Exponential Random Graph Models with Big Networks: Maximum Pseudolikelihood Estimation and the Parametric Bootstrap

Christian Schmid
In collaboration with Bruce Desmarais

The Pennsylvania State University

July 31st, 2017

Overview

- The Exponential Random Graph Model (ERGM) is a probability model for network data with an intractable normalizing constant.

Overview

- The Exponential Random Graph Model (ERGM) is a probability model for network data with an intractable normalizing constant.
- The state-of-the-art approach is to approximate the normalizing constant by Monte Carlo maximum likelihood (MCMLE).

Overview

- The Exponential Random Graph Model (ERGM) is a probability model for network data with an intractable normalizing constant.
- The state-of-the-art approach is to approximate the normalizing constant by Monte Carlo maximum likelihood (MCMLE).
- MCMLE is accurate when a large sample of networks is used, but also becomes computationally expensive for large networks.

Overview

- The Exponential Random Graph Model (ERGM) is a probability model for network data with an intractable normalizing constant.
- The state-of-the-art approach is to approximate the normalizing constant by Monte Carlo maximum likelihood (MCMLE).
- MCMLE is accurate when a large sample of networks is used, but also becomes computationally expensive for large networks.
- For large networks the consistent and computationally fast maximum pseudolikelihood (MPLE) is an option, even though it generally underestimates standard errors.

Overview

- The Exponential Random Graph Model (ERGM) is a probability model for network data with an intractable normalizing constant.
- The state-of-the-art approach is to approximate the normalizing constant by Monte Carlo maximum likelihood (MCMLE).
- MCMLE is accurate when a large sample of networks is used, but also becomes computationally expensive for large networks.
- For large networks the consistent and computationally fast maximum pseudolikelihood (MPLE) is an option, even though it generally underestimates standard errors.
- We show that a resampling method - the parametric bootstrap - results in accurate coverage probabilities for confidence intervals.

The Exponential Random Graph Model

Idea: Take the adjacency matrix of an observed network A with N nodes as a manifestation of a matrix-like random variable Y.

Definition:

$$\mathrm{P}(Y = A|\theta) = \frac{\exp(\theta^T \cdot \Gamma(A))}{c(\theta)}$$

where

- $\theta \in \mathbb{R}^q$, is a vector of parameters
- $\Gamma : \mathcal{A}(N) \to \mathbb{R}^q \ , \ A \mapsto (\Gamma_1(A), \dots, \Gamma_q(A))^T$, is a vector of network statistics
- $c(\theta) := \sum_{A^* \in \mathcal{A}(N)} \exp(\theta^T \cdot \Gamma(A^*))$, is a normalization constant

Parameter Estimation Method 1: MCMLE

Idea: Fix an auxiliary parameter vector $\theta_0 \in \mathbb{R}^q$. Then, we can show

$$\frac{c(\theta)}{c(\theta_0)} = \mathbb{E}_{\theta_0}\left[exp((\theta - \theta_0)^T \Gamma(A))\right]$$

Simulate a large number of random networks $A_1, \dots, A_L$ from the distribution P_{θ_0} using Metropolis-Hastings.

Then, by the law of big numbers, we get

$$\frac{c(\theta)}{c(\theta_0)} \approx \frac{1}{L} \sum_{i=1}^{L} \left[exp((\theta - \theta_0)^T \Gamma(A_i))\right]$$

Simulating Networks with MCMC

Choose a matrix $A^{(0)} \in \mathcal{A}(N)$ to start with and proceed as follows:

1. Randomly choose a dyad $A_{ij} := A[i,j]$ where $i \neq j$ from $A^{(k)}$
2. Compute the value

$$\pi := \frac{\mathrm{P}(Y_{ij} \neq A_{ij}^{(k)} | Y_{ij}^c = A_{ij}^c, \theta_0)}{\mathrm{P}(Y_{ij} = A_{ij}^{(k)} | Y_{ij}^c = A_{ij}^c, \theta_0)}$$

   where $Y_{ij}^c = A_{ij}^c$ is short for $Y_{pq} = A_{pq}$ for all $(p,q) \neq (i,j)$.

3. Fix $\delta := \min\{1, \pi\}$ and draw a random number Z from $\mathrm{Bin}(1, \delta)$. If
   - $Z = 0$, let $A^{(k+1)} := A^{(k)}$
   - $Z = 1$, set $A_{ij}^{(k+1)} = 1 - A_{ij}^{(k)}$
4. Start at step 1 with $A^{(k+1)}$.

Efficiency of MPLE/MCMLE

- MCMLE is in general favored over MPLE, but MPLE is quick and simple and does not require elaborate MCMC methods.
- MPLE approaches the MLE as the size of the network increases.
  $\Rightarrow$ For large networks MPLE is an alternative to the computationally expensive MCMLE.
- Furthermore, the MPLE outperforms the MCMLE if the number of simulated networks is not chosen large enough.
- It is even more remarkable that the number of simulated networks needed in order for the MCMLE to outperform the MPLE increases as the size of the network increases.

Simulation Study

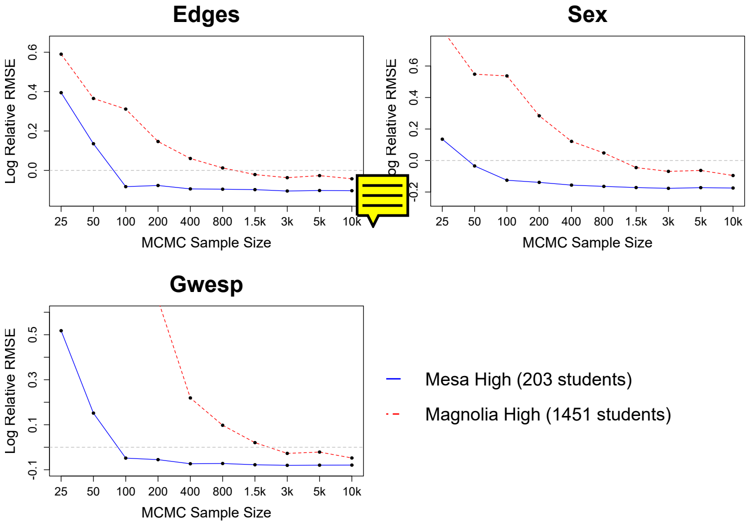
Data:
We use two friendship networks, Faux Mesa High (203 students) and Faux Magnolia High (1451 students), and use the same parametrization (edges, sex and gwesp)

Treatment:

- We take the MCMLE of both networks as the 'true' coefficient θ and simulated 500 networks using these coefficients.
- We estimate the coefficients of these 500 networks using MPLE and MCMLE.
- For every single simulated network the MCMLE calculation is being repeated several times for 25 to 10.000 simulated networks used in the likelihood approximation.

Simulation Study Results

This plot visualizes the log of the ratio of the root MSE of the MCMLE to the MPLE.



Parameter Estimation Method 2: Bootstrapped MPLE

In contrast to the MPLE, the MCMLE does not underestimate the standard error.

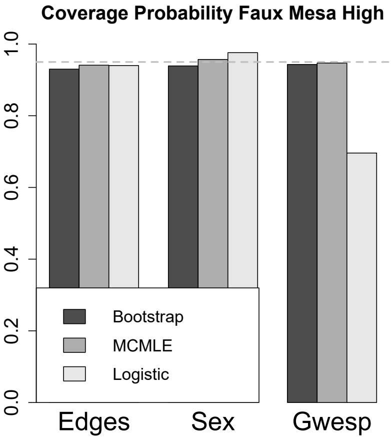
We introduce the **bootstrapped MPLE** in order to obtain more reliable confidence intervals:

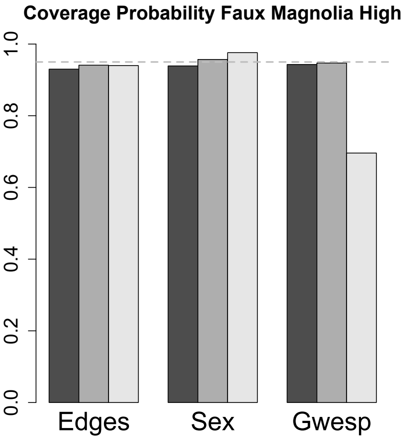
1. Estimate the MPLE
2. Simulate a reasonable number of networks (e.g. 500)
3. Estimate the MPLE for each simulated network
4. Take the 2.5th and 97.5th percentile to obtain a 95% bootstrap confidence interval

Bootstrapped MPLE vs. MCMLE: Coverage Probability

Treatment: MCMLE as 'true' coefficients and simulate 1000 networks.

Calculate CI for each simulated network using bootstrapped MPLE, MCMLE and MPLE and report percentage the CI contains the 'true' value

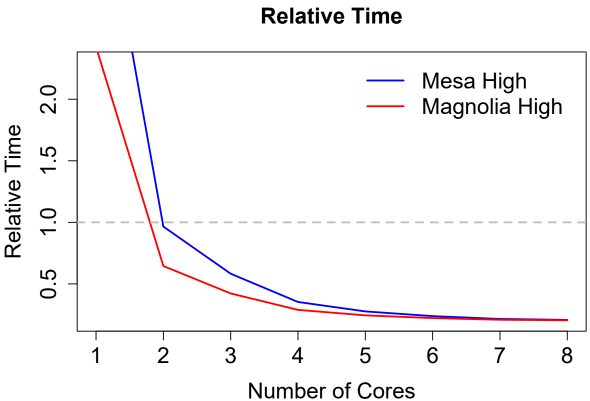




Bootstrapped MPLE vs. MCMLE: Computation Time

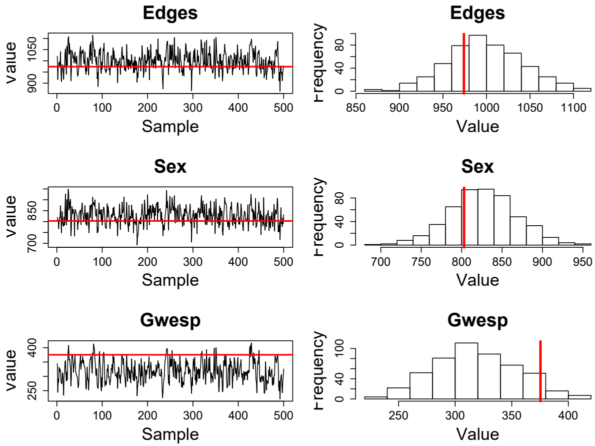
- The bootstrapped MPLE is embarrassingly parallel.
- By using multiple cores the computation time can be further reduced.

This plot visualizes the time ratio of the bootstrapped MPLE to the MCMLE.



Assessing Degeneracy

Degeneracy occurs if the stochastic process generated by the MCMC-algorithm does not hold through the model's defined distribution as stationary distribution.



Conclusion

	Pros	Cons
MCMLE	• approximately exact • can assess degeneracy	• Computationally expensive • θ_0 has to be picked close to θ
MPLE	• simple and fast • consistent estimator	• underestimates st. error
b. MPLE	• simple and fast • can assess degeneracy • consistent estimator • reasonable CIs	• slower than MPLE

Thank you for your attention!