Exponential Random Graph Models Under Measurement Error

Zoe Rehnberg

Advisor: Dr. Nan Lin

Abstract

Understanding social networks is increasingly important in a world dominated by social media and access to enormous amounts of data. When analyzing social network data, we are often interested in the underlying structures that exist in the network. Exponential random graph models (ERGMs) are frequently used by analysts to gain a better understanding of the formation and operation of these structures. Data collection, however, always involves measurement error that causes the observed network to differ from the true network. This, in turn, introduces error into the resulting fitted model. In our study, we introduced simulated measurement error into a social network of high school friendships in order to investigate the robustness of ERGMs when faced with noisy data. The resulting ERGM coefficients and descriptive statistics of the perturbed networks were compared to the original, unperturbed values.

I gratefully acknowledge the support of the Advanced Research Training for Undergraduates Program at Washington University in St. Louis and the NSF Award DMS-1055897.

Contents

1 Introduction

1.1 Network Data

Network data differ from conventional data in their structure. Conventional datasets consist of subjects or cases and information about those subjects or cases. The information is stored, often in a rectangular array, as scores for different variables or attributes [4]. For example, a conventional dataset could hold the gender, height, weight, and mile pace for the students in a high school. A network, on the other hand, consists of subjects or nodes and the connections or edges between them [4]. A network could be formed from the mutual friendships of the students in the high school.

The simplest and most common type of network is binary. This means each edge indicates either the presence or absence of some relationship between two nodes. A network can also, however, have edges that contain more information. For instance, each edge may indicate different nominal relationships. The possibilities for edge values would represent a variety of potential relations, such as friend, family, co-worker, or no relation. Edges can also be ordinal, indicating, perhaps, a positive, neutral, or negative relationship between nodes. Finally, edges can be weighted to indicate a difference in the strength of a relation between two nodes. These weights can be both positive and negative. Most networks, however, are converted into binary networks because they are easier to work with and analyze [4].

Directedness is another characteristic of edges. A directed network can be thought of as containing nodes that send edges to other nodes. In a study of friendship, suppose Jack identifies Jill as his friend, but Jill does not identify Jack as hers. In this directed network, there is an edge from Jack to Jill, but not one in reverse. An undirected network, on the other hand, just indicates the presence of some mutual relation. In the friendship example, an undirected edge indicates a mutual friendship between two people.

1.1.1 Adjacency Matrix

Network data are often organized in a square array called an adjacency matrix. The nodes in the network form both the rows and the columns, and the cell entries indicate information about the edge between those two nodes. For a binary network, the adjacency matrix indicates the presence or absence of an edge with a 1 or 0, respectively (see Figure 1) [4]. An undirected network

	1	2	$\cdots$	$\cdots$	j	$\cdots$	n
1	0	1					
2	1	0					
$\vdots$			$\ddots$				
i				0	$w_{i,j}$		
$\vdots$					$\ddots$		
$\vdots$						$\ddots$	
n							0

Figure 1: An adjacency matrix where $w_{i,j}$ indicates the presence (1) or absence (0) of an edge between node i and node j.

has a symmetric adjacency matrix where the entries above the diagonal are identical to the entries below. A directed network does not necessarily have a symmetric adjacency matrix. For an edge in a directed network, the sender forms the row and the receiver forms the column. The main diagonal of an adjacency matrix, indicating a reflexive edge, is often set to zero and ignored. Adjacency matrices do not have to be binary; the entries can indicate ordinal or weighted edges as well.

An important aspect of adjacency matrices is the fact that the entries in the matrix, the edges of the network, are considered random variables. This doesn't mean they form in random ways; rather, it means it is simply not possible to know everything about how these relationships form [6]. Because of this randomness, probability models are used to understand the structure of networks.

1.1.2 Social Network Examples

Social networks exist everywhere in our society and simple examples are easy to find. The most common example of a social network, already used above, is a friendship network. In this example, each node represents an individual, and each edge indicates some sort of relationship between those individuals. If the graph is directed, an edge would represent one person's feeling of friendship for another. If the graph is undirected, an edge would represent a mutual friendship between the two individuals.

Another example of a social network is an authorship network [9]. If I am interested in studying collaboration among statistics professors, I could set up a network where each professor who has authored a paper forms a node. Any two professors who are listed as co-authors would have an edge between them. This network would be undirected, but it could be modified to include more information about the collaborations. For example, weighted edges could be used to indicate the number of papers that two professors have worked on together.

Social networks do not, however, have to include people. The online encyclopedia Wikipedia forms a huge network where each page is a node. A directed edge exists between two nodes if there is a link on one page leading to the other. On the "social network" Wikipedia page there is a link to the "sociology" Wikipedia page, and vice versa. This means, in the directed Wikipedia network, there is an edge from "social network" to "sociology" and back [7, 8].

1.2 Descriptive Statistics

In order to describe a social network, there are a number of descriptive statistics that can be calculated. These statistics give a general idea of how nodes are related to each other and how they are embedded in the larger network as a whole. There are many such descriptive statistics, but I will give just a few simple examples that will be used later in this paper.

First, for an undirected network, nodal degree is calculated as the row sum (or column sum) of the adjacency matrix and indicates the number of connections each node has. In Figure 2, each outer node of the star has degree 1 and the middle node A has degree 6. The degree for node i is calculated as follows:

$$k_i = \sum_j w_{i,j} \tag{1}$$

For a directed network, each node has an out-degree and an in-degree based on the number of edges it sends out and receives, respectively. The out-degree is the row sum of the adjacency matrix and the in-degree is the column sum. In this research, I focused on undirected networks where the out-degree equals the in-degree and is simply called the degree.

Another simple descriptive statistic is degree centralization, which is based on nodal degree. This measure compares the node with the highest degree, denoted k^*, to all other nodes in the network. This is a measure of the tendency of a single point to dominate the network, or to be more cen-

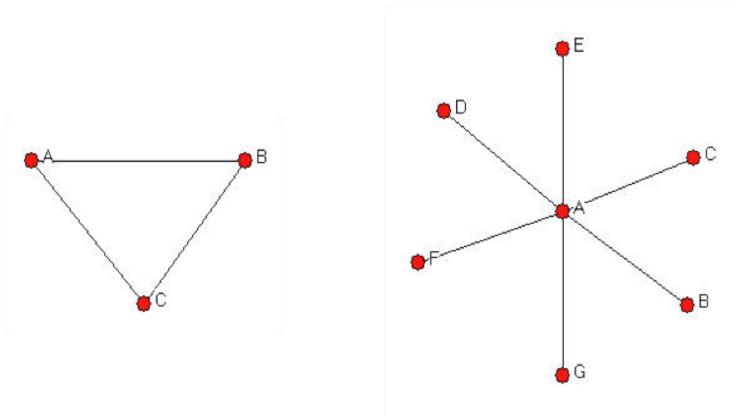


Figure 2: Two simple network structures: on the left is a triangle and on the right, a star [4].

tralized in the network than all other points [2]. The degree centralization for a network is calculated as follows:

$$C_D \propto \sum_i (k^* - k_i) \tag{2}$$

Finally, the triangle is a very basic network structure. The left graph in Figure 2 is a triangle, and there are none in the right graph. While this is an extremely simple concept, triangles appear quite frequently in social networks. It is common to become friends with the friends of your friends. This relationship would create a closed triangle in a friendship network.

1.3 Measurement Error

Measurement error refers to the discrepancies between the observed network and the true network. There are many such types of error, but in this research, I focused specifically on two: missing edges (false negatives) and spurious edges (false positives). There are two common sources that cause these types of error in the data. The first is error that comes from the researcher due to mistakes in the collection or coding of the data. For example, limiting the allowed number of relations that respondents can list may cause the network to be artificially sparse [9]. The second major source is error that the research subjects make, often due to differences in perception. If a group of people are asked to write down the names of their friends, different people will have different ideas about the type of relationship necessary for someone to be called a friend [9].

1.4 Exponential Random Graph Model (ERGM)

An exponential random graph model (ERGM) is one kind of probability distribution for the random adjacency matrix W [1]. To create a model, the number of nodes n is considered to be fixed. The set of possible networks, then, includes all of the different arrangements of edges given that fixed number of nodes. This means that the observed configuration of edges, w, is one out of a very large set of possible configurations [6]. The probability is modeled as:

$$P(W = w \mid \theta) = \frac{exp\{\theta^T g(w)\}}{z(\theta)} \tag{3}$$

or

$$P(W = w \mid \theta) \propto exp\{\theta^T g(w)\} \tag{4}$$

where: w is the observed adjacency matrix
θ is a vector of model coefficients
$g(w)$ is a vector of statistics
$z(\theta) = \sum_W exp\{\theta^T g(w)\}$ is a normalizing constant

The normalizing constant $z(\theta)$ in equation 3 is a sum over all possible network configurations, making it very difficult to calculate in anything other than a very small network [1]. In practice, therefore, estimation methods are used to fit ERGMs so that the normalizing constant does not have to be calulated directly.

An ERGM explains how likely it is that the observed set of edges exists as a network, based on some hypotheses about the structure of the network. This probability depends on both the vector of statistics and the vector of model coefficients. The vector of statistics contains information about the hypotheses about how nodes are related to each other. When choosing which descriptive statistics to include in the model, we make a prediction about what structures or configurations of edges were important in the formation of the network. For example, we might include a triangle statistic in $g(w)$

in a the model of high school friendships because we think it is likely that closed triangles of friendship have formed.

The vector of coefficients corresponds to the vector of statistics. There is a model coefficient for each statistic which relates to the probability that that type of edge will exist. Consider again the friendship network and the hypothesis that triangles have formed. A statistic for triangles is included in $g(w)$ and the corresponding coefficient in θ relates the probability of existence for an edge that forms a closed triangle. Positive coefficients indicate what is more likely to happen, and negative coefficients indicate what is less likely. Given an adjacency matrix for some dataset, we want to estimate the model coefficient for each statistic.

Exponential random graph models are a method of quantification of the structure of a network. This allow researchers to make easier comparisons between different networks. Additionally, ERGMs can potentially be generalized. A model estimated from a network of friendships in a high school can give insight into the relationships that form in other high schools of similar size and with similar demographics.

1.4.1 Software

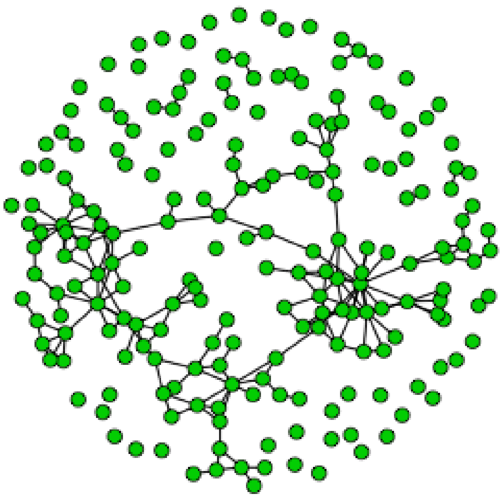
In this research, I used the ergm() function found in the **statnet** packages of the R software to fit exponential random graph models. This function uses Monte Carlo maximum likelihood estimation to estimate the vector of coefficients [3].

2 Data

In this work, I used the dataset Faux Mesa High found in the **statnet** packages of the R software [3, 5]. This dataset is a simulated network representing a high school population in the southwestern United States. The network has 205 nodes, which represent individual students, and 203 undirected edges, which represent mutual, reciprocated friendships. A graph of the Faux Mesa High dataset can be seen in Figure 3.

In the simulation of Faux Mesa High, each student was randomly assigned a race, gender, and grade in school corresponding to the values of an observed high school population. Race has six possible values: white and non-Hispanic, black and non-Hispanic, Asian and non-Hispanic, Hispanic,

Figure 3: Faux Mesa High.



Native American, and other and non-Hispanic. Grade takes on values 7 to 12, and the gender values indicate male or female.

3 Exploring ERGM Robustness

The goal of this research was to better understand the behavior of exponential random graph models when the data involves measurement error. We did this by fitting an ERGM to the Faux Mesa High dataset, using the resulting model coefficients to simulate a similar, but new network, and introducing measurement error by randomly adding and removing edges. I will go into more detail about these steps in the next few sections.

3.1 The Model

To start looking at the robustness of ERGMs, I first had to fit a model to the Faux Mesa High dataset. The model that I used consists of five terms (see *model* in Figure 4). The first term of the model is *edges*, which takes into account the total number of edges in the network. This term reflects the propensity of any type of edge existing.

The next three terms in the model, called *nodematch* for race, sex, and grade, look for assortative mixing in the network. Assortative mixing is the

```
library("statnet")
data(faux.mesa.high)
dat <- faux.mesa.high

model <- ergm(dat ~ edges + nodematch("Grade") + nodematch("Race") + nodematch("Sex") + gwesp(0.4, fixed = T),
              control = control.ergm(MCMC.samplesize = 1e+5, seed = 123))

sim.net <- simulate(model, seed = 1534)
```

Figure 4: Fitting the model to the Faux Mesa High dataset.

idea that in social networks, nodes tend to form relations with others who have similar attributes. In this case, we are hypothesizing that people in high school are more likely to be friends with people of the same grade, the same race, and the same gender as themselves. This seems to be a good hypothesis based on the experience of being in high school.

The final term in the model is *gwesp*, which stands for geometrically-weighted edgewise shared partners. This is basically a method of looking at closed triangles in the network. It is based on the number of times that two connected nodes (a dyad) have partners in common. The statistic is a vector whose entries indicate the number of pairs with one shared partner, two shared partners, and so on, up to five shared partners. Including this term in the model hypothesizes that two friends are likely to have other friends in common.

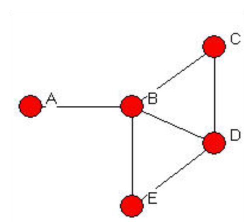


Figure 5: In this simple network, the dyad B – D has two edgewise shared partners: nodes C and E [4].

After choosing appropriate statistics, I fit that model to the Faux Mesa High dataset, which resulted in coefficient values for each of the five statistics, number of edges, assortative mixing for grade, race, and gender, and number of edgewise shared partners. Using those coefficient values, I then simulated a new network (see *sim.net* in Figure 4). This allowed me to know the true coefficient values for my new network *sim.net*, making it possible to more

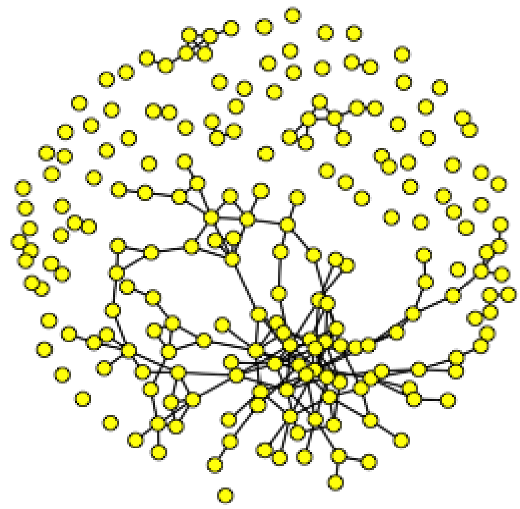


Figure 6: *sim.net*: the network simulated from Faux Mesa High's ERGM coefficient estimates.

easily and accurately evaluate deviations from these true values in my later analysis. The baseline network, what I used to test robustness, therefore, is different from the Faux Mesa High dataset. The simulated network *sim.net* still has 205 nodes, because we assume the number of nodes is constant, but has only 192 total edges.

3.2 Simulating Measurement Error

As mentioned above, in this research I focused on measurement error manifested through missing and spurious edges in the network. To simulate this type of error, I randomly added and removed edges from the network with various probabilities. To do so, I ran through every possible pair of nodes in the network. If there was an edge between that pair, it was removed with probability p; if there was not an edge between that pair, it was added with probability q. The adding probabilities (q) were 0.001, 0.005, 0.01, and 0.05; the removing probabilities (p) were 0.01, 0.02, . . . , 0.20.

The choice of these probabilities was affected by the sparseness of the original network *sim.net*. There are 205 nodes in that network, which means there are 20,910 possible edges for an undirected network. The network *sim.net*, however, only has 192 existing edges. Therefore, even the smallest adding probability of 0.001 would increase the number of edges by close to one tenth. Because of this feature of *sim.net*, I chose small adding probabilities and a wide range of removal probabilities.

After perturbing the network to simulate measurement error, I then refit the original ERGM to get new coefficient estimates. I also calculated some descriptive statistics of the perturbed network, including degree centralization and number of edgewise shared partners. I compared these estimated coefficients and statistics to the original values for *sim.net* to try to understand the effect of the measurement error on the model and on the structure of the network.

This process of perturbing, modeling, and calculating statistics was done 100 times at each combination of adding and removing probabilities. With adding probability $q = 0.05$, however, these computations for 100 networks took over 5 hours to complete. Because of this extended time frame, I decided to perturb only 50 networks for each removal probability, which cut the computation time approximately in half.

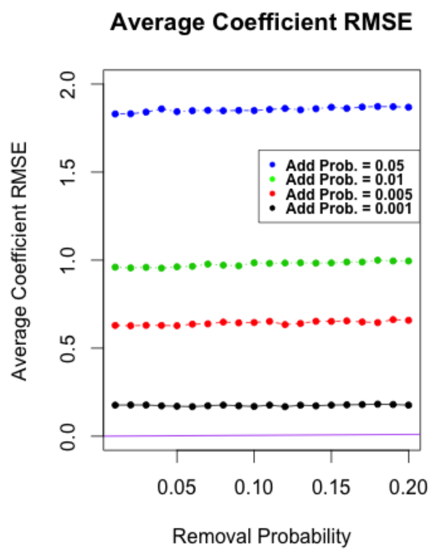
3.3 Results

After simulating a range of measurement error, I needed to quantify the differences between the perturbed networks and the original *sim.net*. To do so, I first looked at the changes in coefficient estimates by calculating the root mean square error (RMSE) of the perturbed ERGM coefficients compared to the originally calculated or true coefficients.

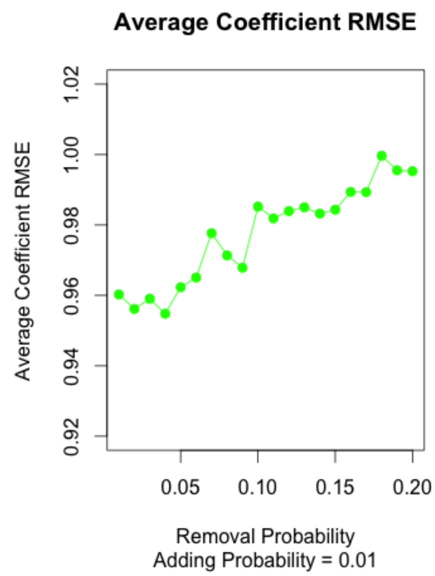
edges	nodematch.Grade	nodematch.Race	nodematch.Sex	gwesp.fixed.0.4
-6.6670613	1.9720185	0.2808436	0.5395049	1.3286320

Figure 7: The true coefficient values for *sim.net* without any simulated measurement error.

Graph (a) in Figure 8 shows the average of the root mean square error that was calculated for each network at each combination of perturbation probabilities. Therefore, individual data points represent the average of the RMSE for all 100 perturbed networks at that probability combination (or average of 50 networks for $q = 0.05$). It is clear from this graph that the error in the coefficient estimates increases as the adding probability increases. With more and more spurious edges, the estimated coefficient values get further from the true values. This makes sense because large random perturbations



(a)



(b)

Figure 8: The average ERGM coefficient root mean square error for (a) all the perturbed networks and (b) $q = 0.01$.

significantly change the structure of the network. These changes in structure would then cause the ERGM coefficients to also change in major ways.

Graph (b) in Figure 8 is the plot of adding probability $q = 0.01$, the same as the green line in graph (a), displayed with a smaller vertical scale. This allows us to more clearly see the trend as removal probability increases. Graph (b) shows that the average RMSE also has an upward trajectory as the removal probability increases; more and more missing edges causes the estimated coefficients to get further from the true values. Similarly to adding extra edges, removing edges will also change the network structure and the ERGM coefficients. It is not the total number of edges that matters. If I randomly add 20 edges and then randomly remove 20 edges, the network has the same number of edges that it started with. The structure, however, has been in changed in as many as 40 different places. This will cause the coefficient estimates to be very different from the original values. Similar graphs of the average coefficient RMSE for $q = 0.001, 0.005$, and 0.05 can be found in Appendix A.1.

By comparing these two graphs, we can further see that the average coefficient RMSE increases more quickly with increasing adding probabilities than with increasing removal probabilities. The average RMSE increases by almost 1 when q increases from 0.01 to 0.05. On the other hand, the average RMSE increases by less than 0.05 when p increases from 0.01 to 0.05. This is presumably due to the sparseness of the original network *sim.net*, as mentioned above.

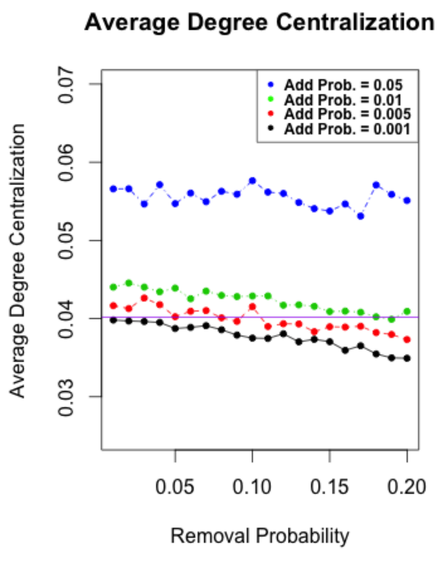
While some error in the estimated ERGM coefficients is expected, it is the magnitude of this error that is interesting. As outlined earlier, the network *sim.net* was simulated from a set of coefficients that are treated as the true coefficient values. When I fit the ERGM to *sim.net*, however, the resulting coefficient estimates were not the same as the true values. There is some error in the estimation process. In the unperturbed *sim.net*, however, the RMSE is approximately 0.0522 (denoted by the purple line in graph (a) of Figure 8). This RMSE is much lower than that of any of the perturbed networks, indicating that introducing measurement error has a noticeable effect on the ERGM coefficient estimates.

In addition to calculating error in the coefficient values, I also looked at a few descriptive statistics to investigate how they were affected by measurement error. These statistics included number of edges, degree centralization, and number of edgewise share partners. The statistic for number of edges in the network was the most straightforward. With increasing adding probability, the number of edges increased, and with increasing removal probability, the number of edges decreased. This is a direct result from the method of introducing measurement error.

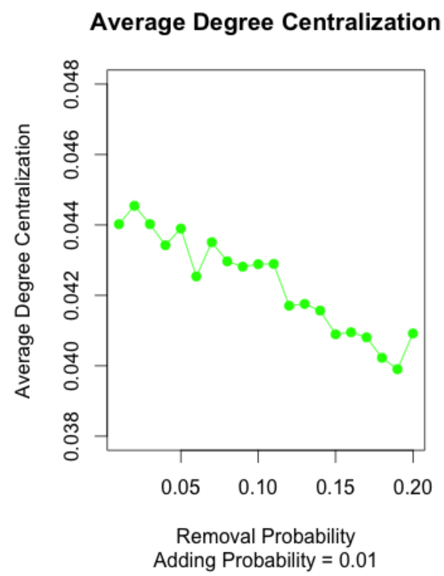
A more interesting statistic to look at is the degree centralization of the network, a measure of the dominance of a single point compared to all other points. The network *sim.net* has degree centralization 0.0402 (denoted by the purple line in graph (a) of Figure 9). Plot (a) in Figure 9 shows how that value changes with the introduction of simulated measurement error. Average degree centralization appears to increase as more and more edges are randomly added to the network.

Plot (b) of Figure 9 shows the graph of the average degree centralization for adding probability 0.01 and across the range of removal probabilities. In this plot, degree centralization appears to decrease steadily as the removal probability increases. This trend is also seen in the plots for $q = 0.001$ and 0.005, found in Appendix A.2, though it is less clear for $q = 0.05$.

The explanation for these trends is not completely clear. The behavior of



(a)



(b)

Figure 9: The average degree centralization for (a) all the perturbed networks and (b) $q = 0.01$.

degree centralization is complicated when there are both random additions and removals of edges present in the network. More investigation into the specifics of degree centralization may make this phenomenon clearer.

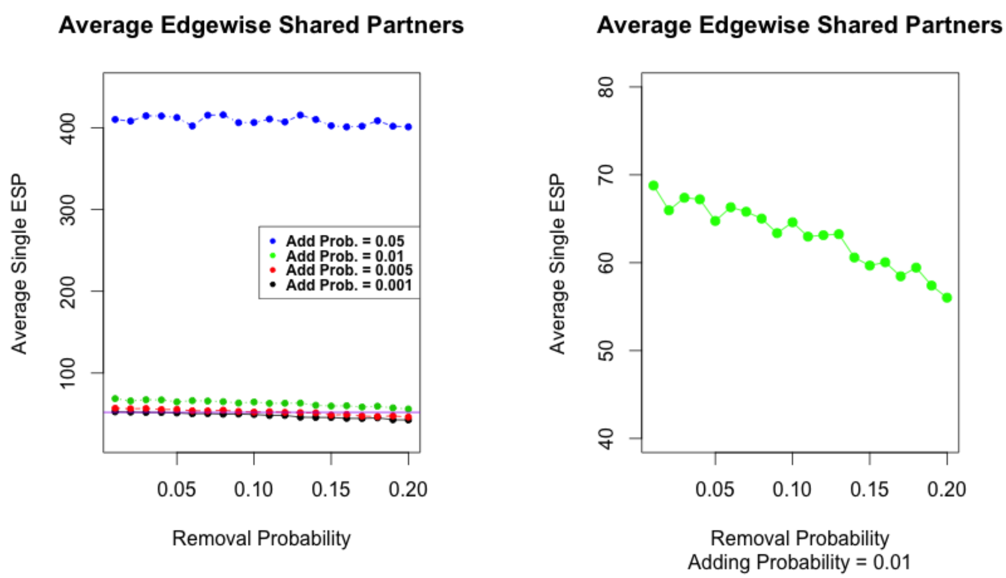
Finally, I looked at the number of dyads with one edgewise shared partner, which is the number of dyads that form a closed triangle with only one other node in the network. The original network *sim.net* has 52 dyads with only one edgewise shared partner (denoted by the purple line in graph (a) of Figure 10). Plot (a) of Figure 10 shows the difference in the number of single edgewise shared partners across adding probabilities; the number increases as more edges are added. The increase is small for adding probabilities $q = 0.001, 0.005$, and 0.01, but very large for $q = 0.05$ where the number of edgewise shared partners increases by more than 4 times.

It appears that the inital small increases are due to the fact that the simulated measurement error adds edges randomly and without a propensity to form closed triangles. With adding probability $q = 0.05$, however, around 1000 edges are added to the network. With this many new edges, there will be many more edgewise shared partners simply due to the volume.

In graph (b) of Figure 10, the plot of the average number of single edgewise shared partners for adding probability $q = 0.01$ shows a clearly decreasing trend. This trend is mirrored in the plots for $q = .0001$ and 0.005 shown in Appendix A.3, though it is less clear for $q = 0.05$.

The decrease in number of edgewise shared partners is rather steep in this plot because the triangle is a slightly more complex structure. For a network to lose an edgewise shared partner, only one of the three edges forming the triangle has to be removed. Additionally, removing that single edge could eliminate as many as three triangles from the network.

Although there are many descriptive statistics that I could have used as indicators of the network's structure, I chose to focus mainly on the number of edges, the degree centralization, and the number of single edgewise shared partners, as analyzed above. I also collected data, however, on the density of the network, the number of components, and the propensity of assortative mixing. These statistics, though, have behavior very similar to the focal



(a) (b)

Figure 10: The average number of dyads with a single edgewise shared partner for (a) all the perturbed networks and (b) $q = 0.01$.

statistics that I used. In fact, density, the ratio of existing edges to possible edges, is just a function of the number of edges in the network.

4 Conclusions

The clear conclusion from these results is that exponential random graph models are not very robust and that they do not do well when any amount of measurement error is present in the data. Even with small perturbations, the model coefficients were significantly different from the unperturbed network *sim.net*. The descriptive statistics also deviated largely from the unperturbed network, though less so at small measurement error values.

Therefore, because ERGMs are so sensitive to error in the data, they should ideally be used only in situations where the data is very clean and precise. Unfortunately, that is something that is very difficult to ensure, or even to know. This lack of robustness, however, needs to be kept in mind when fitting ERGMs to potentially noisy data.

5 Future Research

There are a few areas where this research could be expanded in the future. To begin with, it would be beneficial to do similar work on a network with different characteristics. The Faux Mesa High dataset that I used is a very sparse network – only about 1% of possible edges actually exist. Although I am unable to say conclusively, I believe that some of the effects of the simulated measurement error in this research may be more pronounced due to the sparsity of the network. For example, in Appendix A.2 and A.3 it is clear that the plots of adding probability $q = 0.05$ does not neatly follow the same downward trend that the other plots show. I hypothesize that this is due to the sparsity of the network. Adding probability $q = 0.05$ adds such a huge number of edges to the original network *sim.net* compared to the number of edges removed by the range of removal probabilities. This may cause the effect of removing edges to be unnoticeable in these statistics.

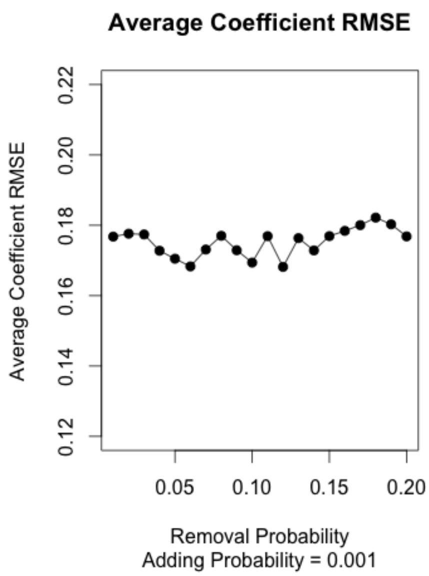
Looking at the effects of measurement error on a more dense network would allow me to compare results. From the analyses of the two different networks, I could potentially see differences due to the number of edges and make more general conclusions.

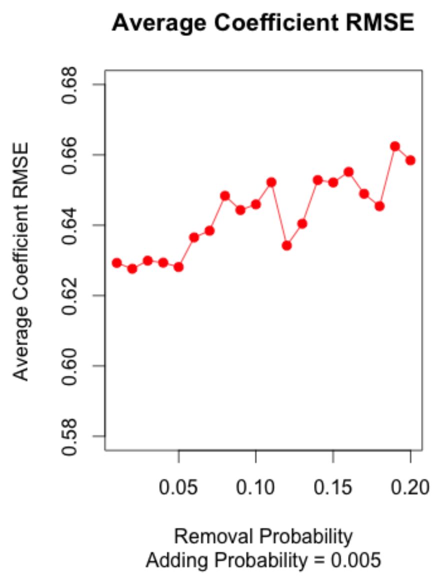
Additionally, we could expand this research to include other types of measurement error. In this work, I looked only at error that was manifested through missing or spurious edges. There are, however, other types of measurement error that are commonly found in network data. Similar to what we looked at for edges, observed data can also have missing or spurious nodes. For example, when creating an authorship network, an author's name may be spelled differently in two different papers, one correct and one incorrect. This will cause a single node to show up in the network as two separate nodes. On the other hand, in a friendship survey, two people might have the same name and be interpreted as a single node [9]. These types of measurement error may impact the network and fitted exponential random graph models differently.

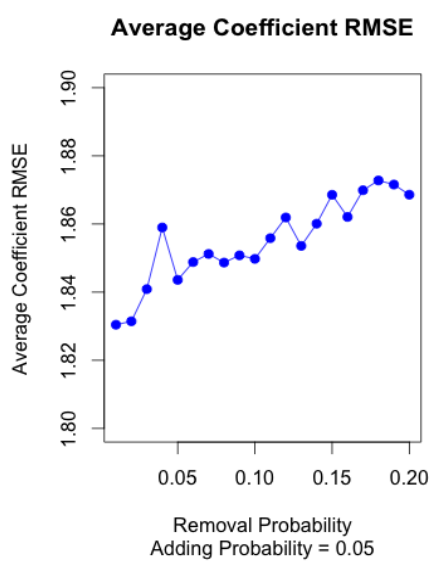
Finally, a major avenue for future research would be to look into improving model accuracy. This work has shown that measurement error has a large detrimental effect on the accuracy of exponential random graph models, as seen through both ERGM coefficients and descriptive statistics. A logical next step, therefore, is to work on methods for removing the error from the data.

A Appendix

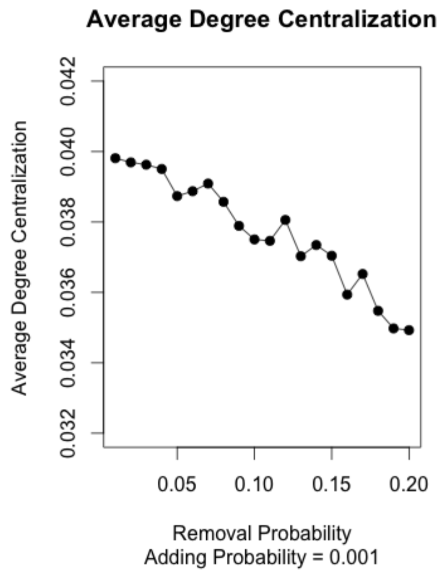
A.1 Average Coefficient Error

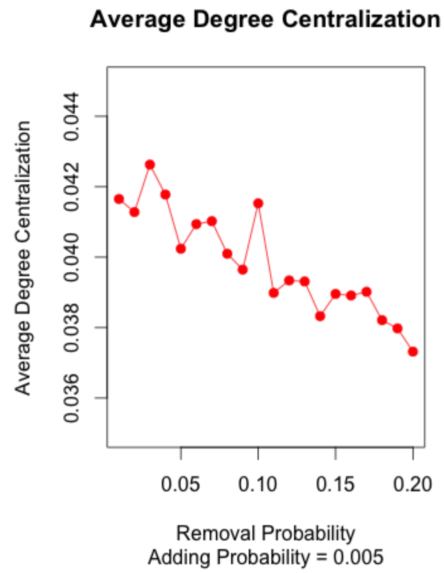


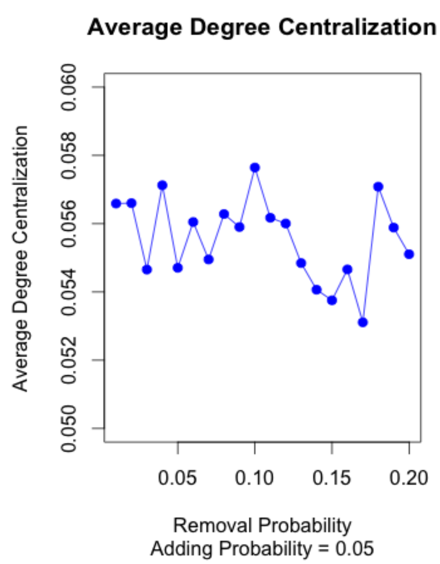




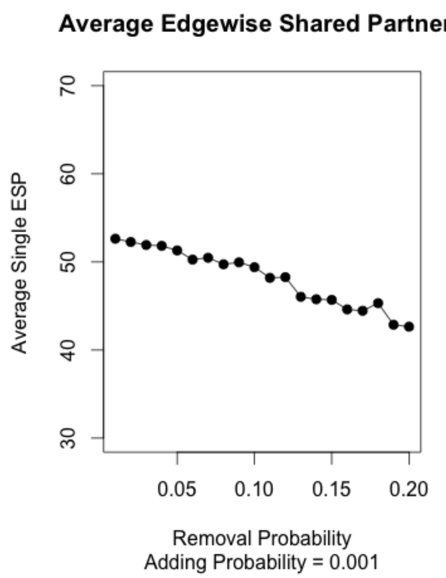
A.2 Average Degree Centralization

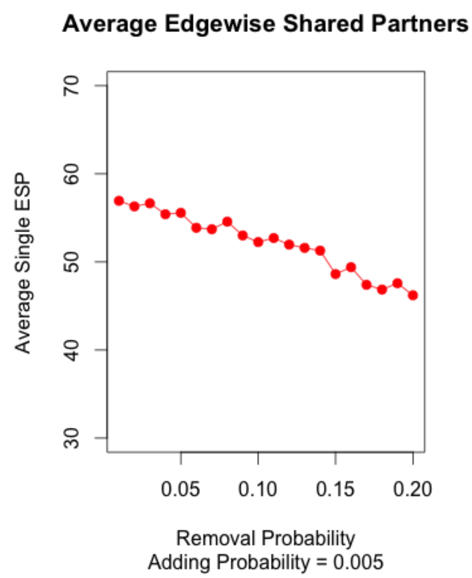


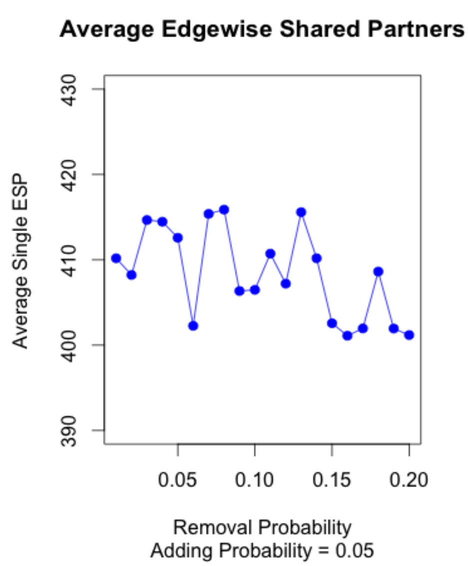




A.3 Average Single Edgewise Shared Partners







References

[1] Caimo, A. & Friel, N. (2010). Bayesian inference for exponential random graph models. *Social Networks*. http://arxiv.org/abs/ 1007.5192

[2] Freeman, L. C. (1979). Centrality in Social Networks Conceptual Clarification. *Social Networks*, 1. Retrieved from https://cs.stanford.edu/people/jure/pubs/error-socnets12.pdf

[3] Handcock, M. S., Hunter, D. R., Butts, C. T., Goodreau, S. M., & Morris, M. (2003). ***statnet****: Software tools for the Statistical Modeling of Network Data*. Retrieved from http://statnetproject.org

[4] Hanneman, R. A. & Riddle, M. (2005). Introduction to social network methods. Riverside, CA: University of California, Riverside. Retrieved from http://faculty.ucr.edu/ hanneman/

[5] Resnick M.D., Bearman, P.S., Blum R.W., et al. (1997). Protecting adolescents from harm. Findings from the National Longitudinal Study on Adolescent Health. *Journal of the American Medical Association*, 278: 823-32.

[6] Robins, G., Pattinson, P., Kalish, Y., & Lusher, D. (2007). An introduction to exponential random graph (p^*) models for social networks. *Social Networks*. doi:10.1016/j.socnet.2006.08.002

[7] Social networks. (n.d.). *Wikipedia*. Retrieved March 16, 2016, from https://en.wikipedia.org/wiki/Social_network

[8] Sociology. (n.d.). *Wikipedia*. Retrieved March 16, 2016, from https://en.wikipedia.org/wiki/Sociology

[9] Wang, D.J., et al. (2012). Measurement error in network data: A reclassification. *Social Networks*. doi:10.1016/j.socnet.2012.01.003