Small network statistics for the network science of teams[1]

George G. Vega Yon, MS Kayla de la Haye, PhD

NetSciX 2019, SCL
January 3, 2019

[1]Contact: vegayon@usc.edu. We thank members of our MURI research team, USC's Center for Applied Network Analysis, Andrew Slaughter, and attendees of the NASN 2018 conference for their comments.

Funding Acknowledgement

This material is based upon work support by, or in part by, the U.S. Army Research Laboratory and the U.S. Army Research Office under grant number W911NF-15-1-0577

Computation for the work described in this paper was supported by the University of Southern California's Center for High-Performance Computing (hpc.usc.edu).

Network Science of Teams
a Multidisciplinary University Research Initiative

UC SANTA BARBARA
USC University of Southern California
MIT Massachusetts Institute of Technology
NORTHWESTERN UNIVERSITY

Context: A tale about social abilities and team performance

Recruited

Context: A tale about social abilities and team performance

Recruited

- 42 mixed gender groups of 3 to 5 participants (unknown)
- Eligibility: (1) 18+ years, (2) Native English speaker

Context: A tale about social abilities and team performance

Recruited

- 42 mixed gender groups of 3 to 5 participants (unknown)
- Eligibility: (1) 18+ years, (2) Native English speaker

2 hour group session

Context: A tale about social abilities and team performance

Recruited

- 42 mixed gender groups of 3 to 5 participants (unknown)
- Eligibility: (1) 18+ years, (2) Native English speaker

2 hour group session

- Group tasks (2 sets of tasks x 30 minutes each)
- Measure group social networks and individual social intelligence (SI)

Context: A tale about social abilities and team performance

Recruited

- 42 mixed gender groups of 3 to 5 participants (unknown)
- Eligibility: (1) 18+ years, (2) Native English speaker

2 hour group session

- Group tasks (2 sets of tasks x 30 minutes each)
- Measure group social networks and individual social intelligence (SI)

Study motivation

Context: A tale about social abilities and team performance

Recruited

- 42 mixed gender groups of 3 to 5 participants (unknown)
- Eligibility: (1) 18+ years, (2) Native English speaker

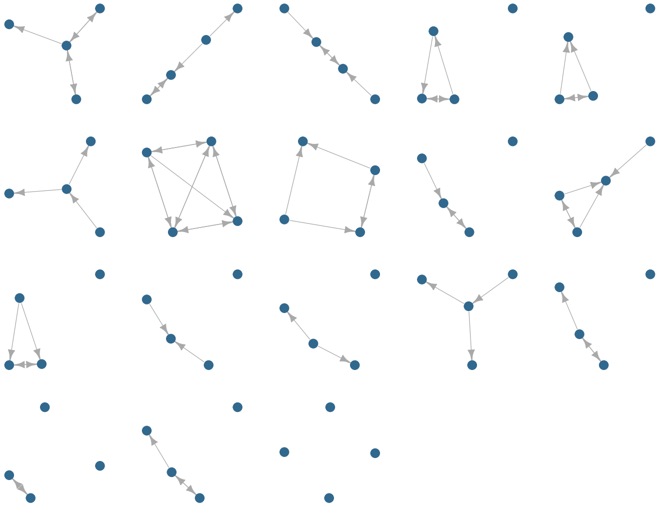
2 hour group session

- Group tasks (2 sets of tasks x 30 minutes each)
- Measure group social networks and individual social intelligence (SI)

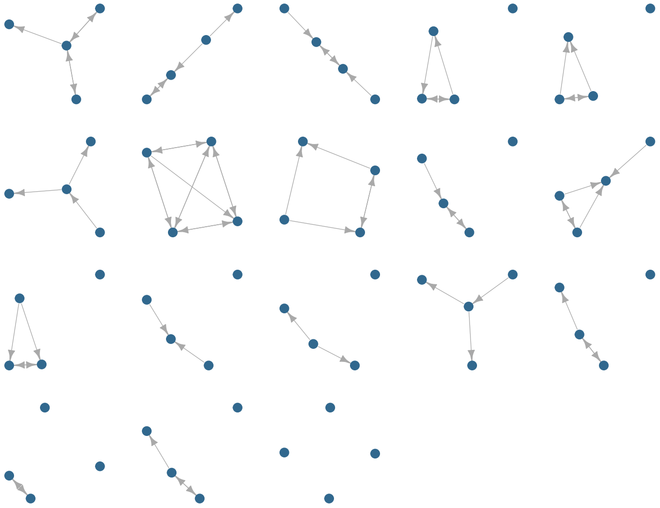
Study motivation

- Overall, a very limited set of SI domains have been tested as predictors of social networks
- Very little research on the emergence of networks in teams.

Context (cont'd)



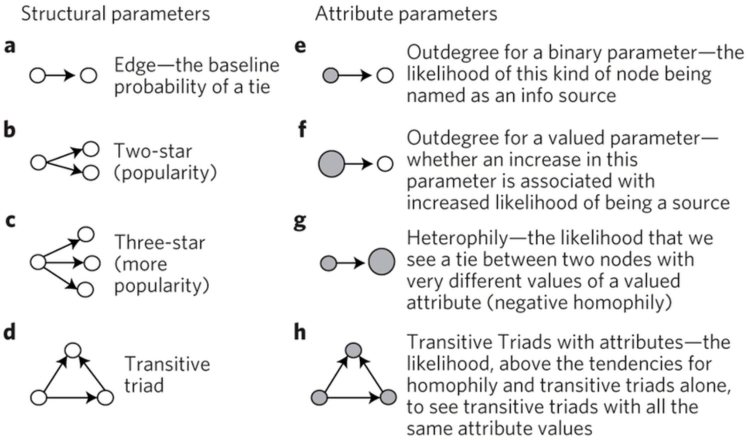
Context (cont'd)



How can we go beyond descriptive statistics?

Small networks and Exponential Random Graph Models

Exponential Random Graph Models: What are the structures that give origin to a given observed graph?



(In general, ties are not IID, moreover, the entire graph is a single observation.)

Small networks and Exponential Random Graph Models (Cont'd)

- Estimating ERGMs is a complex problem.

Small networks and Exponential Random Graph Models (Cont'd)

- Estimating ERGMs is a complex problem. The likelihood function has $2^{n(n-1)}$ terms, which means that for a graph of size 6 we have about 1 billion terms to compute.

Small networks and Exponential Random Graph Models (Cont'd)

- Estimating ERGMs is a complex problem. The likelihood function has $2^{n(n-1)}$ terms, which means that for a graph of size 6 we have about 1 billion terms to compute.
- Current approaches to estimate ERGMs rely on simulation methods, for example MCMC-MLE

Small networks and Exponential Random Graph Models (Cont'd)

- Estimating ERGMs is a complex problem. The likelihood function has $2^{n(n-1)}$ terms, which means that for a graph of size 6 we have about 1 billion terms to compute.
- Current approaches to estimate ERGMs rely on simulation methods, for example MCMC-MLE

When trying to estimate ERGMs in little networks

Small networks and Exponential Random Graph Models (Cont'd)

- Estimating ERGMs is a complex problem. The likelihood function has $2^{n(n-1)}$ terms, which means that for a graph of size 6 we have about 1 billion terms to compute.
- Current approaches to estimate ERGMs rely on simulation methods, for example MCMC-MLE

When trying to estimate ERGMs in little networks

- MCMC fails to converge when trying to estimate a block diagonal (structural zeros) model,

Small networks and Exponential Random Graph Models (Cont'd)

- Estimating ERGMs is a complex problem. The likelihood function has $2^{n(n-1)}$ terms, which means that for a graph of size 6 we have about 1 billion terms to compute.
- Current approaches to estimate ERGMs rely on simulation methods, for example MCMC-MLE

When trying to estimate ERGMs in little networks

- MCMC fails to converge when trying to estimate a block diagonal (structural zeros) model,
- Same happens when trying to estimate an ERGM for a single (little) graph,

Small networks and Exponential Random Graph Models (Cont'd)

- Estimating ERGMs is a complex problem. The likelihood function has $2^{n(n-1)}$ terms, which means that for a graph of size 6 we have about 1 billion terms to compute.
- Current approaches to estimate ERGMs rely on simulation methods, for example MCMC-MLE

When trying to estimate ERGMs in little networks

- MCMC fails to converge when trying to estimate a block diagonal (structural zeros) model,
- Same happens when trying to estimate an ERGM for a single (little) graph,
- Even if it converges, model degeneracy, i.e. bad fit, arises too often.

Rethinking the problem

Rethinking the problem

- 1st Step: Forget about MCMC-MLE estimation, take advantage of small sample and use exact statistic for MLEs:

Rethinking the problem

- 1st Step: Forget about MCMC-MLE estimation, take advantage of small sample and use exact statistic for MLEs:

$$\Pr\left(\mathbf{Y}=\mathbf{y}|\theta,\mathcal{Y}\right)=\frac{\exp\theta^{\mathsf{T}}\mathbf{g}(\mathbf{y})}{\kappa\left(\theta,\mathcal{Y}\right)},\quad \mathbf{y}\in\mathcal{Y}$$

  Where $\mathbf{g}(\mathbf{y})$ is a vector of sufficient statistics, $\theta\in\Theta$ a vector of model parameters, and $\kappa\left(\theta,\mathcal{Y}\right)$ is the normalizing constant (a summation with $2^{n(n-1)}$ terms)

- This solves the problem of been able to estimate a small ergm.

Rethinking the problem

- 1st Step: Forget about MCMC-MLE estimation, take advantage of small sample and use exact statistic for MLEs:

$$\Pr\left(\mathbf{Y}=\mathbf{y}|\theta,\mathcal{Y}\right)=\frac{\exp\theta^{\mathsf{T}}\mathbf{g}(\mathbf{y})}{\kappa\left(\theta,\mathcal{Y}\right)},\quad \mathbf{y}\in\mathcal{Y}$$

  Where $\mathbf{g}(\mathbf{y})$ is a vector of sufficient statistics, $\theta\in\Theta$ a vector of model parameters, and $\kappa\left(\theta,\mathcal{Y}\right)$ is the normalizing constant (a summation with $2^{n(n-1)}$ terms)

- This solves the problem of been able to estimate a small ergm.
- For this we started working on the `lergm` R package (available at https://github.com/muriteams/lergm):

Example 1

Let's start by trying to estimate an ERGM for a single graph of size 4

```
library(lergm)
set.seed(12)
x <- sna::rgraph(4)
lergm(x ~ edges + balance + mutual)
```

```
##
## Little ERGM estimates
##
##  Coefficients:
##    edges  balance   mutual
## -1.9443  -0.2417   3.4961
```

- Cool, we are able to estimate ERGMs for little networks! (we actually call them ~~lergms~~ ERGMitos[2]),

[2] Thanks to George Barnett for suggesting the name!

- Cool, we are able to estimate ERGMs for little networks! (we actually call them ~~lergms~~ ERGMitos[2]),
- Going directly to MLE, we avoid the degeneracy problem.

[2] Thanks to George Barnett for suggesting the name!

- Cool, we are able to estimate ERGMs for little networks! (we actually call them ~~lergms~~ ERGMitos[2]),
- Going directly to MLE, we avoid the degeneracy problem.
- Moreover, due to the size of the networks, we can actually go further and estimate pooled ERGMs

[2]Thanks to George Barnett for suggesting the name!

Solution

- When estimating a block diagonal ERGM we were essentially assuming independence across networks.

Solution

- When estimating a block diagonal ERGM we were essentially assuming independence across networks.
- This means that we can actually do the same with exact statistics approach to calculate a joint likelihood:

Solution

- When estimating a block diagonal ERGM we were essentially assuming independence across networks.
- This means that we can actually do the same with exact statistics approach to calculate a joint likelihood:

$$\Pr\left(\mathbf{Y} = \{\mathbf{y}_i\} | \theta, \{\mathcal{Y}_i\}\right) = \prod_i \frac{\exp \theta^{\mathsf{T}} \mathbf{g}(\mathbf{y}_i)}{\kappa_i\left(\theta, \mathcal{Y}_i\right)}$$

Solution

- When estimating a block diagonal ERGM we were essentially assuming independence across networks.
- This means that we can actually do the same with exact statistics approach to calculate a joint likelihood:

$$
\Pr\left(\mathbf{Y} = \{\mathbf{y}_i\} | \theta, \{\mathcal{Y}_i\}\right) = \prod_i \frac{\exp \theta^{\mathsf{T}} \mathbf{g}(\mathbf{y}_i)}{\kappa_i\left(\theta, \mathcal{Y}_i\right)}
$$

- By estimating a pooled version of the ERGM we can increase the power of our MLEs.

Solution

- When estimating a block diagonal ERGM we were essentially assuming independence across networks.
- This means that we can actually do the same with exact statistics approach to calculate a joint likelihood:

$$
\Pr\left(\mathbf{Y} = \{\mathbf{y}_i\} | \theta, \{\mathcal{Y}_i\}\right) = \prod_i \frac{\exp \theta^{\mathsf{T}} \mathbf{g}(\mathbf{y}_i)}{\kappa_i\left(\theta, \mathcal{Y}_i\right)}
$$

- By estimating a pooled version of the ERGM we can increase the power of our MLEs.
- We implemented this in the `lergm` package

Example 2

Suppose that we have 3 little graphs of sizes 4, 5, and 5:

```r
library(lergm)
set.seed(12)
x1 <- sna::rgraph(4)
x2 <- sna::rgraph(5)
x3 <- sna::rgraph(5)

lergm(list(x1, x2, x3) ~ edges + balance + mutual)
```

```
##
## Little ERGM estimates
##
##  Coefficients:
##    edges  balance   mutual
## -0.3941  -0.2085   1.4156
```

Simulation study

Scenario A

1. Draw parameters for edges and mutual from a uniform(-3, 3).
2. Using those parameters, sampled $n \sim \text{Poisson}(30)$ networks of size 4
3. Estimated the pooled ERGMs using both the MLE and the bootstrap version.

Simulation study

Scenario A

1. Draw parameters for edges and mutual from a uniform(-3, 3).
2. Using those parameters, sampled $n \sim \text{Poisson}(30)$ networks of size 4
3. Estimated the pooled ERGMs using both the MLE and the bootstrap version.

Scenario B

1. Idem.
2. Using those parameters, sampled $n_1 \sim \text{Poisson}(15), n_2 \sim \text{Poisson}(15)$ networks of size 3 and 4 respectively.
3. Idem.

Simulation study

Scenario A

1. Draw parameters for edges and mutual from a uniform(-3, 3).
2. Using those parameters, sampled $n \sim \text{Poisson}(30)$ networks of size 4
3. Estimated the pooled ERGMs using both the MLE and the bootstrap version.

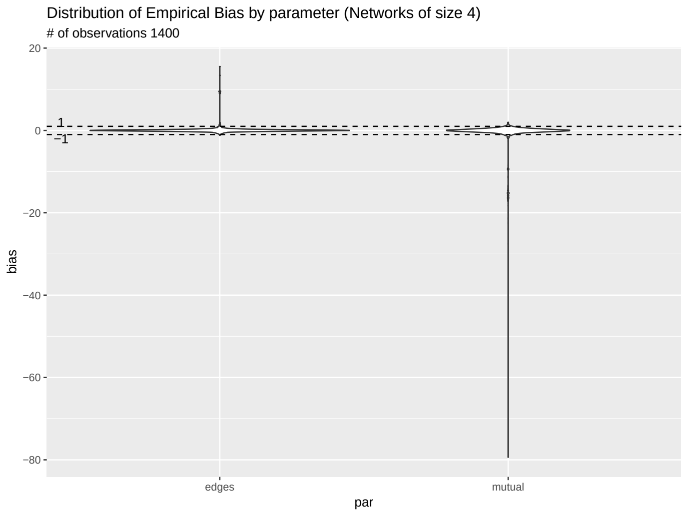
Scenario B

1. Idem.
2. Using those parameters, sampled $n_1 \sim \text{Poisson}(15), n_2 \sim \text{Poisson}(15)$ networks of size 3 and 4 respectively.
3. Idem.

(If anyone asks, I just ran about 3 million ERGMs... :))

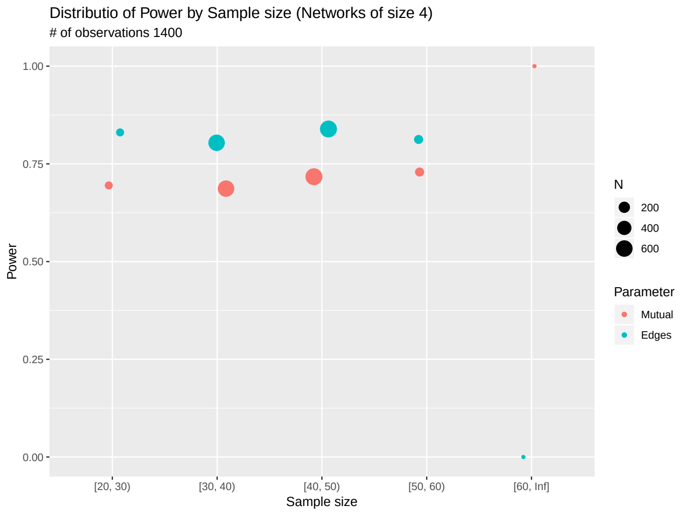
Simulation study: Scenario A

Empirical Bias



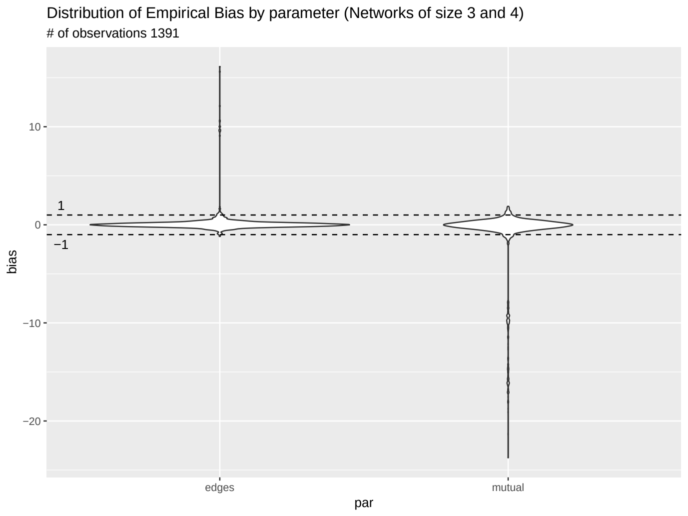
Simulation study: Scenario A

Power



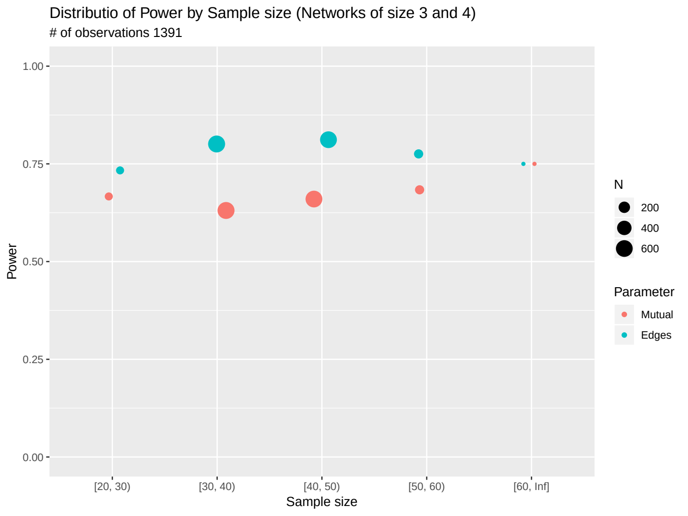
Simulation study: Scenario B

Empirical Bias

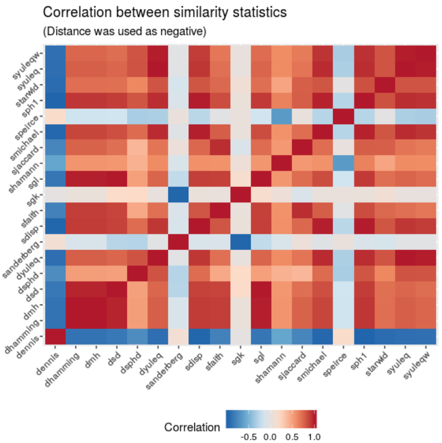


Simulation study: Scenario B

Power



Other approaches



Discussion

- First set of results from the simulation study are encouraging

Discussion

- First set of results from the simulation study are encouraging
- Need to conduct more simulations using nodal attributes and networks of size 5 (right now having problems when building the DGP).

Discussion

- First set of results from the simulation study are encouraging
- Need to conduct more simulations using nodal attributes and networks of size 5 (right now having problems when building the DGP).
- Small structures imply a smaller pool of parameters (which is OK), but can be more useful when including nodal attributes.

Discussion

- First set of results from the simulation study are encouraging
- Need to conduct more simulations using nodal attributes and networks of size 5 (right now having problems when building the DGP).
- Small structures imply a smaller pool of parameters (which is OK), but can be more useful when including nodal attributes.
- When estimating the pooled version, we are essentially hand-waving the fact that parameter estimates implicitly encode size of the graph, i.e.

  > *Does a the estimate of `edge = 0.1` has the same meaning for a network of size 3 to a size 5? (but perhaps is not such a big deal)*

Discussion

- First set of results from the simulation study are encouraging
- Need to conduct more simulations using nodal attributes and networks of size 5 (right now having problems when building the DGP).
- Small structures imply a smaller pool of parameters (which is OK), but can be more useful when including nodal attributes.
- When estimating the pooled version, we are essentially hand-waving the fact that parameter estimates implicitly encode size of the graph, i.e.

  > *Does a the estimate of `edge = 0.1` has the same meaning for a network of size 3 to a size 5? (but perhaps is not such a big deal)*

- Finally, this work can be extended to other types of small networks, including: families, ego-nets, etc. And other methods, such as TERGMs.

Thank you!

Small network statistics for the network science of teams[3]

George G. Vega Yon, MS   Kayla de la Haye, PhD

NetSciX 2019, SCL
January 3, 2019

[3]Contact: vegayon@usc.edu. We thank members of our MURI research team, USC's Center for Applied Network Analysis, Andrew Slaughter, and attendees of the NASN 2018 conference for their comments.

What have we got so far?

```
lergm(networks ~ mutual + edges + triangle + nodematch("male") +
    diff("Empathy") + nodematch("nonwhite"))
```

Table 1: Preliminary results with our small teams data. The table shows 95% confidence intervals for the parameter estimates using the pooled ERGM model.

	All (42)		All but size 3 (35)	
	2.5 %	97.5 %	2.5 %	97.5 %
mutual	-0.40	0.55	-0.45	0.55
edges	-0.91	-0.16	-1.04	-0.29
triangle	0.06	0.24	0.09	0.27
nodematch("male")	-0.36	0.31	-0.34	0.36
diff("Empathy")	0.12	0.59	0.09	0.58
nodematch("nonwhite")	-0.26	0.37	-0.29	0.35