MPNet Tutorial

Introducing MPNet

MPNet is designed for the simulation and estimation of ERGMs and ALAAMs for multilevel networks which consists of two within-level one-mode network and a two-mode meso-level network. As multilevel ERGM/ALAAM are extensions and combinations of one- and two-mode ERGM/ALAAM, MPNet can be used for modelling single or combinations of one- or two-mode networks.

MPNet is a program written at Swinburne University of Technology and the University of Melbourne (Wang, Robins, Pattison, & Koskinen, 2018) for the simulation and estimation of exponential random graph models for social networks. The simulation procedure, based on the Metropolis-Hastings algorithm, is described in Robins, Pattison & Woolcock (2005). For estimations, it follows the algorithm proposed by Snijders (2002) which was firstly implemented in the SIENA program (Snijders et al., 2007).

Using MPNet, we can simulate a distribution of graphs or outcome attribute values following a specified ERGM/ALAAM. MPNet can simulate a distribution of graphs, from which we can take a sample of graphs to examine.

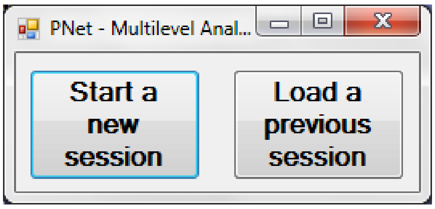
Session Folder in MPNet

NB: Before using MPNet you need to create a folder (called a Session Folder) for all of the analysis inputs and outputs. Do this now (name it “Exercise7” for example). When you are estimating models, it is this session folder that your data should be stored in.

“.pnet” File

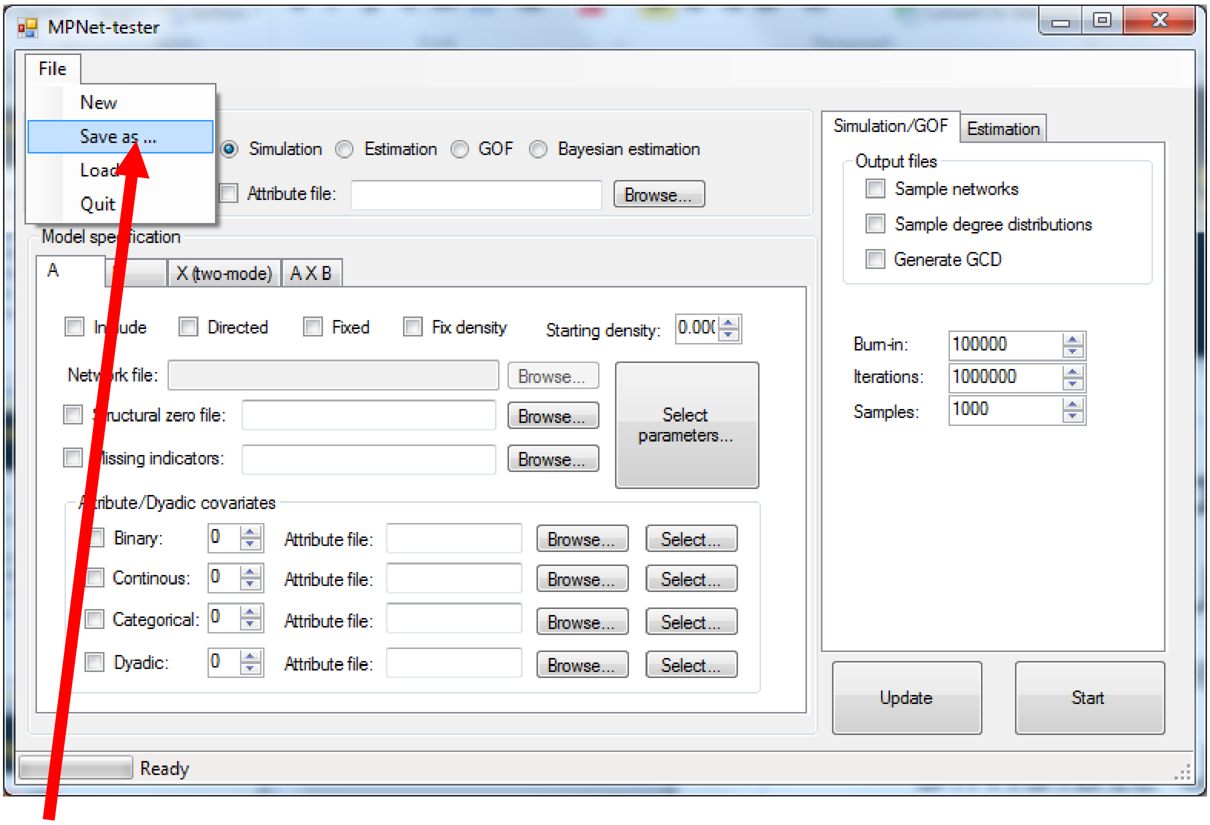
You must also specify a name for the MPNet session which will be written to a session file with “.pnet” as extension name.

You can start MPNet with a new “.pnet” file, or load a previous session using an existing “.pnet” file. We note that you may wish to run a number of different models for the same data set, and by using different “.pnet” session file names you can save the details of these different models for further analysis.



Once a session file is created, the MPNet main user interface will show up. The user interface shows up to three networks (that constitutes a multilevel network).

Network A and B are one-mode networks, and network X is a two-mode (cross-level) network. You may include one or more networks in a simulation, estimation or goodness of fit session.



● Simulation

Starting density: 0.00(

□ Binary: 0

□ Continous: 0

□ Categorical: 0

□ Dyadic: 0

Burn-in: 100000

Iterations: 1000000

Samples: 1000

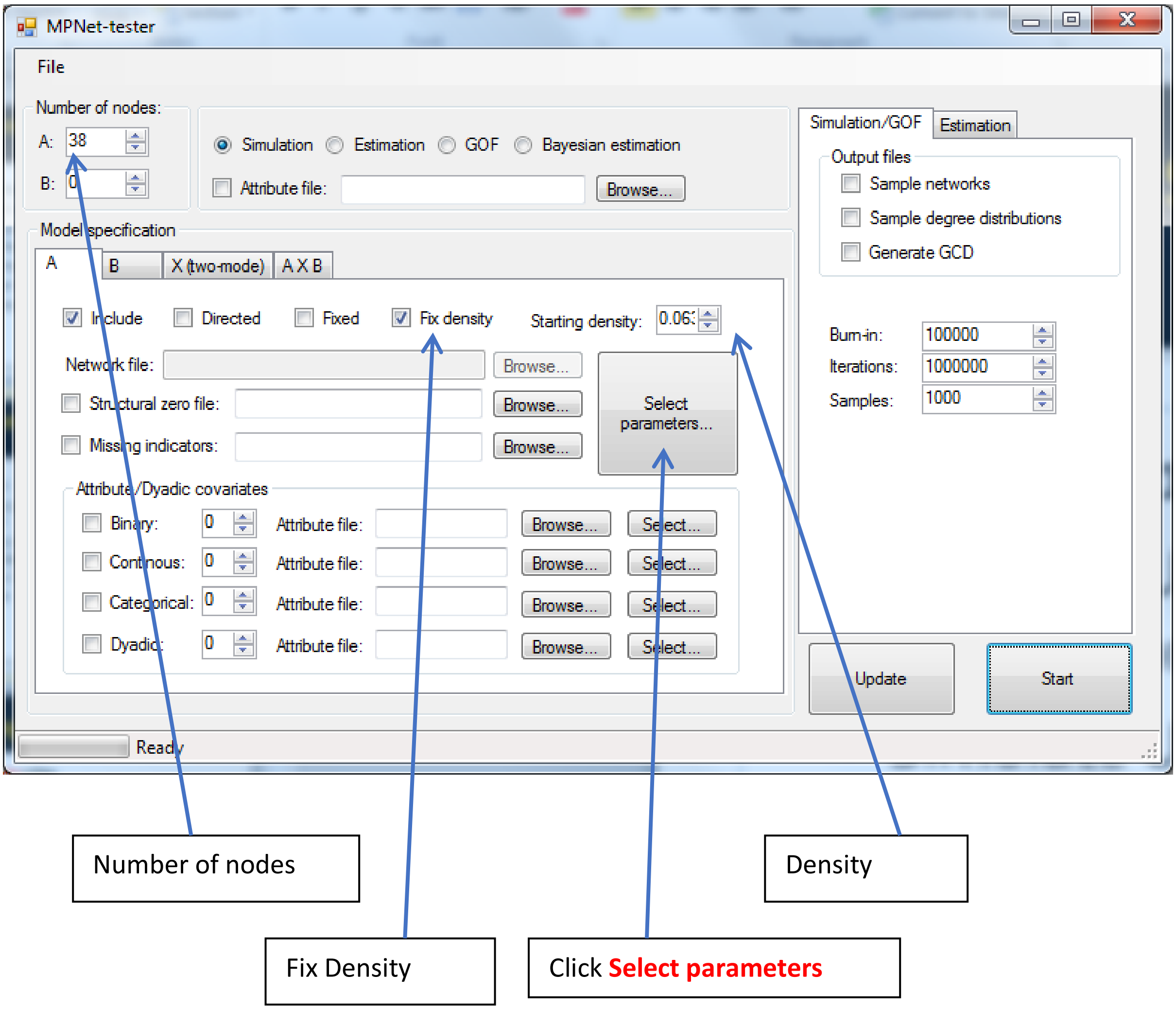
NB: If you want to save a different version of the session file then choose “Save as”

Generating Bernoulli random graphs in MPNet

Using MPNet software

Open MPNet by clicking on the MPNet.exe file. We are using the simulation function for this exercise, so make sure you click the ‘simulation’ option at the top of the program.

Set up MPNet to simulate a distribution of graphs of 38 nodes, with fixed density of 0.06259 (the number of actors and density of the “comm_undirected.txt” network of The Corporation – the example dataset from Lusher, Koskinen & Robins, 2013).



MPNet-tester

Number of nodes: A: 38 B: 0

● Simulation ○ Estimation ○ GOF ○ Bayesian estimation

☐ Attribute file:

Model specification

☑ Include ☐ Directed ☐ Fixed ☑ Fix density Starting density: 0.063

Network file:

☐ Structural zero file:

☐ Missing indicators:

Attribute/Dyadic covariates

☐ Binary: 0 Attribute file:

☐ Continous: 0 Attribute file:

☐ Categorical: 0 Attribute file:

☐ Dyadic: 0 Attribute file:

Output files

☐ Sample networks

☐ Sample degree distributions

☐ Generate GCD

Burn-in: 100000

Iterations: 1000000

Samples: 1000

Ready

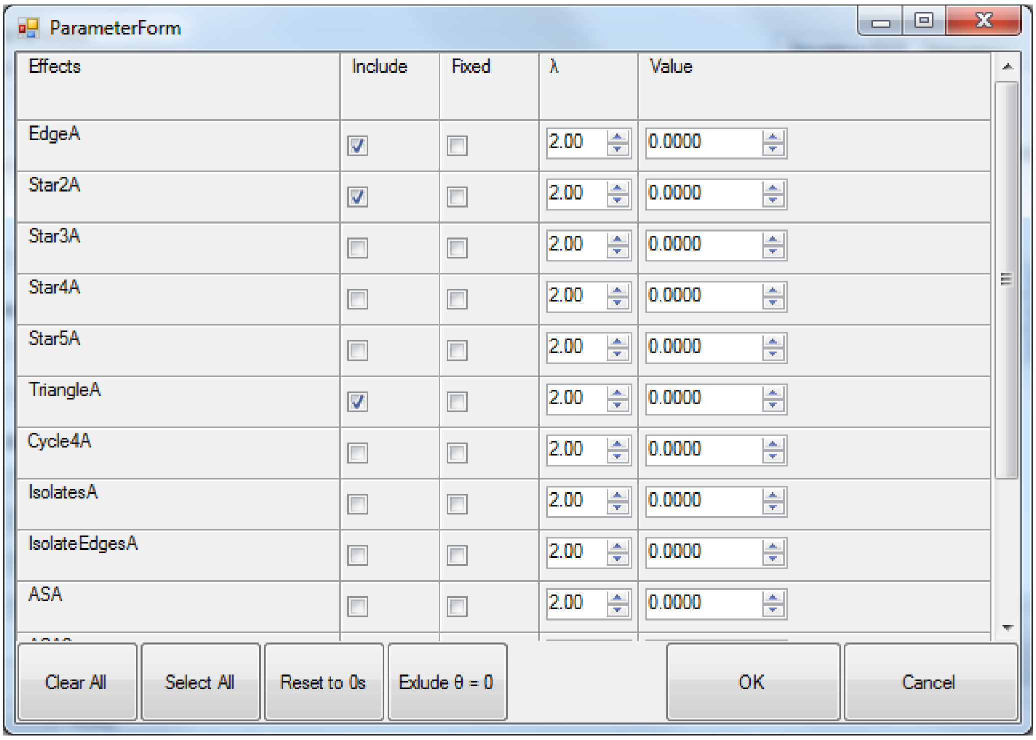
Number of nodes

Density

Fix Density

Click **Select parameters**

Clicking the Select Parameter button opens a new window that permits you to select parameters for the model and parameter values. In this case, we won't have any parameters (other than the fixed density of 0.06259), but by selecting parameters and leaving the parameter value at 0, MPNet will count the relevant configurations in the simulated graphs. In this case, select edge, 2-star and triangle. Then click OK.



ParameterForm

Effects	Include	Fixed	λ	Value
EdgeA	☑	□	2.00	0.0000
Star2A	☑	□	2.00	0.0000
Star3A	□	□	2.00	0.0000
Star4A	□	□	2.00	0.0000
Star5A	□	□	2.00	0.0000
TriangleA	☑	□	2.00	0.0000
Cycle4A	□	□	2.00	0.0000
IsolatesA	□	□	2.00	0.0000
IsolateEdgesA	□	□	2.00	0.0000
ASA	□	□	2.00	0.0000

Back on the main window, you are now set up to simulate a number of 38 node graphs, each of which will have a density of 0.06259. Notice that there is a "burn-in" of 100,000 (a preliminary simulation before the final simulation starts), you will simulate 1,000,000 graphs ("number of iterations"), but the program will record results for 1,000 graphs ("Number of samples ..." – that is every 1,000th graph in the simulation.) Click on Start!

"session name_sim.txt.txt" files

There are a number of output files in the Session folder. The most important is the "simulation-session name.txt" file, which contains the statistics for all the 1000 graphs in your sample from the graph distribution. It lists the number of the graph ('id'), and the number of edges, 2-stars and triangles in each graph. Notice that each graph has the same number of edges, as it must do for a fixed density simulation.

"session name_Network_A_1001000.txt" files

The output file "session name_Network_M_1001000.txt", or "session name_Network_A_1001000.txt" has details for the very last graph in the simulation. It is in the form to be read into Pajek (another useful SNA visualization program) to produce a network visualization.

What is the average number of triangles in your graphs? Import or copy/paste the "simulation-session name.txt" file into Excel or a statistical package and calculate means and standard deviations. Or, if you have SPSS on your computer, open the "sim_session name.sps" file, which is an SPSS syntax file that will read in this data and produce histograms, with means and standard deviations, and scatterplots of the statistics across the simulation.

Simulating Bernoulli random graph distributions using MPNet

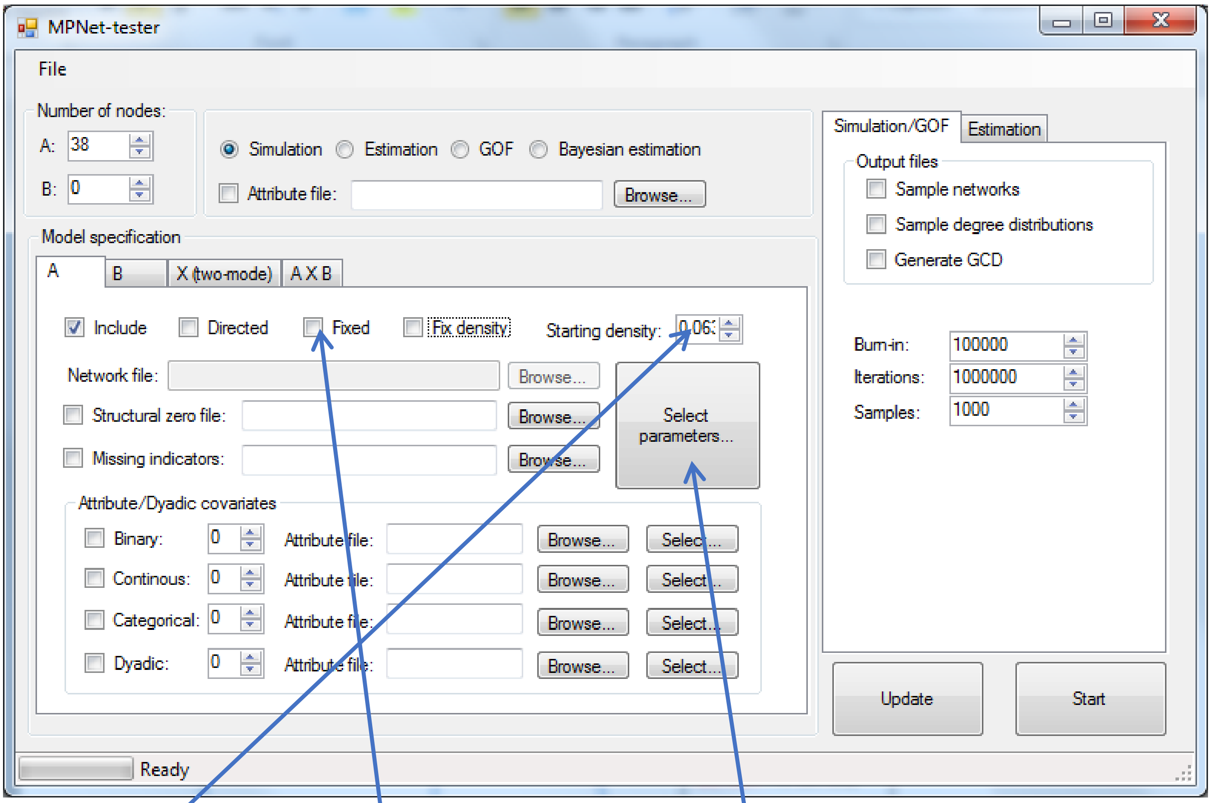
Maybe it is the fixed density that limits the number of triangles. Instead of fixing the density at exactly 0.06259, we could have a model where the chances of a tie occurring between two nodes was 0.06259. This is the Bernoulli graph distribution. The only non-zero parameter in a Bernoulli graph model is the edge parameter.

The only parameter in a Bernoulli graph distribution (a simple random graph distribution) is an edge parameter (L). Selecting a value for the edge parameter fixes the probability p of an edge between any pair of nodes. The relationship between p and the L parameter is:

$$L \textit{ parameter} = \ln(p/(1-p))$$

It follows from this calculation that an L parameter value of -2.7065 (approx) corresponds to a probability of 0.06259. A distribution of graphs with that probability should have an AVERAGE density of 0.06259.

Start back with MPNet. Give this session a new name (or delete or remove the previous output files from the Session folder). Change the Fix density instruction. The starting graph density is now not important (it will be removed through the burn-in), so you can just leave it.



MPNet-tester

File

Number of nodes:
A: 38
B: 0

● Simulation ○ Estimation ○ GOF ○ Bayesian estimation

□ Attribute file: Browse...

Model specification

A | B | X (two-mode) | A X B

☑ Include □ Directed □ Fixed □ Fix density Starting density: 0.06

Network file: Browse...

□ Structural zero file: Browse...

□ Missing indicators: Browse...

Select parameters...

Attribute/Dyadic covariates

□ Binary: 0 Attribute file: Browse... Select...

□ Continous: 0 Attribute file: Browse... Select...

□ Categorical: 0 Attribute file: Browse... Select...

□ Dyadic: 0 Attribute file: Browse... Select...

Simulation/GOF | Estimation

Output files
□ Sample networks
□ Sample degree distributions
□ Generate GCD

Burn-in: 100000
Iterations: 1000000
Samples: 1000

Update Start

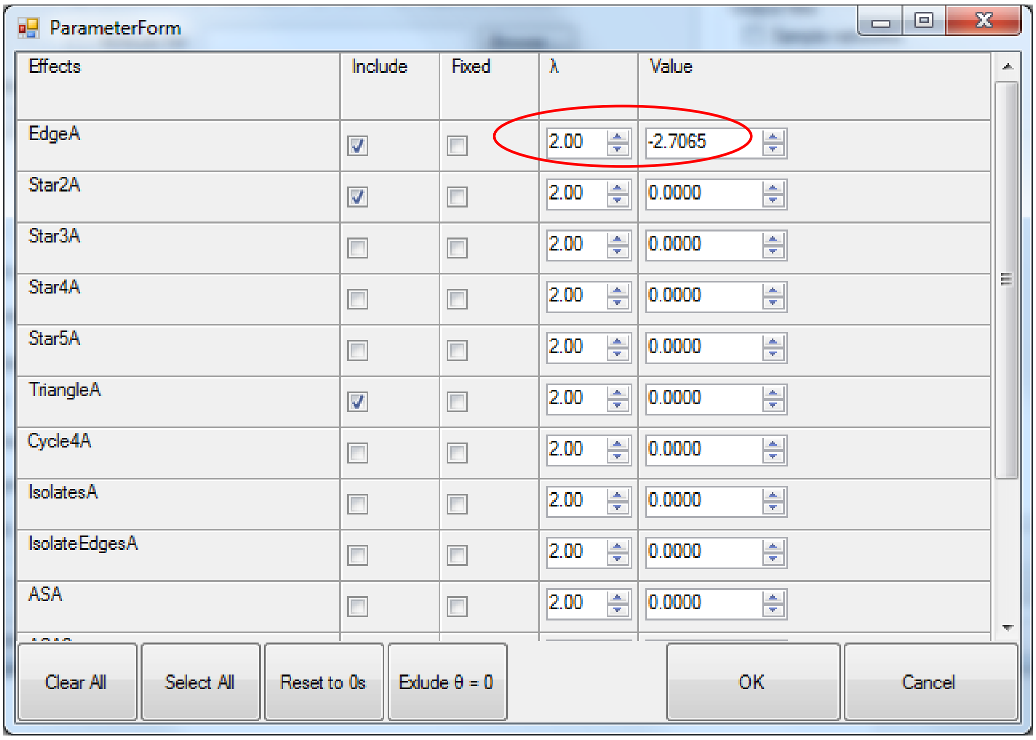
Ready

This doesn't matter now

DO NOT Fix Density

Click **Select parameters**

When the structural parameter window appears, enter -2.7065 as the edge parameter value. Then run the simulation as before.



Effects	Include	Fixed	λ	Value
EdgeA	☑	□	2.00	-2.7065
Star2A	☑	□	2.00	0.0000
Star3A	□	□	2.00	0.0000
Star4A	□	□	2.00	0.0000
Star5A	□	□	2.00	0.0000
TriangleA	☑	□	2.00	0.0000
Cycle4A	□	□	2.00	0.0000
IsolatesA	□	□	2.00	0.0000
IsolateEdgesA	□	□	2.00	0.0000
ASA	□	□	2.00	0.0000

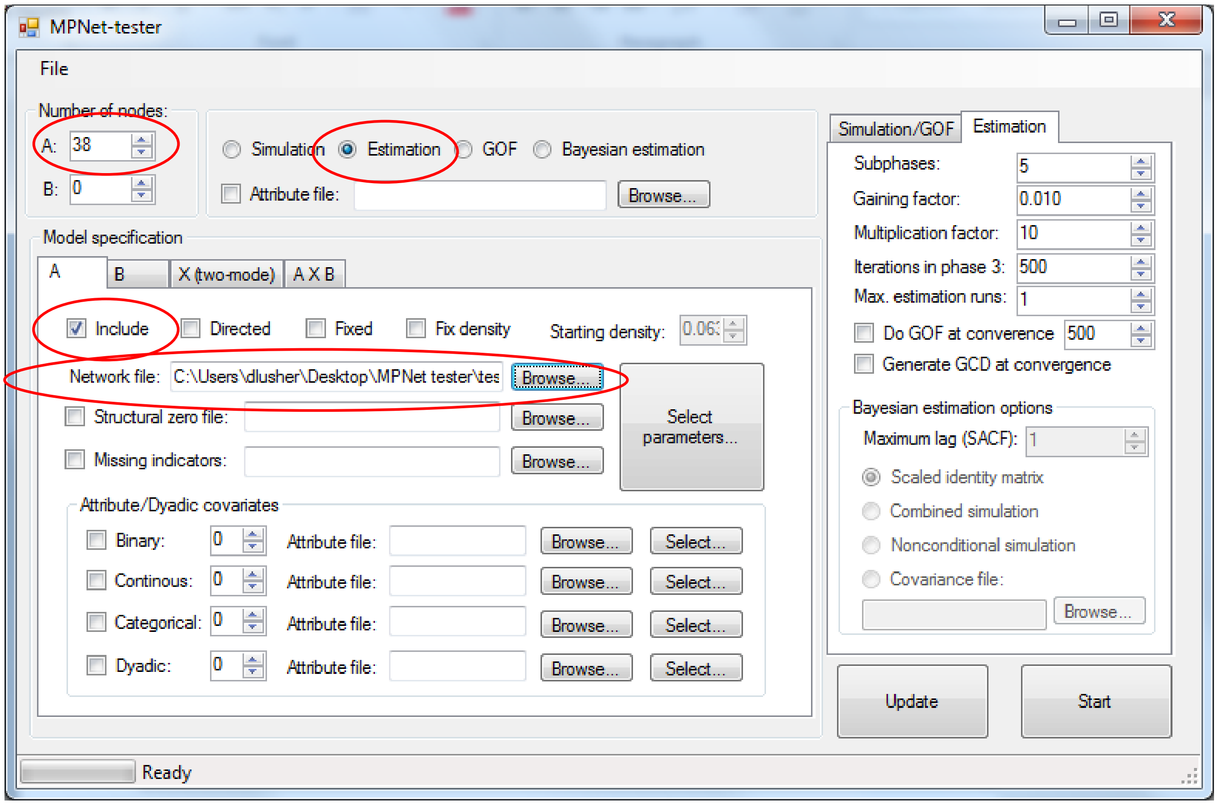
Then run the simulation as before. Notice now when the graph density is not fixed that the number of edges varies from graph to graph.

Enter the data from “simulation-session name.txt” data into SPSS using the syntax file, or into Excel or another statistical package.

Estimating Bernoulli and Markov ERGMs

To estimate parameters, select the Estimation tab in MPNet. Make sure that the datafile (in this case, The Corporation data – comm_undirected.txt) is in the session folder, enter the number of actors (38) and the name of the Network datafile (using Browse...). Select structural parameters.

Network data: *comm_undirected.txt*



MPNet-tester

File

Number of nodes:

A: 38

B: 0

○ Simulation ● Estimation ○ GOF ○ Bayesian estimation

□ Attribute file: Browse...

Model specification

A | B | X (two-mode) | A X B

☑ Include □ Directed □ Fixed □ Fix density Starting density: 0.06:

Network file: C:\Users\dlusher\Desktop\MPNet tester\tes Browse...

□ Structural zero file: Browse...

□ Missing indicators: Browse...

Select parameters...

Attribute/Dyadic covariates

□ Binary: 0 Attribute file: Browse... Select...

□ Continous: 0 Attribute file: Browse... Select...

□ Categorical: 0 Attribute file: Browse... Select...

□ Dyadic: 0 Attribute file: Browse... Select...

Simulation/GOF | Estimation

Subphases: 5

Gaining factor: 0.010

Multiplication factor: 10

Iterations in phase 3: 500

Max. estimation runs: 1

□ Do GOF at converence 500

□ Generate GCD at convergence

Bayesian estimation options

Maximum lag (SACF): 1

● Scaled identity matrix

○ Combined simulation

○ Nonconditional simulation

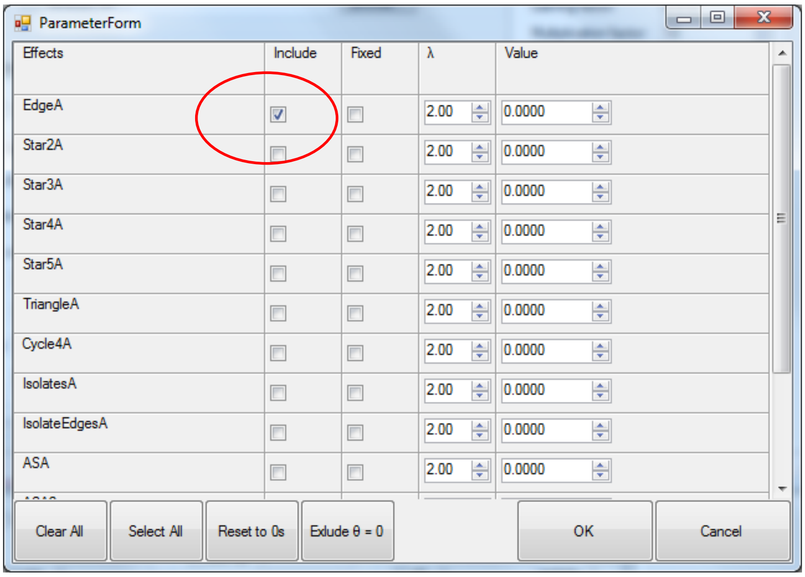
○ Covariance file: Browse...

Update Start

Ready

Estimating a Bernoulli model

The only parameter that is to be estimated in a Bernoulli model is the Edge parameter. Before in E~~e~~xercise 1, when we simulated a Bernoulli model, we selected star and triangle parameters under the simulation tab but kept their values at 0, so that MPNet would count the numbers of triangles and 2-stars. When estimating here, however, we ONLY select the Edge parameter. If we selected the star and triangle parameters, they would be estimated as well, and it would no longer be a Bernoulli model (it would be a Markov model – later.)



ParameterForm

Effects	Include	Fixed	λ	Value
EdgeA	☑	□	2.00	0.0000
Star2A	□	□	2.00	0.0000
Star3A	□	□	2.00	0.0000
Star4A	□	□	2.00	0.0000
Star5A	□	□	2.00	0.0000
TriangleA	□	□	2.00	0.0000
Cycle4A	□	□	2.00	0.0000
IsolatesA	□	□	2.00	0.0000
IsolateEdgesA	□	□	2.00	0.0000
ASA	□	□	2.00	0.0000

Clear All | Select All | Reset to 0s | Exlude θ = 0 | OK | Cancel

Go back to the main window and click Start! to commence estimation.

At the bottom of the output, you will have something like:

```
Effects Lambda     Parameter     Stderr       t-ratio       SACF
EdgeA  2.0000      -2.6935       0.152        -0.009        0.278   *
```

- Effects: The parameter name
- Lambda: More on this later.
- Parameter: The parameter estimate
- Stderr (Estimated standard error): Each parameter estimate has a standard error (SE) which is a measure of the precision or how certain we are of the parameter estimate. A small SE indicates greater precision and certainty, while a large SE indicates less certainty.
- t-ratio (the convergence statistic: a measure of how stable the parameter estimates are (NOT whether a parameter is significant – see below). For a quick convergence check, you want the convergence t-statistic to be less than 0.1 (in absolute value).

  If the convergence statistic is not less than 0.1, you can repeat the estimation but starting this time from the last finishing estimate. Go back to the MPNet window, click Update and check what has happened to the starting parameter value by clicking on Select parameters... Then click Start again. The new estimate will again appear at the bottom of the output file.

- SACF (sample autocorrelation function)
- Asterisk: indicates a significant effect (Absolute value of the estimate is more than twice the standard error.)

Estimating a Markov model

NB: Markov models are not usually successful models either, except perhaps for small datasets like this one. It is presented here only for illustrative purposes.

Select parameters again, and now select Edge, 2-star, 3-star and Triangle parameters. Run the estimation. Check at the bottom of the output. If any convergence statistic is not less than 0.1, run the estimation again - after Updating! - until you have convergence FOR ALL FOUR PARAMETERS. The Update! process sets the previous unconverged estimates as the starting point for a new estimation run.

If you have trouble getting a converged model, then try increasing the Multiplication Factor (under the Estimation Tab to the right of the MPNet window) to 30.

You should have results like:

```
Effects         Lambda          Parameter       Stderr  t-ratio  SACF
EdgeA           2.0000          -3.1643         0.737   -0.027  0.249    *
```

Star2A	2.0000	0.1747	0.281	-0.033	0.440	
Star3A	2.0000	-0.1895	0.132	-0.051	0.579	
TriangleA	2.0000	1.8762	0.249	-0.034	0.696	*

Interpretation: For these results we have significant effects for edge and triangle parameters, but not the 2-star or 3-star parameters.

The negative and significant edge effect indicates that we see few edges unless they are contained within other network structures. That is, there are relatively few isolated edges between dyads in this data – thus where edges occur they are within other network structures.

The positive and significant triangle effect indicates that we see more triangles than we would expect to see by chance, given the other effects in the model.