximate an unknown population mean by a sample mean.

Thus,

$$\begin{aligned}c(\theta)/c(\theta_0) &= E_{\theta_0} \left(\exp \{ (\theta - \theta_0)^t s(X) \} \right) \\ &\approx \frac{1}{M} \sum_{i=1}^M \exp \{ (\theta - \theta_0)^t s(X^{(i)}) \},\end{aligned}$$

where $X^{(1)}, X^{(2)}, \dots, X^{(M)}$ is a random sample of networks from the distribution defined by the ERGM with parameter θ_0 .

An approximate loglikelihood

- Using the LOLN approximation, we find

$$\begin{aligned}
 \ell(\theta) - \ell(\theta_0) &= -\log \frac{c(\theta)}{c(\theta_0)} \\
 &= -\log E_{\theta_0} \left(\exp \left\{ (\theta - \theta_0)^t s(X) \right\} \right) \\
 &\approx -\log \frac{1}{M} \sum_{i=1}^M \exp \left\{ (\theta - \theta_0)^t s(X^{(i)}) \right\}.
 \end{aligned}$$

- Now, all we need is a random sample of networks from P_{θ_0} .

An approximate loglikelihood

- Using the LOLN approximation, we find

$$\begin{aligned}
 \ell(\theta) - \ell(\theta_0) &= -\log \frac{c(\theta)}{c(\theta_0)} \\
 &= -\log E_{\theta_0} \left(\exp \left\{ (\theta - \theta_0)^t s(X) \right\} \right) \\
 &\approx -\log \frac{1}{M} \sum_{i=1}^M \exp \left\{ (\theta - \theta_0)^t s(X^{(i)}) \right\}.
 \end{aligned}$$

- Now, all we need is a random sample of networks from P_{θ_0} .

An approximate loglikelihood

- Using the LOLN approximation, we find

$$\begin{aligned}
 \ell(\theta) - \ell(\theta_0) &= -\log \frac{c(\theta)}{c(\theta_0)} \\
 &= -\log E_{\theta_0} \left(\exp \left\{ (\theta - \theta_0)^t s(X) \right\} \right) \\
 &\approx -\log \frac{1}{M} \sum_{i=1}^M \exp \left\{ (\theta - \theta_0)^t s(X^{(i)}) \right\}.
 \end{aligned}$$

- Now, all we need is a random sample of networks from P_{θ_0} .

An approximate loglikelihood

- Using the LOLN approximation, we find

$$\begin{aligned}\ell(\theta) - \ell(\theta_0) &= -\log \frac{c(\theta)}{c(\theta_0)} \\ &= -\log E_{\theta_0} \left(\exp \left\{ (\theta - \theta_0)^t s(X) \right\} \right) \\ &\approx -\log \frac{1}{M} \sum_{i=1}^M \exp \left\{ (\theta - \theta_0)^t s(X^{(i)}) \right\}.\end{aligned}$$

- Now, all we need is a random sample of networks from P_{θ_0} .

Obtaining samples via MCMC

MCMC Idea:

Simulate a discrete-time Markov chain whose stationary distribution is the distribution we want to sample from.

We'll discuss two common ways to run such a Markov chain:

- Gibbs sampling
- A Metropolis algorithm

Obtaining samples via MCMC

MCMC Idea:

Simulate a discrete-time Markov chain whose stationary distribution is the distribution we want to sample from.

We'll discuss two common ways to run such a Markov chain:

- Gibbs sampling
- A Metropolis algorithm

Gibbs sampling

- First, select a pair of nodes at random, say (i, j) .
- Decide whether to set $X_{ij} = 0$ or $X_{ij} = 1$ at the next time step according to the conditional distribution of X_{ij} given the rest of the network (X_{ij}^c).
- Based on an earlier calculation, we obtain

$$P_{\theta_0}(X_{ij} = 1 | X_{ij}^c = x_{ij}^c) = \frac{\exp\{\theta_0^t \Delta(s(x))_{ij}\}}{(1 + \exp\{\theta_0^t \Delta(s(x))_{ij}\})}$$

Note: To run the MCMC, the values of $s(x_{ij}^+)$ and $s(x_{ij}^-)$ are not needed; only the difference $\Delta(s(x))_{ij}$ matters.

Gibbs sampling

- First, select a pair of nodes at random, say (i, j) .
- Decide whether to set $X_{ij} = 0$ or $X_{ij} = 1$ at the next time step according to the conditional distribution of X_{ij} given the rest of the network (X_{ij}^c).
- Based on an earlier calculation, we obtain

$$P_{\theta_0}(X_{ij} = 1 | X_{ij}^c = x_{ij}^c) = \frac{\exp\{\theta_0^t \Delta(s(x))_{ij}\}}{(1 + \exp\{\theta_0^t \Delta(s(x))_{ij}\})}$$

Note: To run the MCMC, the values of $s(x_{ij}^+)$ and $s(x_{ij}^-)$ are not needed; only the difference $\Delta(s(x))_{ij}$ matters.

Gibbs sampling

- First, select a pair of nodes at random, say (i, j) .
- Decide whether to set $X_{ij} = 0$ or $X_{ij} = 1$ at the next time step according to the conditional distribution of X_{ij} given the rest of the network (X_{ij}^c).
- Based on an earlier calculation, we obtain

$$P_{\theta_0}(X_{ij} = 1 | X_{ij}^c = x_{ij}^c) = \frac{\exp\{\theta_0^t \Delta(s(x))_{ij}\}}{(1 + \exp\{\theta_0^t \Delta(s(x))_{ij}\})}.$$

Note: To run the MCMC, the values of $s(x_{ij}^+)$ and $s(x_{ij}^-)$ are not needed; only the difference $\Delta(s(x))_{ij}$ matters.

Metropolis algorithm

- First, select a pair of nodes at random, say (i, j) .
- Calculate the ratio

$$\begin{aligned}\pi &= \frac{P(X_{ij} \text{ changes} | X_{ij}^c = x_{ij}^c)}{P(X_{ij} \text{ does not change} | X_{ij}^c = x_{ij}^c)} \\ &= \exp\{\pm\theta_0^t \Delta(s(x))_{ij}\}\end{aligned}$$

- Accept the change of X_{ij} with probability $\min\{1, \pi\}$.
- This scheme generally has better properties than Gibbs sampling.

Note: To run the MCMC, the values of $s(x_{ij}^+)$ and $s(x_{ij}^-)$ are not needed; only the difference $\Delta(s(x))_{ij}$ matters.



Metropolis algorithm

- First, select a pair of nodes at random, say (i, j) .
- Calculate the ratio

$$\begin{aligned}\pi &= \frac{P(X_{ij} \text{ changes} | X_{ij}^c = x_{ij}^c)}{P(X_{ij} \text{ does not change} | X_{ij}^c = x_{ij}^c)} \\ &= \exp\{\pm\theta_0^t \Delta(s(x))_{ij}\}\end{aligned}$$

- Accept the change of X_{ij} with probability $\min\{1, \pi\}$.
- This scheme generally has better properties than Gibbs sampling.

Note: To run the MCMC, the values of $s(x_{ij}^+)$ and $s(x_{ij}^-)$ are not needed; only the difference $\Delta(s(x))_{ij}$ matters.



Metropolis algorithm

- First, select a pair of nodes at random, say (i, j) .
- Calculate the ratio

$$\begin{aligned}\pi &= \frac{P(X_{ij} \text{ changes} | X_{ij}^c = x_{ij}^c)}{P(X_{ij} \text{ does not change} | X_{ij}^c = x_{ij}^c)} \\ &= \exp\{\pm\theta_0^t \Delta(s(x))_{ij}\}\end{aligned}$$

- Accept the change of X_{ij} with probability $\min\{1, \pi\}$.
- This scheme generally has better properties than Gibbs sampling.

Note: To run the MCMC, the values of $s(x_{ij}^+)$ and $s(x_{ij}^-)$ are not needed; only the difference $\Delta(s(x))_{ij}$ matters.



Metropolis algorithm

- First, select a pair of nodes at random, say (i, j) .
- Calculate the ratio

$$\begin{aligned}\pi &= \frac{P(X_{ij} \text{ changes} | X_{ij}^c = x_{ij}^c)}{P(X_{ij} \text{ does not change} | X_{ij}^c = x_{ij}^c)} \\ &= \exp\{\pm\theta_0^t \Delta(s(x))_{ij}\}\end{aligned}$$

- Accept the change of X_{ij} with probability $\min\{1, \pi\}$.
- This scheme generally has better properties than Gibbs sampling.

Note: To run the MCMC, the values of $s(x_{ij}^+)$ and $s(x_{ij}^-)$ are not needed; only the difference $\Delta(s(x))_{ij}$ matters.

How should θ_0 be chosen?

- Theoretically, the estimated value of $\ell(\theta) - \ell(\theta_0)$ converges to the true value as the size of the MCMC sample increases, regardless of the value of θ_0 .
- However, this convergence can be agonizingly slow, especially if θ_0 is not chosen close to the maximizer of the likelihood.
- A choice that sometimes works is the MPLE (maximum pseudolikelihood estimate)

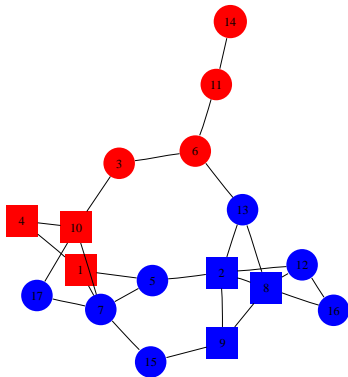
How should θ_0 be chosen?

- Theoretically, the estimated value of $\ell(\theta) - \ell(\theta_0)$ converges to the true value as the size of the MCMC sample increases, regardless of the value of θ_0 .
- However, this convergence can be agonizingly slow, especially if θ_0 is not chosen close to the maximizer of the likelihood.
- A choice that sometimes works is the MPLE (maximum pseudolikelihood estimate)

How should θ_0 be chosen?

- Theoretically, the estimated value of $\ell(\theta) - \ell(\theta_0)$ converges to the true value as the size of the MCMC sample increases, regardless of the value of θ_0 .
- However, this convergence can be agonizingly slow, especially if θ_0 is not chosen close to the maximizer of the likelihood.
- A choice that sometimes works is the MPLE (maximum pseudolikelihood estimate)

A numerical example



```
=====
Summary of output
=====
```

```
Newton-Raphson iterations: 32
MCMC sample of size 10000
```

```
Monte Carlo MLE Results:
```

	theta0	estimate	s.e.	p-value
match.grade	1.0706	1.4118	0.4988	0.0054
dmatch.sex.0	1.0383	1.4660	0.7482	0.0522
dmatch.sex.1	-0.9387	-0.7195	0.6767	0.2897
triangle	1.1864	1.0389	0.5750	0.0732
kstar1	9.3754	8.2408	5.3026	0.1226
kstar2	-8.1424	-7.5155	4.4219	0.0916
kstar3	5.2464	5.0092	3.3080	0.1324
kstar4	-2.4226	-2.4512	1.8068	0.1773

```
Log likelihood of g: -78.52976
```

Some Useful References

- Frank, O. and D. Strauss (1986), Markov graphs, *JASA*
- Geyer, C. J. and E. Thompson (1992), Constrained Monte Carlo maximum likelihood for dependent data, *J. Roy. Stat. Soc. B*
- Holland, P. W. and S. Leinhardt (1981), An exponential family of probability distributions for directed graphs, *JASA*
- Snijders, Tom A. B. (2002), Markov chain Monte Carlo estimation of exponential random graph models, *J. Soc. Struct.*
- Strauss, D. and M. Ikeda (1990), Pseudolikelihood estimation for social networks, *JASA*
- Wasserman, S. and P. Pattison (1996), Logit models and logistic regression for social networks: I. An introduction to Markov graphs and p^* , *Psychometrika*