IEMS 441/COMM ST 525/MTS 525: Social Network Analysis

p\* / Exponential Random Graph Modeling (p\*/ERGM)

The purpose of this exercise is to demonstrate your ability to test hypotheses about a network’s structure using exponential random graph modeling (ERGM) techniques. We will use the program StatNet. The first part of this assignment provides instructions on using StatNet. The second part provides instructions on using StatNet to test hypotheses.

**Installation Instructions for StatNet**

**StatNet Website:** [**http://csde.washington.edu/statnet/**](http://csde.washington.edu/statnet/)

1. R installation

In order to install and use StatNet, you have to install R first (Skip this step if you have already installed R).

Go to: <http://www.r-project.org/>. Click the “download R” link, which will take you to a list of all the mirror sites. Scroll down to USA, we recommend Iowa State University, Ames IA (<http://streaming.stat.iastate.edu/CRAN/>). This will take you to a new screen, where you can choose your operating system. Follow the installation instructions and keep all the defaults.

**Output instructions**

Prepare a report on your work that includes three sections, Method (which states what you did), Results (which tells what you discovered), and Discussion (which indicates what it means, your substantive interpretations), as if it were part of a larger paper being prepared for publication. Follow the guidelines of the Publication Manual of the American Psychological Association (5th ed.) for the three sections. Cut and paste all relevant computer output (tables) as part of the document you use to provide your report. Upload your report to Assignments.

**Analysis instructions**

The purpose of this exercise is to demonstrate your ability to test hypotheses about a network’s structure using exponential random graph modeling (ERGM) techniques.

***Theory & Hypotheses:***

Specifically, this assignment tests two hypotheses and poses one research questions deduced from Theory of Transactive Memory (discussed on pages 198-203 of Monge & Contractor, 2003).

*Hypothesis 1*: Members tend to allocate information to other members, from whom they retrieve information.

*Hypothesis 2*: Members tend to retrieve information from other members with high expertise.

*Research question*: Do members have an independent tendency to retrieve information from other members?

***Data***

We will analyze the communication behaviors within a team of 17 members who were involved in designing military installations.

*CRIeq.txt* is each team member’s communication to retrieve information from other team members on the topic of environmental quality (*eq*). This is a directed, binary relation.

*CAIeq.txt* is each team member’s communication to allocate information to other team members on the topic of environmental quality (*eq*). This is a directed, binary relation.

*EXeq\_cons.txt* is each team member’s expertise on the topic of environmental quality (*eq*) as perceived on average by all team members. This is a continuous attribute.

***Estimation***

The analysis will use the CRIeq.txt as the *network* file, EXeq\_cons.txt as the *attribute* file, and CAIeq.txt as the *covariate network* file.

Special Tips:

* Included with this lab assignment, there is a file named “ERGM\_Lab3.R”. To open this file, start R and choose “File->Open Script” or open it with Notepad++ (<http://notepad-plus-plus.org/>). You will find all commands that will be used in this lab assignment. You can run these commands by selecting them, right-click on these selected command and choose “run line or selection”, or copying and pasting from Notepad++. You can always switch between the script window and R console window to run commands and view results. Another way to run commands is to type or cut and paste these commands directly in “R Console” window. You can choose whichever way you like.
* When you exit R program, you will be asked whether or not you want to “save workspace image”. “No” will clear all objects and variables previously defined and “Yes” will save them so that you can use them when you restart R.

To use Statnet, start R first.

1. StatNet Installation
   1. Open R-Studio
   2. In the command window titled “R Console”, input the following command (these commands are also included in the ERGM\_lab3.r file, you can copy and paste them from there):

install.packages("statnet")

In the pop-up window, choose the mirror site USA(IL) and click “OK”.

Next, install the package “coda”, which is required by this lab assignment. In the command window, type the following command:

install.packages("coda")

Now you have completed the installation of R and Statnet.

* 1. A user’s guide to StatNet can be found here: <http://csde.washington.edu/statnet/users_guide.shtml>

A tutorial is available here: <http://csde.washington.edu/statnet/Online_Users_Guide/v24i08.pdf>

Load statnet package using the following command:

library(statnet)

*Step 1*: Convert txt file to network object.

All data files in this lab assignment are in text form. We need to convert matrices to network objects in R.

* 1. Put all three text files in the same folder and designate your working directory. To find out what your current working directory is:

getwd()

Then set your working directory to be the folder that contains the R script and data files: Session > Set Working Directory > Source File Location

list.files()

* 1. Run the following commands to load networks and attributes for nodes:

A <- matrix(scan("CRIeq.txt", n = 17\*17), 17, 17, byrow = TRUE)

CRIeq <- as.network.matrix(A, matrix.type="adjacency")

att <- read.table("EXeq\_cons.txt", header=T)

att <- as.vector(att)

set.vertex.attribute(CRIeq, "EX", att, v=1:17)

B <- matrix(scan("CAIeq.txt", n = 17\*17), 17, 17, byrow = TRUE)

CAIeq <- as.network.matrix(B, matrix.type="adjacency")

* 1. To get some basic information of these two networks, use the following commands:

summary(CRIeq)

network.size(CRIeq)

plot(CRIeq)

plot(CRIeq,displaylabels=T,boxed.labels=F)

betweenness(CRIeq)

isolates(CRIeq)

summary(CAIeq)

plot(CAIeq,displaylabels=T,boxed.labels=F)

*Step 2*: Build Model

To test our hypotheses and answer the research question, we build the following model:

model <- ergm(CRIeq ~ edges + mutual + gwesp(0.2, fixed=T) + edgecov(CAIeq)+ nodeicov("EX")+nodeocov("EX"))

In this model, we include 6 network statistics.

edges: number of edges in the network

mutual: number of reciprocal links in the network

gwesp: number of links that belongs to at least one triangle structure in the network

edgecov: covariance between edges of two networks

nodeicov: covariance between in degree of nodes and attributes of nodes

nodeocov: covariance between out degree of nodes and attributes of nodes

For more details on each of the parameters, type:

help(‘ergm-terms’)

*Step 3*: View Results

To view the results from the above model, use command:

**summary(model)**

For each statistic, p-value <0.05 indicates significant result. Interpret the results and check if hypotheses are supported. Also, answer the research question based on these results.

*Step 4*: Model Diagnostics

To judge convergence of the MCMC processes in the model, we use the following commands (Before run these command, make sure to close the graphic windows of network visualizations in R. Otherwise, there may be device conflict problem.):

**pdf("modeldiagnostics.pdf")**

**mcmc.diagnostics(model)**

**dev.off()**

These commands will generate a pdf file of model diagnostics located in your designated working directory. Or you can execute only the second command to see the result in R window.

The plots indicate change of model statistics during the last iteration of the MCMC estimation procedure. For each model statistic, the left hand side plot gives the change of the statistic with iterations and the right hand side plot is a histogram of statistic. Both are normalized, so the observed data locate at 0.

Include model diagnostics into your report and interpret the plots you got. Has the MCMC process converged to desired state?

*Step 5*: Goodness of Fit

This step is to check how well the estimated model captures certain features of the observed network, for example triangles in the network.

First, we simulate many networks from the estimated model and get 100 samples from the simulation process. Please note, this is a very computationally intensive process and make take 5 minutes or more to compute.

**sim <- simulate(model, burnin=100000, interval=100000, nsim=100, verbose=T)**

Then, we get the number of triangles from each of the 100 samples:

**model.tridist <- sapply(1:100, function(x) summary(sim[[x]] ~triangle))**

We compare the distribution of triangles in the sampled networks with the observed network:

**hist(model.tridist,xlim=c(20,120))**

**CRIeq.tri <-summary(CRIeq~triangle)**

**arrows(CRIeq.tri,20, CRIeq.tri,5,col="red",lwd=3)**

The last several commands will generate a histogram of the triangles in the sampled network with an arrow indicating the value of observed network. Interpret your result. Is the estimated model a good one in terms of triangle measure? Include your results in your lab report.

We can repeat this goodness-of-fit evaluation process for a variety of other network statistics such as degree distribution, distribution of edgewise shared partners, and the distribution of geodesics. Note that “1e+5” = 105 = 100,000.

**gof<-gof(model~idegree+odegree+espartners+distance,verbose=T,burnin=1e+5,  
interval=1e+5)**

This code simulates networks as we did above, compiles statistics for these simulations as well as the observed network, and calculates p-values like PNET. However, we’ll just plot the distributions of all of the aforementioned values to evaluate the correspondence between the networks simulated by the model and the observed network.

**par(mfrow=c(2,2))**

**plot(gof)**

The resulting graph should have 4 subplots corresponding to the 4 parameters from the gof function above. The dark black line represents the data for the observed network. The boxplots represent the distribution of corresponding degrees across the simulated networks. The soft lines are the 95% confidence intervals. Copy this image and include it in your writeup.

Report the p-values for the simulation:

**gof**

In general, for configurations in the model, the fit is considered good if │t│≤ 0.1. For configurations not included in the model, the fit is considered good if 0.1<│t│≤ 1, and not extreme if 1< │t│≤ 2.

Reminder (same as output instructions at the beginning): Prepare a report on your work that includes three sections, Method (which states what you did), Results (which tells what you discovered), and Discussion (which indicates what it means, your substantive interpretations), as if it were part of a larger paper being prepared for publication. Follow the guidelines of the Publication Manual of the American Psychological Association (5th ed.) for the three sections. Cut and paste all relevant computer output (tables) as part of the document you use to provide your report. Upload your report to Assignments in Canvas.