TSCS Analysis

(Dynamic) Exponential Random Graph Models

Jude C. Hays

April 14, 2021

The Exponential Random Graph Model (ERGM)

- In an ERGM, the distribution for $\mathbf{Y}$, a random *nxn* network matrix, with support over $\mathbf{Y}_m$ is parameterized as

$$Pr_{\theta,Y_m}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\theta^T \mathbf{g}(\mathbf{y})\}}{\kappa(\theta, \mathbf{Y}_m)}, \mathbf{y} \in \mathbf{Y}_m$$

  where $\mathbf{y}$ is an observed network matrix, θ is a vector of coefficients, and $\mathbf{g}(\mathbf{y})$ is a vector of statistics calculated from the observed network.
- The denominator, κ, is the normalizing factor, which ensures a proper probability distribution.
- The sample space in $\mathbf{Y}_m$ contains up to $m = 2^{n(n-1)}$ networks, making the calculation of the normalizing constant the primary barrier to inference. (For example, with only 5 actors, there are more than one million possible configurations in a directed network.)

Estimating ERGMs

- The log-likelihood function is

$$\ell(\theta) = \theta^T \mathbf{g}(\mathbf{y}) - \log \kappa(\theta, \mathbf{Y}_m)$$

- Subtracting the log-likelihood for an arbitrary vector of coefficients, θ_0 gives

$$\ell(\theta) - \ell(\theta_0) = (\theta - \theta_0)^T \mathbf{g}(\mathbf{y}) - \log \left[\frac{\kappa(\theta, \mathbf{Y}_m)}{\kappa(\theta_0, \mathbf{Y}_m)} \right]$$

- The ratio of normalizing factors can be shown to equal

$$\frac{\kappa(\theta, \mathbf{Y}_m)}{\kappa(\theta_0, \mathbf{Y}_m)} = E_{\theta_m} \exp \left\{ (\theta - \theta_0)^T \mathbf{g}(\mathbf{Y}) \right\}$$

  where E_{θ_m} is the expectation assuming $\mathbf{Y}$ has the distribution Pr_{θ, Y_m}.
- This allows us to calculate the log-likelihood by sampling from $\mathbf{Y}_m$, which we do using Markov-Chain-Monte-Carlo (MCMC) methods.

Simulating Networks via Markov Chains

- The likelihood (more specifically, the log-ratio of likelihoods) can be approximated by

$$\ell(\theta) - \ell(\theta_0) \approx (\theta - \theta_0)^T \mathbf{g}(\mathbf{y}) - \log \left[\frac{1}{r} \sum_{i=1}^{r} \exp \left\{ (\theta - \theta_0)^T \mathbf{g}(\mathbf{Y}_i) \right\} \right]$$

  where $\mathbf{Y}_1, ..., \mathbf{Y}_r$ is a random sample from the distribution Pr_{θ, Y_m}.
- The coefficients in θ_0 should be as close to the maximum likelihood values as possible. Pseudo-likelihood values, which assume the edges in the network are mutually independent, can be obtained by logit regression.

Simulating Networks via Markov Chains

The MCMC sampling proceeds in the following steps

1. Start with a network from $\mathbf{Y}_m$ and make a large number of sampled Markov transitions. That is, choose pairs of nodes (ij) uniformly at random and set Y_{ij} equal to one or zero using the conditional probability

$$\text{logit}\left[\Pr_{\theta,\mathbf{Y}_m}\left(Y_{ij}=1|\mathbf{Y}_{ij}^c=\mathbf{y}_{ij}^c\right)\right]=\theta^T\delta_{\mathbf{g}}(\mathbf{y})_{ij}$$

   where $\delta_{\mathbf{g}}(\mathbf{y})_{ij}$ is a change statistic defined as

$$\delta_{\mathbf{g}}(\mathbf{y})_{ij}=\mathbf{g}(\mathbf{y}_{ij}^+)-\mathbf{g}(\mathbf{y}_{ij}^-)$$

   and the networks $\mathbf{y}_{ij}^+$ and $\mathbf{y}_{ij}^-$ are those realized by fixing $Y_{ij}=1$ and $Y_{ij}=0$ respectively.
2. Save the network and repeat $r-1$ times.
3. Use the r networks to approximate the ratio of normalizing factors.
4. Update the parameter vector and repeat from the top.

Adding Temporal Dynamics: TERGM

- A TERGM takes the form

$$\Pr(\mathbf{Y}^t = \mathbf{y}^t | \theta, \mathbf{Y}_m^{t-1}, \ldots, \mathbf{Y}_m^{t-K}) = \frac{\exp\{\theta^T \mathbf{g}(\mathbf{y}^t, \mathbf{y}^{t-1}, \ldots, \mathbf{y}^{t-K})\}}{\kappa(\theta, \mathbf{Y}_m^{t-1}, \ldots, \mathbf{Y}_m^{t-K})}, \mathbf{y}^t \in \mathbf{Y}_m^t$$

$$\Pr(\mathbf{Y}^{K+1}, \ldots, \mathbf{Y}^T | \theta, \mathbf{Y}^1, \ldots, \mathbf{Y}^K) = \prod_{t=K+1}^{T} \Pr(\mathbf{Y}^t | \theta, \mathbf{Y}^{t-K}, \ldots, \mathbf{Y}^{t-1})$$

- Chose the right K, and the ERGMs from $K+1$ to T are independent.

Memory Terms

- First, write the equation for a individual tie from the ERGM.

$$\pi_{ij}^{t}(\theta) = \Pr(y_{ij}^{t} = 1 | y_{\sim ij}^{t}, \theta) = \text{logit}^{-1}\left(\theta \delta_{\mathbf{g}}(\mathbf{y}^{t})_{ij}\right)$$

- A memory term is one designed to capture temporal dependence in a TERGM.
- A particularly important memory term is the positive autoregression term.

$$g_a = \sum y_{ij}^{t} y_{ij}^{t-1}, \text{where}$$

$\delta_{\mathbf{g}}(\mathbf{y}^{t})_{ij} = +1$ when $y_{ij}^{t-1} = 1$ and $+0$ otherwise.

Maximum Pseudo-Likelihood Estimation

- The MPLE replaces the joint likelihood used for ML estimation with a product of conditional likelihoods, or sums when these likelihoods are logged.

$$\arg\max_{\theta} = \sum_{t=K+1}^{T} \sum_{ij} \ln \left[\left(\pi_{ij}^{t}(\theta)\right)^{y_{ij}^{t}} \left(1 - \pi_{ij}^{t}(\theta)\right)^{1-y_{ij}^{t}} \right]$$

- MPLE assumes the conditional ties are independent.
- MPLE consistently estimates θ, but it is inconsistent when it comes to varcov(θ).
- Fortunately, standard errors can be bootstrapped using the package **btergm**.