Visual Inference for Significance and Goodness-of-Fit Testing for a Social Network Model

Samantha Tyner*
Department of Statistics and Statistical Laboratory, Iowa State University
and
Heike Hofmann
Department of Statistics and Statistical Laboratory, Iowa State University

October 27, 2017

Abstract

Two of the most imporant pieces of statistical modeling are significance testing of model parameters and goodness-of-fit measures and tests. The more complicated the model, the harder it is to determine fit or whether to include or exclude a parameter. Some particularly complicated sets of models are those designed to model network change. By using the visual inference methodology of Buja et al. (2009), we can look at the entire dataset simulated from a network model as opposed to a single metric on the network such as outdegree, or a p-value for a single parameter in the model.

Keywords: social network analysis, visual inference, dynamic networks, network visualization, network mapping, goodness-of-fit, hypothesis testing

*The authors gratefully acknowledge funding from the National Science Foundation Grant # DMS 1007697. All data collection has been conducted with approval from the Institutional Review Board IRB 10-347

Contents

1 Introduction

Two of the most important pieces of statistical modeling are significance testing of model parameters and goodness-of-fit tests. In the former, the data are usually assumed to come from a simple model under the null hypothesis, and additional parameters are tested whether they significantly contribute to explaining variability in the data. In the latter, the model of interest is examined to determine how well it fits the data. Both of these aspects of statistical modeling increase greatly in difficulty as the complexity of the model of interest increases.

Some particularly complicated sets of models are those designed to model network change. A *network* is any set of things, such as people, computers, or neurons, that are connected in some way, through social relations, internet connection, or electrical impulses in the brain. We refer to the "things" in the network as *nodes*, or *actors* in a social network, and the connections as *edges*, or *ties* in a social network. Dependencies inherent to the data make network objects particularly difficult to model. Even more challenging is the situation when we go beyond single instances of a network and consider the dynamics of network change between observed instances. This type of modelling for dynamic networks is often performed on social network data, such as for friendship networks among students or the spread of HIV in sexually active communities.These models lack the asymptotics required to perform many well-known goodness-of-fit tests, and the maximum likelihood estimation of parameters is so difficult that it can make significance testing significant as well (Goldenberg et al., 2010). is the previous sentence a quote? I'm not sure that I follow.

We propose new methods for significance and goodness-of-fit testing for a set of social network models, stochastic actor-oriented models for dynamic network data (Snijders, 1996). Specifically, we are using *visual inference* in place of traditional statistical methods for social network models, such as Wald tests for significance of parameters and in- and out-degree distribution metrics for determining goodness-of-fit. Visual inference, introduced by Buja et al. (2009), allows us to look at the entire dataset simulated from a network model as opposed to a (set of) usually one-dimensional metric(s) derived from the network such as outdegree, or a p-value for a single parameter in the model.

The paper is outlined as follows: Section 2 gives a basic overview of visual inference and the lineup protocol. Section 3 provides an introduction to the our models of interest, stochastic actor-oriented models. Sections 4 and 5 discuss problems with traditional significance tests and goodness-of-fit measures for these models and how visual inference methods can be used as solutions. Section 6 outlines our experimental protocol, and we close with a discussion in Section 7.

where do the results come in?

2 Visual Inference

Data visualizations are an important component of data analysis, providing a mechanism for discovering patterns in data. Pioneering research by Gelman (2004), Buja et al. (2009) and Majumder et al. (2013) provide methods to quantify the significance of discoveries made from visualizations. Buja et al. (2009) introduced two protocols, the Rorschach and the lineup protocol, which bridge the gulf between traditional statistical inference and exploratory data analysis. Here, we use the lineup protocol. Under this protocol, a plot of the observed data is placed randomly among a set of $m - 1$ null plots (where $m = 20$, usually), and human observers are then asked to examine the lineup and to identify the most different plot. If an observer identifies the data plot, this is quantifiable evidence against the null hypothesis. Similarly, an observer has a chance of 1 in m to pick the data plot from the lineup by simply guessing, ie in a situation where the data plot is virtually indistinguishable from the null plots.

The lineup protocol places a plot firmly in the framework of hypothesis tests: a plot of the data is considered to be the test statistic, which is compared against the sampling distribution under the null hypothesis represented by the null plots. Obviously, the null generating mechanism, i.e. the method of obtaining the data for null plots, is crucial for both the lineup and the Rorschach protocol, as the null hypothesis directly affects the choice of null generating method. Null generating methods are typically based on (a) simulation, if the null hypothesis allows us to directly specify a parametric model, (b) sampling, as for example in the case of large data sets, or (c) permutation of the original data (see e.g. Good, 2005), which allows for non-parametric testing that preserves marginal distributions while ensuring independence in higher dimensions. The model of interest here allows us to simulate directly from a parameteric model for dynamic social network data.

The lineup protocol was formally tested in a head-to-head comparison with the equivalent conventional test in Majumder et al. (2013). The experiment utilized human subjects from Amazon's Mechanical Turk (Amazon, 2010) and used simulation to control conditions. The results suggest that visual inference is comparable to conventional tests in a controlled conventional setting. This provides support for its appropriateness for testing in real exploratory situations where no conventional test exists.

3 Stochastic Actor-Oriented Models

Stochastic Actor-Oriented Models (SAOMs) are a family of models for dynamic network data (Snijders, 1996) that incorporate both network structure and node-level information to describe how a network observed on two or more occasions changes over time. The two titular properties of SAOMs, stochasticity and actor-orientation, are crucial to understanding networks as they exist naturally: social networks are ever-changing as relationships decay or grow in seemingly random ways, and most actors in them have characteristics that could affect how they change their ties to other nodes in the network. These unique properties allow for the fitting of some very complicated models to inherently complex data, so it can be exceedingly difficult to interpret parameters and their corresponding estimates. The sheer amount of possible parameters to include in the model combined with the difficulty of interpretation make parameter selection and goodness-of-fit testing burdensome as well.

Broadly, a SAOM takes network structure and node covariate information into account in two ways and models the network changes as a continuous time Markov chain (CTMC). First, the rate of change between states is dictated by a rate function that describes *how often* changes in the network occur, and secondly, the objective function describes *what* those state changes are. As in many other network models, the variables of interest, are the binary edges of the network. Let x_{ij} denote the edge between nodes i and j, where $i, j \in \{1, 2, \ldots, n = \text{the number of nodes}\}$. x_{ij} is modelled as a binary variable, i.e.

$$x_{ij} = \begin{cases} 1 & \text{if an edge from } i \text{ to } j \text{ exists} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Edges are treated as *directed*, i.e. in general $x_{ij} \neq x_{ji}$, and self-referencing edges or loops are not allowed, i.e. $x_{ii} = 0$ for all i. Assume, the network is observed M times at time points $t_1 < t_2 < ... < t_M$, then the entire network at time point t_m is denoted as $x(t_m)$. In sections 3.1 and 3.2 we discuss the rate and objective functions of a SOAM in more depth. Additional details on SOAMs can be found in Snijders (1996, 2001); Snijders et al. (2010b, 2007, 2010a); Snijders (2017),

3.1 Rate Function

All changes in SAOMs are treated as changes made by the nodes, or *actors*, in the network, i.e. each actor, i, gets a chance to make a change according to the rate function, typically denoted λ_i, which dictates when relationships between nodes in the network can change. In general, the rate function can take the network structure e.g. outdegree of node i, and the node covariates into account, but we use the simple rate function, which is constant over all nodes in a given time period. We denote the rate from t_m to t_{m+1} as α_m for $m = 1, \ldots, M-1$. Using this notation, the waiting time to the next chance for actor i to make a change is exponentially distributed with expected value α_m^{-1}. Since the rate is the same for all actors, the waiting time for *any* actor to get the chance to change is exponentially distributed with expected value $(n\alpha_m)^{-1}$.

3.2 Objective Function

After actor i has been given the opportunity to change, it probabilistically chooses one of its current ties, x_{ij}, to change. The probability that actor i changes its current tie to actor j is determined by the *objective function* of the model and a random component, U. XXX can you paraphrase the purpose of U in half a sentence? Actor i is aiming to maximize the objective function f_i given the current state of the network, x and the node-level covariates, $\mathbf{Z}$, given as:

$$f_i(x, \boldsymbol{\beta}, \mathbf{Z}) = \sum_{k=1}^{K} \beta_k s_{ik}(x, \mathbf{Z}), \tag{2}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)$ are additional model parameters, each associated with some network statistics, $s_{ik}(x, \mathbf{Z})$, $s_{ik}(x, \mathbf{Z})$, calculated with respect to actor i. Network statistics range from the simple outdegree, $s_i(x) = \sum_{i \neq j} x_{ij}$, to the more complicated *transitive triplets jumping to different covariate*, $s_i(x, \mathbf{Z}) = \sum_{i \neq j \neq h} x_{ij} x_{ih} x_{hj} \cdot \mathbb{I}(z_i = z_h \neq z_j)$. Version 1.2-3 of `RSiena` (Ripley et al., 2013), the software used to fit the models here, provides over 80 possible effects that can be included in the objective function. We discuss these statistics in more detail in Section 3.4.

Objective function $f_i(x, \boldsymbol{\beta}, \mathbf{Z})$ and random component U are combined to form the *transition probability*, p_{ij}, of the network changing from its current state x to the state with changed tie x_{ij}, denoted as $x(i \rightsquigarrow j)$:

$$p_{ij} = \frac{\exp\{f_i(x(i \rightsquigarrow j), \boldsymbol{\beta}, \mathbf{Z})\}}{\sum_h \exp\{f_i(x(i \rightsquigarrow h), \boldsymbol{\beta}, \mathbf{Z})\}} \tag{3}$$

This probability dictates which edge change is made by the acting node. The acting node can also choose to *not* change at all, in which case $j \equiv i$.

XXX how is the probability computed in the case that no change is made? Why is the notation $j \equiv i$ used for no change?

According to Snijders et al in the `RSiena` manual, at least two parameters must be included in the objective function: the density and the reciprocity. We denote the density, or out-degree, parameter by β_1 and the associated statistic as $s_{i1}(x) = \sum_j x_{ij}$. Similarly, we denote the reciprocity parameter by β_2 and the associated statistic as $s_{i2}(x) = \sum_j x_{ij} x_{ji}$. We refer to the model with only these two parameters in the objective function as M1.

3.3 Example Data

The data we use are collaboration networks in the United States Senate during the 111^{th} through 114^{th} Congresses, overlapping with Barack Obama's presidency. These senates began on January 6, 2009 and ended on January 3, 2017[1]. There are three legislative ways that senators can show support for legislation: they can author a bill, cosponsor a bill, and vote for a bill. We use cosponsorship as a metric because it results in a network that is unimodal (all nodes are senators) and directed. In this network, ties are directed from senator i to senator j when senator i signs on as a cosponsor to the bill that senator j

[1] Details of how this data can be downloaded are provided by Franois Briatte at `https://github.com/briatte/congress`

authored. There are many hundreds of ties between senators when they are connected in this way, so we simplify the network by computing a single value for each senator-senator collaboration called the *weighted propensity to cosponsor* (WPC). This value is defined in Gross et al. (2008) as

$$WPC_{ij} = \frac{\sum_{k=1}^{n_j} \frac{Y_{ij(k)}}{c_{j(k)}}}{\sum_{k=1}^{n_j} \frac{1}{c_{j(k)}}} \tag{4}$$

where n_j is the number of bills in a congressional session authored by senator j, $c_{j(k)}$ is the number of cosponsors on senator j's k^{th} bill, where $k \in \{1, \ldots, n_j\}$, and $Y_{ij(k)}$ is a binary variable that is 1 if senator i cosponsored senator j's k^{th} bill, and is 0 otherwise. This measure ranges in value from 0 to 1, where $WPC_{ij} = 1$ if senator i is a cosponsor on every one of senator j's bills and $WPC_{ij} = 0$ if senator i is never a cosponsor any of senator j's bills. Because SAOMs require binary edges, we construct the edges as follows:

$$x_{ij} = \begin{cases} 1 & WPC_{ij} > 0.25 \\ 0 & WPC_{ij} \leq 0.25 \end{cases} \tag{5}$$

For each of the four senate sessions, we have the WPC value between any two senators in the session, the party affiliation of each senator, the number of bills they authored in each session, and their gender. We explored each of these covariates in the model to determine if they affect the overall network structure and how ties are formed between senators. The node-link diagram representations of the data we use for modelling are shown in Figure 1. We have labelled some of the nodes in these networks whose names will be familiar to US readers, because they are leaders in their party or they have run for president. The size of the nodes represent how many bills the senator authored in a session, the color represents party affiliation, and the shape represent gender. In each of the four sessions, there is one very large connected component tying many of the prominent senators together, with many smaller groups of two to ten senators surrounding the larger component. In each senate, the structure changes slightly as new senators arrive or come to prominence.

For Senate 111, for instance, we see Hillary Clinton, serving out her second term in the senate until she became Secretary of State. She is isolated in Figure 1, but in actuality,

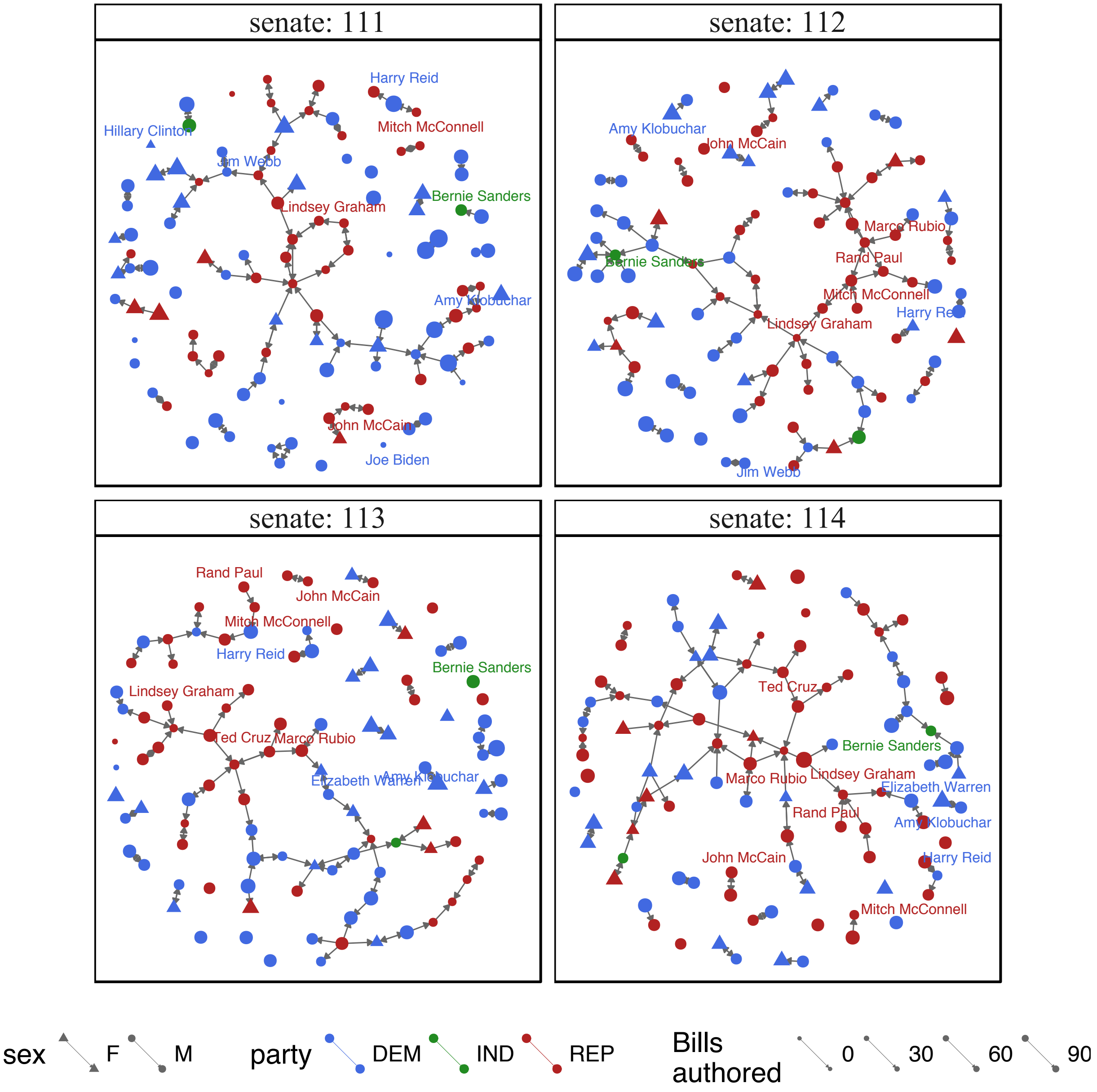


Figure 1: The four senate collaboration networks that we use as our example data to visually assess the SAOM effects. Color represents party, shape represents gender, and size represents number of bills authored in a session. The Frucherman-Reingold layout is shown.

she had many cosponsors on two pieces of legislation she authored in that short time, as is shown in Figure 2. We chose to remove Clinton and her edges from the network because they make the overall structure look so different from the other three senates, showing that the pattern is not typical of a senate in any other year. We suspect that because Hillary Clinton had just been appointed Secretary of State, the cosponsorships were largely symbolic, so the 111^{th} Senate without Hillary Clinton is more typical than the 111^{th} Senate with her.

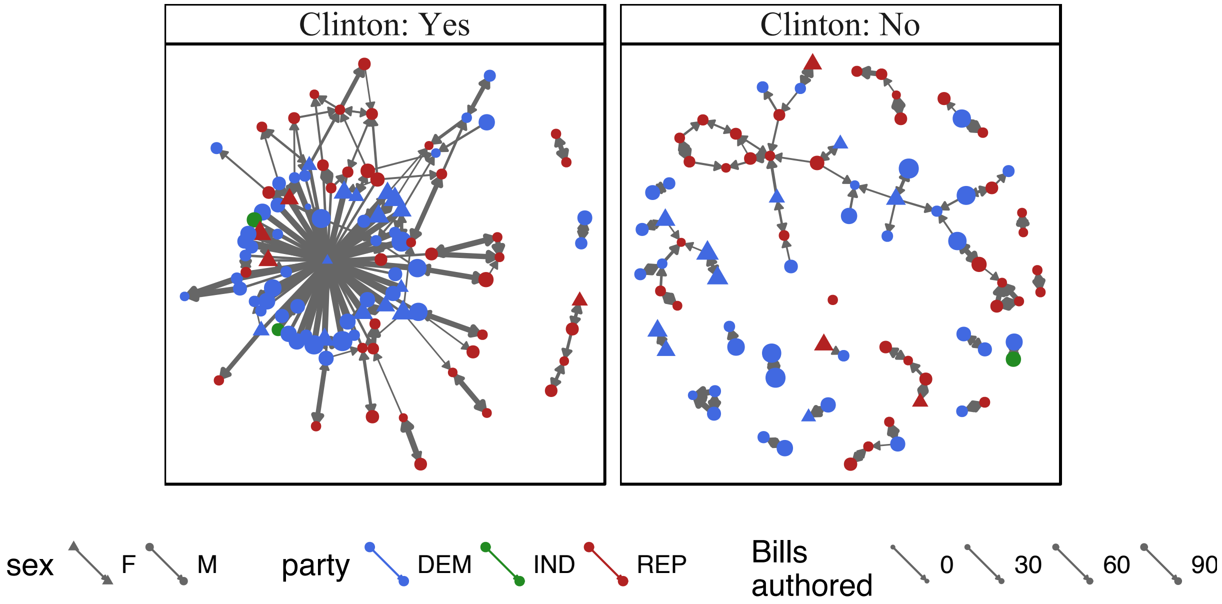


Figure 2: We removed Hillary Clinton's ties from the network because she had abnormally high collaboration with senators during the time she was in the 111th senate and before she left office to become Secretary of State.

In legislative cosponsorship networks, it is well known that party affiliation and reciprocity of relationships are major influences on structure (Nils Ringe and Cho, 2016). We focus on these two covariates when choosing which SAO models to fit to the data.

3.4 Models of Interest

In addition to considering already well-known effects in legislative networks for application of our significance and goodness-of-fit methods, we first fit many other possible models and selected a few significant effects. To determine the effects that we would move forward with, we followed this procedure:

1. Define the simple effects structure of the data: the rate parameters and the outdegree and reciprocity parameters.

2. Add each additional possible evaluation effect in `RSiena` one-at-a-time to the model structure, as determined by the effects documentation function (Ripley et al., 2013).

3. Fit each model to the data and check for convergence.
    (a) If the model converged, move to 4.
    (b) If the model did not converge, use the previous fitted values as starting values and repeat 5 times or until convergence, whichever comes first.
4. Test the added parameter for significance using a Wald-type test.
5. Report out the estimate of the additional parameter, its standard error, Wald p value, and convergence criterion.

After completing the procedure for all model effects, we selected effects whose estimates converged, had a Wald p-value of less than 0.10, and seemed to have a reasonable interpretation for our data according to well-known properties of legislative networks (Nils Ringe and Cho, 2016).

The parameters we use for the remainder of the paper are detailed in Table 1. The most significant effect was the jumping transitive triplet (JTT) parameter for the party covariate, which was estimated to be about -6 with a standard error of 0.11, resulting in a p-value of less than 0.0001. This estimate of the parameter associated with this statistic relies on the number of transitive closures formed between two senators from different parties. The negative estimate is an indication that forming transitive ties between two people from different parties is discouraged, which tracks with the divisive nature of American politics, where party affilitation is dominant. Another significant effect was the same JTT parameter for the sex covariate, with an estimate of about 3 with a standard error of 0.89. The covariate-related similarity score-weighted transitive triplets parameter estimate for the number of bills authored by a senator was also significant. This effect was estimated at about 10 with standard error of 3.9, and the high positive effect suggests senators tend to collaborate with other senators who author about the same number of bills they do. This tendency of senators to cosponsor bills written by senators who are similarly “prolific” corresponds to another well-known property of the U.S. Senate structure: the tendency of senators to be either “workhorses” or “showhorses”. Senators known as workhorses author many pieces of legislation in a session, and largley stay out of the public arena. The showhorse senators, on the other hand, author relatively few pieces of legislation, and tend

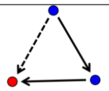
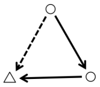
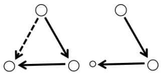
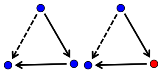
β_k	**Effect name**	**Interaction Variable**	**Formula**	**Picture**	**Initial estimate**	**Wald p-value**
β_3	jumping transitive triplet	party	$s_{i3}(x, \mathbf{p}) = \sum_{j \neq h} x_{ij} x_{ih} x_{hj} \cdot \mathbb{I}(p_i = p_h \neq p_j)$		-5.884	< 0.0001
β_4	jumping transitive triplet	sex	$s_{i4}(x, \mathbf{s}) = \sum_{j \neq h} x_{ij} x_{ih} x_{hj} \cdot \mathbb{I}(s_i = s_h \neq s_j)$		3.335	0.0002
β_5	similarity transitive triplet	bills	$s_{i5}(x, \mathbf{b}) = \sum_{j} x_{ij} x_{ih} x_{hj} \cdot (sim^b_{ij} - \overline{sim}^b)^*$		9.821	0.0128
β_6	same transtive triplet	party	$s_{i6}(x, \mathbf{p}) = \sum_{j} x_{ij} x_{ih} x_{hj} \cdot \mathbb{I}(p_i = p_j)$		1.306	0.0642

Table 1: The additional effects we used in the SAOMs fit to the senate data. * - $sim^b_{ij} = \frac{\max_{hk}|b_h - b_k| - |b_i - b_j|}{\max_{hk}|b_h - b_k|}$ is the similarity score between two senators based on the number of bills authored, and $\overline{sim}^b = \frac{1}{n(n-1)} \sum_{i \neq j} sim^b_{ij}$ is the average bill similarity score between any two senators.

to appear on television, radio, and other media a great deal. Finally, we found the same party transitive triplet effect was also significant, with a fitted value of 1.3 and standard error of 0.7, meaning that transitive relationships between senators tend to form when they are from the same party.

We examine a total of six models, each identified by its objective function:

1. Model M1: $f_i(x, \boldsymbol{\beta}) = \beta_1 s_{i1}(x) + \beta_2 s_{i2}(x)$
2. Model M2: $f_i(x, \boldsymbol{\beta}, \mathbf{p}) = \beta_1 s_{i1}(x) + \beta_2 s_{i2}(x) + \beta_3 s_{i3}(x, \mathbf{p})$
3. Model M3: $f_i(x, \boldsymbol{\beta}, \mathbf{s}) = \beta_1 s_{i1}(x) + \beta_2 s_{i2}(x) + \beta_4 s_{i4}(x, \mathbf{s})$
4. Model M4: $f_i(x, \boldsymbol{\beta}, \mathbf{b}) = \beta_1 s_{i1}(x) + \beta_2 s_{i2}(x) + \beta_5 s_{i5}(x, \mathbf{b})$

Model	α_1	α_2	α_3	β_1	β_2	β_3	β_4	β_5	β_6
M1	2.441	2.46	2.204	-4.903	4.893	–	–	–	–
M2	2.44	2.46	2.204	-4.902	4.893	-3.45	–	–	–
M3	2.438	2.461	2.211	-4.918	4.898	–	3.34	–	–
M4	2.442	2.459	2.206	-4.917	4.89	–	–	10.091	–
M5	2.443	2.461	2.205	-4.911	4.881	–	–	–	1.329
M6	2.441	2.459	2.21	-4.923	4.892	–	2.374	6.966	0.205

Table 2: The final estimates from repeated estimation of models M1 through M6.

5. Model M5: $f_i(x, \boldsymbol{\beta}, \mathbf{p}) = \beta_1 s_{i1}(x) + \beta_2 s_{i2}(x) + \beta_6 s_{i6}(x, \mathbf{p})$

6. Model M6: $f_i(x, \boldsymbol{\beta}, \mathbf{p}, \mathbf{b}, \mathbf{s}) = \beta_1 s_{i1}(x) + \beta_2 s_{i2}(x) + \beta_4 s_{i4}(x, \mathbf{s}) + \beta_5 s_{i5}(x, \mathbf{b}) + \beta_6 s_{i6}(x, \mathbf{p})$

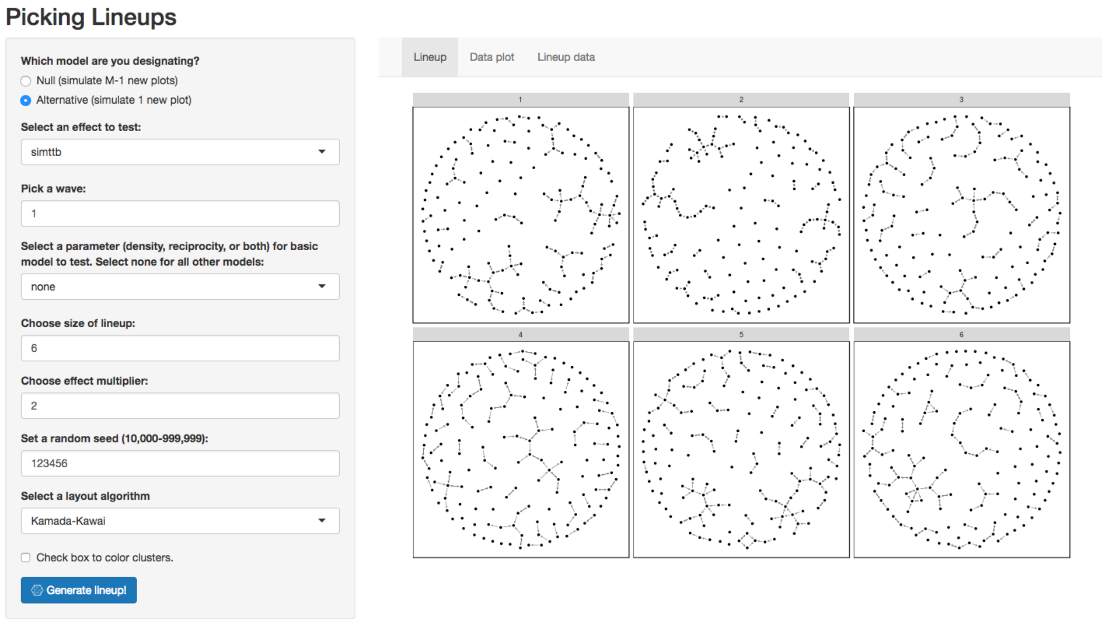
We fit models M1 through M6 in `RSiena` using Markov Chain Monte Carlo (MCMC) methods to approximate the method of moments estimates of the parameters. Because the estimation is done through MCMC simulation, we fit each model to the data 1,000 times to get a better estimate of the true value of $\boldsymbol{\beta}$. From the simulations that converged, which made up over 90% of the fits for each model, we computed the mean of the 1,000 estimates of each parameter to get final estimates of $\hat{\boldsymbol{\beta}}$ for each model, which are given in Table 2.

We use the estimates given in Table 2 to simulate from models M1 through M6. We discuss the simulation procedure and how we use the simulations in Section 4.

4 Experiment Set-Up

We want to explore three different aspects of the SAOM models using the lineup protocol: significance of parameters, goodness-of-fit of a model, and visual detection of additional parameters. Each on of these requires a different setup, which we describe in detail, but we use the lineup protocol to examine each aspect. In each lineup, there are two models shown: the null model and the alternative model. The definition of the null and alternative model will vary according to which aspects of the SAOMs we are exploring.

To construct a lineup, we simulate five networks from the null model and one network from the alternative model. We chose to expose our participants to only six plots at a time



Picking Lineups

Which model are you designating?

○ Null (simulate M-1 new plots)

● Alternative (simulate 1 new plot)

Select an effect to test:

simttb

Pick a wave:

1

Select a parameter (density, reciprocity, or both) for basic model to test. Select none for all other models:

none

Choose size of lineup:

6

Choose effect multiplier:

2

Set a random seed (10,000-999,999):

123456

Select a layout algorithm

Kamada-Kawai

□ Check box to color clusters.

Generate lineup!

Figure 3: A screen grab of the web application we created to design our lineup experiment. More details about this application are given in Section 4.3. In the lineup, M4 is the alternative model and one estimate from model M4 with β_5 set to twice its estimated value given in Table 2 is placed at random among five observations simulated from the null model, M1. Which plot comes from model M4?

in order to show the node-link diagrams in more detail and to lower their cognitive load. Typically, lineup experiments are done with sets of 20 plots at a time c.f. Loy et al. (2015); Vander Plas and Hofmann (2015), but we determined that not enough structure could be shown in each plot for 20 node-link diagrams. An example of a lineup like those shown to our participants is shown on the right side of the image in Figure 3. In this lineup, model M4 is the alternative model, and model M1 is the null model. To simulate lineups from the models, we set the parameters to the values given in Table **??** for all parameters within the respective models, with the exception of β_5. For β_5, twice the estimated value was used. More detail on why we use twice the fitted value is provided in Section 4.3.[2]. To get the simulations, we used the `siena07` function in `RSiena` (Ripley et al., 2013).

We want to explore the role of each of these parameters in the objective functions for each model.

[2] If you would like to explore the kinds of lineups we use in further detail, please visit `https://sctyner.shinyapps.io/saom_lineup_creation/`

4.1 Significance Testing

4.2 Goodness-of-Fit

4.3 Visual Detection

Through visual inference, we want to determine at which point an effect becomes noticeable in a SAOM. By *noticeable*, we mean that the inclusion of the effect alters the appearance of networks simulated from a model. An effect becomes noticeable when a simulation from a model with a noticeable effect appears different from simulations from a model where the same affect is absent. In model M1, with only two parameters in the objective function, we varied both the density and reciprocity parameteter values one at a time. In models M2 through M5, we vary the additional parameter, β_3 through β_6. Thus, we have six different parameters of interest to us: $\beta_1, \ldots, \beta_6$. We want to determine how the size of these parameters affects the overall structure of the network data simulated from the models M1 through M5, so we also vary the value of the parameters in two directions: positive and negative.

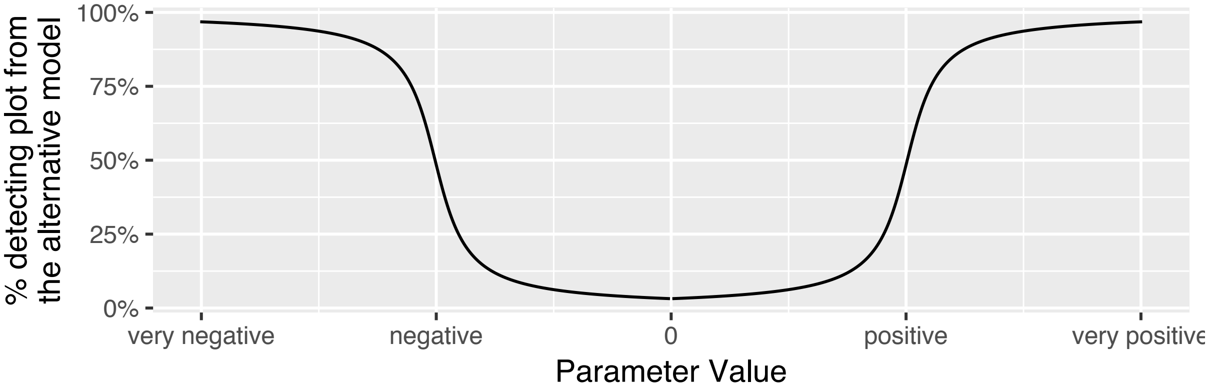


Figure 4: We hypothesize that as the parameter value of interest increases in absolute value, more viewers of the lineup will pick the alternative data out of a lineup.

To determine the threshold at which the effect becomes noticeable, we examined six different levels of the effect, three negative and three positive. We hypothesized that the perception of the effect by our participants would look something like Figure 4. The higher in absolute value the parameter is, the more likely participants are to choose the alternative model out of the lineup. To determine the six levels of the effects we want to test, started with the estimates of the parameter at hand as determined by the procedure described in Section 3.4, and multiply it or changing its sign to determine what *we* notice about the effect of the parameter in simulations from the changed models.

In order to determine which specific values of the parameters we wanted to examine, we constructed an online application that created the lineup protocol for us to be the guinea pigs of our own experiment (Chang et al., 2017). A screen grab of the app is shown in Figure 3. On the left side of the screen, the user[3] can input the information necessary for creating a lineup of the models M1 through M6 for the data in Section 3.3: first, choose to simulate only one plot from the specified model (analogous to changing the alternative model in the lineup protocol) or to simulated $M - 1$ plots from the specified model (analogous to changing the null model in the lineup protocol); the model of interest; the wave of the data to examine; if model M1 is selected, whether to alter the density or the reciprocity parameter; the size of the lineup; the amount by which to multiply the effect selected; a random seed for replicability; and a layout algorithm to use for the node-link diagrams. The plot(s) that appear(s) that are *not* from the model specified using the other options are simulations from model M1 with the estimates of the rate parameters, β_1 and β_2 as calculated by the procedure described in Section 3.4.

Using the "Picking Lineups" web application, we settled on six parameter values to test for each of our six effects, $\beta_1, \ldots \beta_6$. The complete details of the parameters tested using the lineup protocol is given in Table 3. In the case of both β_4 and β_6, we could not determine a value for negative effects that made the data simulated from M3 and M5 look different than null model simulations from model M1. Because we could not detect these effects at these values, we decided that the less experienced Turkers would also not be able to. So instead of testing the negative values of these effects, we examined a different scenario: we placed 5 simulations from positive values of the parameter with one simulation from model M1 in a lineup. We did this to determine if the perception of the effect size would be symmetric: if an effect is noticed $x\%$ of the time at value $\beta = 5$ when a simulation from the model is placed among null plots, then when five simulations from the model are put together with one simulation from model M1, the plot from the simpler model should be

[3] Please visit `https://sctyner.shinyapps.io/saom_lineup_creation/` to create lineups for yourself.

noticed about $x\%$ of the time as well.

	Parameter	Lineup Type	Easy Value	Medium Value	Hard Value
1	beta1	-1	-7.354	-6.6187	-5.883
2	beta1	1	-3.922	-4.1674	-4.412
3	beta3	-1	-17.249	-10.3497	-3.450
4	beta3	1	10.350	6.8998	5.175
5	beta4	-1	8.351	6.6806	5.010
6	beta4	1	6.681	5.0105	3.340
7	beta2	-1	0.000	0.0005	0.049
8	beta2	1	7.340	6.8504	6.361
9	beta6	-1	5.316	3.9872	3.323
10	beta6	1	5.316	3.9872	3.323
11	beta5	-1	-30.272	-20.1817	-10.091
12	beta5	1	20.182	17.6590	16.145

Table 3: All conditions in our MTurk experiment. Note that for β_4 and β_6, the negative type experiments are the reverse lineups, where 5 plots were simulated from the model with the parameter value given in the table, while 1 plot was simulated from M1.

5 Experiment Results

We recruited 250 participants for our experiment through Amazon Mechanical Turk. Each participant read over some brief training material before beginning the experiment. After reading this material and agreeing to participate, the participants saw two trial plots, one where they were required to select the plot that was most different from the others due to its complex structure, and one where they were required to select the plot that was most different from the others due to its very simple structure. Upon getting the trial plots correct, they moved into the experiment. Each participant was randomly assigned 13 lineups to look at. The were asked to select one or more plots that they perceived as "most different" from the others, and provide a reasoning for their choice. They could select from

"Most simple overall structure," "Most complex overall structure," or "Other" and provide their own text description of their reasoning. One of the 13 lineups the participants saw was the true data from the 112th senate shown in Section 3.3. Each participant only saw the data one in order to avoid bias.

5.1 Significance Testing

For a SAOM, there are two ways a significance test of the parameters can be performed. In `RSiena`, there are t-type tests and Wald-type test for a single parameter and for multiple parameters. The t-type test statistic is simply the parameter estimate divided by its standard error, and compared to a standard normal distribution. The Wald-type test statistic for a single parameter, β_k is

$$\frac{(\hat{\beta_k})^2}{var(\hat{\beta_k})} \sim \chi^2_1, \tag{6}$$

which is compared to a Chi-square distribution with one degree of freedom. Testing the significance of multiple parameters depends on the hypothesis we wish to test, and a $P \times K$ matrix, A, must be appropriately designed to test the P hypotheses of interest. The null hypothesis is that $A\boldsymbol{\beta} = \mathbf{0}$, and the test statistic is

$$(A\hat{\boldsymbol{\beta}})'\hat{\Sigma}^{-1}A\hat{\boldsymbol{\beta}} \sim \chi^2_p, \tag{7}$$

where $\hat{\Sigma}$ is the estimated covariance matrix of $\boldsymbol{\beta}$. This statistic is then compared to a Chi-square distribution with P degrees of freedom.

5.2 Goodness-of-Fit Testing

Goodness-of-fit testing for network models is notoriously difficult. Most network models, other than the most simple, lack the asymptotics required to develop the goodness-of-fit methods required (Goldenberg et al., 2010). Some methods have been developed based on what Snijders et al call "auxiliary statistics" such as the indegree or outdegree distribution on the nodes. In `RSiena`, the `sienaGOF` function performs goodness-of-fit testing as follows:

1. Auxiliary statistics are computed on the observed data ($\mathbf{u}_d$) and on N simulated observations from the model ($\mathbf{u}_1 \ldots \mathbf{u}_N$). (Usually, $N = 1000$)

2. The mean vector, $\overline{\mathbf{u}}$ and covariance matrix, $\mathbf{S}$ of the statistics on the simulations from the model are computed, and the Mahalanobis distance, $d_M(\mathbf{u})$ from the observed statistics to the distribution of the simulated statistics is computed:

$$d_M(\mathbf{u}) = \sqrt{(\mathbf{u} - \overline{\mathbf{u}})' S^{-1} (\mathbf{u} - \overline{\mathbf{u}})} \tag{8}$$

3. The Mahalanobis distance for each of the N simulations is calculated and $d_M(\mathbf{u}_d)$ is compared to this distribution of distances.

4. An empirical p-value is found by computing the proportion of simulated distances found in step 4 that are as large or larger than $d_M(\mathbf{u}_d)$. A SAOM is thus considered a good fit to the data if p is large. A plot comparing the data to the simulations is also considered, and a similar plot is shown in Figure 5

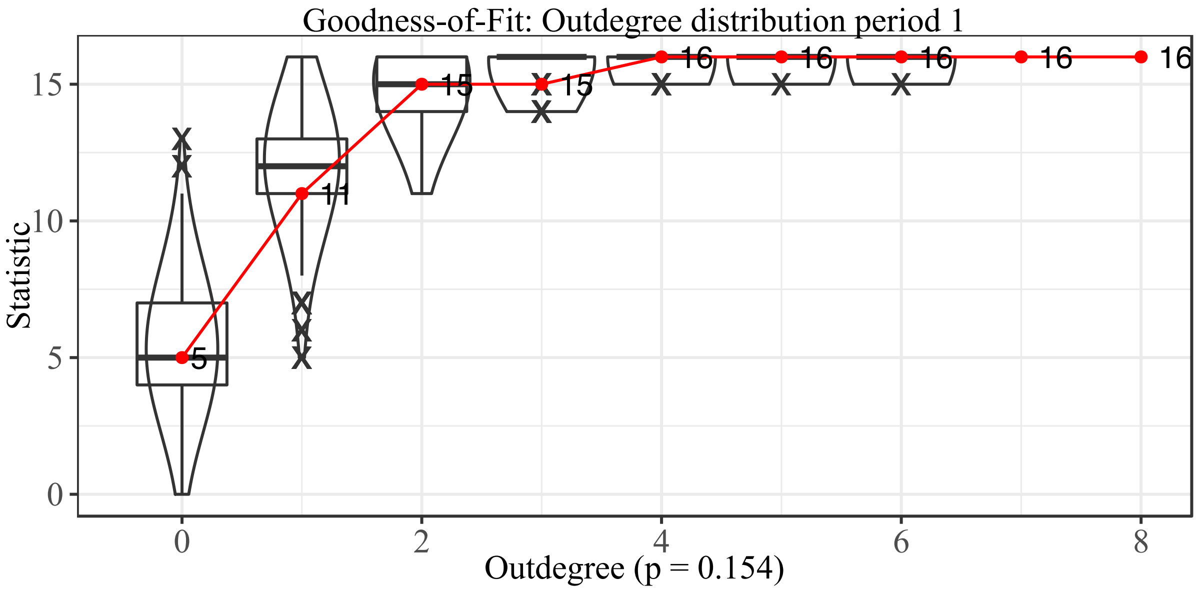


Figure 5: An example of what a goodness-of-fit plot from `RSiena` looks like. The overlaid boxplots and violin plots show the distribution of each of the outdegree values on the simulated networks, and the red points and lines are the observed data values.

The `RSiena` software also provides a Rao score-type test for goodness-of-fit for assessing one or more parameters, the test statistic of which is compared to a Chi-square distribution with P degrees of freedom, where P has the same definition as in Section 5.1. For full detail on the score-type test, see Schweinberger (2012).

These methods are similar in that they are both restriced: the `sienaGOF` method only considers one measure on the data and simulations from the model, while the score-type tests only consider subsets of parameters, "nuisance parameters" in Schweinberger (2012), not the entire set of parameters. By using visual inference instead of more traditional statistical methods, we hope to perform a more holistic goodness-of-fit test.

Using the lineup protocol, we show each Amazon Mechanical Turk worker the data once, in a lineup with five other plots of simulated data from one of the models we chose. We examined four different models, M2, M3, M4, and M6, and examined three repititons of each, for a total of 12 goodness-of-fit lineups. In each lineup, the "null model" is one of the four models and the "alternative" model is the true, unknown model that generated the senate network data The hypotheses for our goodness-of-fit tests are:

H_0: The senate network data come from the null model.

H_A: The senate network data do not come from the null model.

If a lineup viewer picks out the data among the five simulations from the null model, that is evidence in favor of the alternative hypothesis. On the contrary, if the lineup viewer picks one of the null plots, that is evidence against the alternative hypothsis. Because the size of the lineups is small, the probability of picking the data by chance is high, $\frac{1}{6}$, but if *many* independent viewers pick out the data from the nulls, the evidence in favor of the alternative hypothesis becomes stronger. Results from our MTurk goodness-of-fit plots are provided in Table 4.

The p-values were calcluated using the `vinference` package by Hofmann and Röttger (2016). This package contains methods to calculate *Visual distributions* for lineup experiment data. The distribution depends on the number of evaluations of a plot, K, the size of the lineup, m, and the lineup scenario, which here is that each lineup containing the same data and the same set of null plots is shown to K independent observers. Using these p-values, all but one lineup results in a rejection of the null hypothesis at Type-I error rate of $\alpha = 0.05$. The lineup that resulted in a failure to reject the null hypothesis is shown in Figure 6. The null model in this lineup is M4, and the senate data is shown in panel number $3^2 - 7$, but the panel most chose was number four, and the most common reasoning for that choice was that it had the most simple structure.

Model	Replicate	Data Picks	Total Viewers	p-value
M3	1	29	36	< 0.0001
M3	2	13	18	4e-04
M3	3	16	20	< 0.0001
M4	1	13	16	< 0.0001
M4	2	7	20	0.115
M4	3	29	34	< 0.0001
M5	1	9	21	0.0414
M5	2	21	24	< 0.0001
M5	3	14	16	< 0.0001
M7	1	17	20	< 0.0001
M7	2	14	28	0.0093
M7	3	28	37	< 0.0001

Table 4: An overview of the results from our 12 goodness-of-fit lineup tests.

The smallest p-value for one of the goodness-of-fit lineups was for the third replicate of the null model M4. This contradicts our previous finding that the only lineup to fail to reject the null was also when the null model was M4. The plot of this lineup is shown in Figure 7. In the remaining replicate of M4 as the null model, 13 of 16 viewers identified the data plot, corresponding to a p-values of less than 0.0001, just like the third replicate. This leads us to believe that the high p-value observed in the second replicate is more of a fluke. Furthermore, for the first lineup replicate with model M5 as the null model, only 9 of 21 participants were able to pick out the data, but the Visual distribution p-value is less than 0.05. This is much higher than the corresponding Binomial probability of $Pr(X = 9|n = 21, p = \frac{1}{6}) = \binom{21}{9}(\frac{1}{6})^9(\frac{5}{6})^{21-9} \approx 0.003$, but we are not sure that having fewer than half of our participants identify the data plot is a truly significant result. We could move the threshold from $\alpha = 0.05$ to $\alpha = 0.01$ or $\alpha = 0.001$, but more work needs to be done to determine which, if any of these, is appropriate.

We believe this goodness-of-fit testing method holds promise for the future of social network analysis. The participants in our experiments are very good overall at picking out

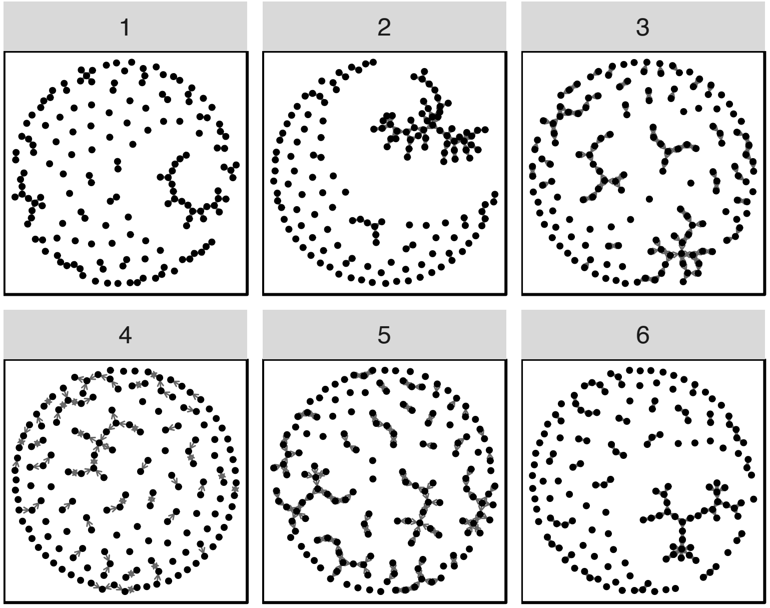


Figure 6: The goodness-of-fit lineup that failed to reject the null hypothesis. The null model for this lineup is M4. Only 7 of 21 viewers of this lineup selected the data plot as the most different from the others.

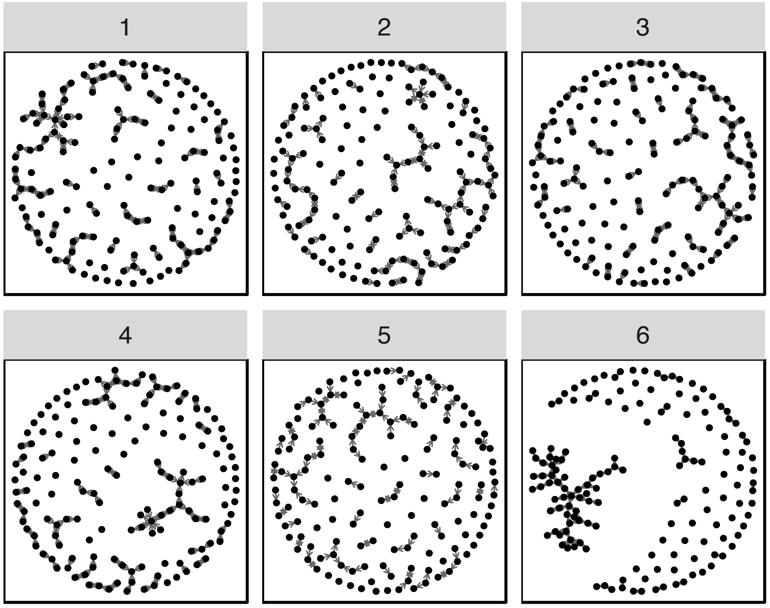


Figure 7: The lineup resulting in the smallest p-value rejecting the null hypothesis. Surprisingly, this another repetition for M4 as the null model.

the data when it is noticeably different from the null plots in the lineups. In addition, as in replicate three for null model M4, when the null plots contain similarly sized structures as the data plot, our participants have a hard time distinguishing the data. We believe that running these tests multiple times using several different sets of null models to adequately explore the possible structures generated by the models is a step in the right direction for a more comprehensive goodness-of-fit test for network models.

5.3 Visual Detection

A summary of the results from our experiment is shown in Figure 8. On the x axis, we plot the value of the parameter of interest, and on the y axis, the proportion of times the plot of data simulated from the alternative model was picked out for each lineup. The results are split into groups based on the sign of the parameter. We can see clear patterns in all of the groups: as the parameter value approaches 0, fewer participants identified the alternative plot.

We futher explore this relationship between identification of the alternative data in the lineup and the parameter, effect size, and lineuptype with a generalized linear mixed model. The response variable, $Y_{ijk\ell m}$, is binary, indicating whether participant m picked the alternative data plot in lineup ℓ. The covariate $x_{\ell 1}$ takes on the value -1 or 1 according to the lineup type code in Table 3. In most cases, -1 indicates the parameter value is less than original estimate, while 1 indicates the parameter value is at or above the original estimate. The continuous covariate $x_{\ell 2}$ is the centered and scaled size of the effect of interest from which the alternative data were simulated, the values of which are labeled “easy”, “medium”, and “hard” in Table 3 according to how difficult we thought the Turk participants would find each lineup. In Equation 9, $i \in \{1, 2, 3, 4, 5\}$ corresponding to the effects $\beta_3, \beta_4, \beta_2, \beta_6, \beta_5$, respectively. We include random effects in the model for each lineup, δ_ℓ and each participant, ϵ_m, and fit a hierarchical model as follows:

$$
\begin{aligned}
Y_{i\ell m} &\sim \text{Bernoulli}(\pi_{i\ell m}) && (9)\\
\text{logit}(\pi_{i\ell m}) &= \mu + \alpha_i + \theta\mathbb{I}(x_{\ell 1} = 1) + \gamma x_{\ell 2} + && (10)\\
&\quad (\alpha\theta)_i\mathbb{I}(x_{\ell 1} = 1) + (\alpha\gamma)_i x_{\ell 2} + (\theta\gamma)\mathbb{I}(x_{\ell 1} = 1)x_{\ell 2} + && (11)\\
&\quad (\alpha\theta\gamma)_i\mathbb{I}(x_{\ell 1} = 1)x_{\ell 2} + \delta_\ell + \epsilon_m && (12)\\
\delta_\ell &\sim N(0, \sigma_\delta^2) && (13)\\
\epsilon_m &\sim N(0, \sigma_\epsilon^2) && (14)\\
& && (15)
\end{aligned}
$$

The results of fitting this model using `glmer` from the `lme4` package are summarized in Table 5 (Bates et al., 2015).

6 Discussion

By using visual inference methods, we have developed new ways to perform significance and goodness-of-fit testing for a complicated and intractable set of statistical models for social network data. These new methods can be used to supplement traditional methods and check our assumptions about network models. The traditional methods only look at one piece or one measure of a network model, but our methods look at the models holistically for a broader sense of what it means for a parameter to be significant or a model to be a good fit. By looking at an entire network simulated from a SAOM side-by-side with other instances of networks simulated from another model, we develop an idea of the model in terms of the data itself, instead of in terms of statistical summaries of the data.

We hope to apply these methods further for different types of network data and different types of network models. We accept the limitations of this type of network data visualization, in that even in small instances, the cognitive load of looking at a lineup is very high for the average observer. We would therefore like to explore larger datasets, different layout algorithms, and different ways of visualizing network data, such as adjacency matrix visualizations, through visual inference to see if similar patterns emerge.

Parameter	Estimate	Std Error	p-value	Odds Multiplier
μ	-9.934	3.166	0.002	0
α_1	7.29	3.263	0.026	1465.131
α_2	7.242	3.436	0.035	1396.885
α_3	-7.062	3.365	0.036	9×10^{-4}
α_4	4.004	3.45	0.246	54.828
α_5	5.247	3.277	0.109	190.071
θ	47.232	6.015	0	3.254×10^{20}
γ	-14.504	4.378	9×10^{-4}	0
$(\alpha\theta)_1$	-47.389	6.14	0	0
$(\alpha\theta)_2$	-46.618	6.214	0	0
$(\alpha\theta)_3$	-37.069	7.482	0	0
$(\alpha\theta)_4$	-42.466	6.368	0	0
$(\alpha\theta)_5$	-48.386	6.758	0	0
$(\alpha\gamma)_1$	13.396	4.417	0.002	6.576×10^5
$(\alpha\gamma)_2$	16.907	4.883	5×10^{-4}	2.201×10^7
$(\alpha\gamma)_3$	-214.581	22.108	0	0
$(\alpha\gamma)_4$	29.596	5.581	0	7.138×10^{12}
$(\alpha\gamma)_5$	12.328	4.394	0.005	2.26×10^5
$\theta\gamma$	89.864	11.939	0	1.065×10^{39}
$(\alpha\theta\gamma)_1$	-84.281	12.033	0	0
$(\alpha\theta\gamma)_2$	-88.02	12.36	0	0
$(\alpha\theta\gamma)_3$	152.993	22.996	0	2.778×10^{66}
$(\alpha\theta\gamma)_4$	-99.196	12.682	0	0
$(\alpha\theta\gamma)_5$	-84.424	12.081	0	0
σ^2_δ	0.5638			
σ^2_ϵ	0.3416			

Table 5: Summary of the results from fitting the model given in Equation 9

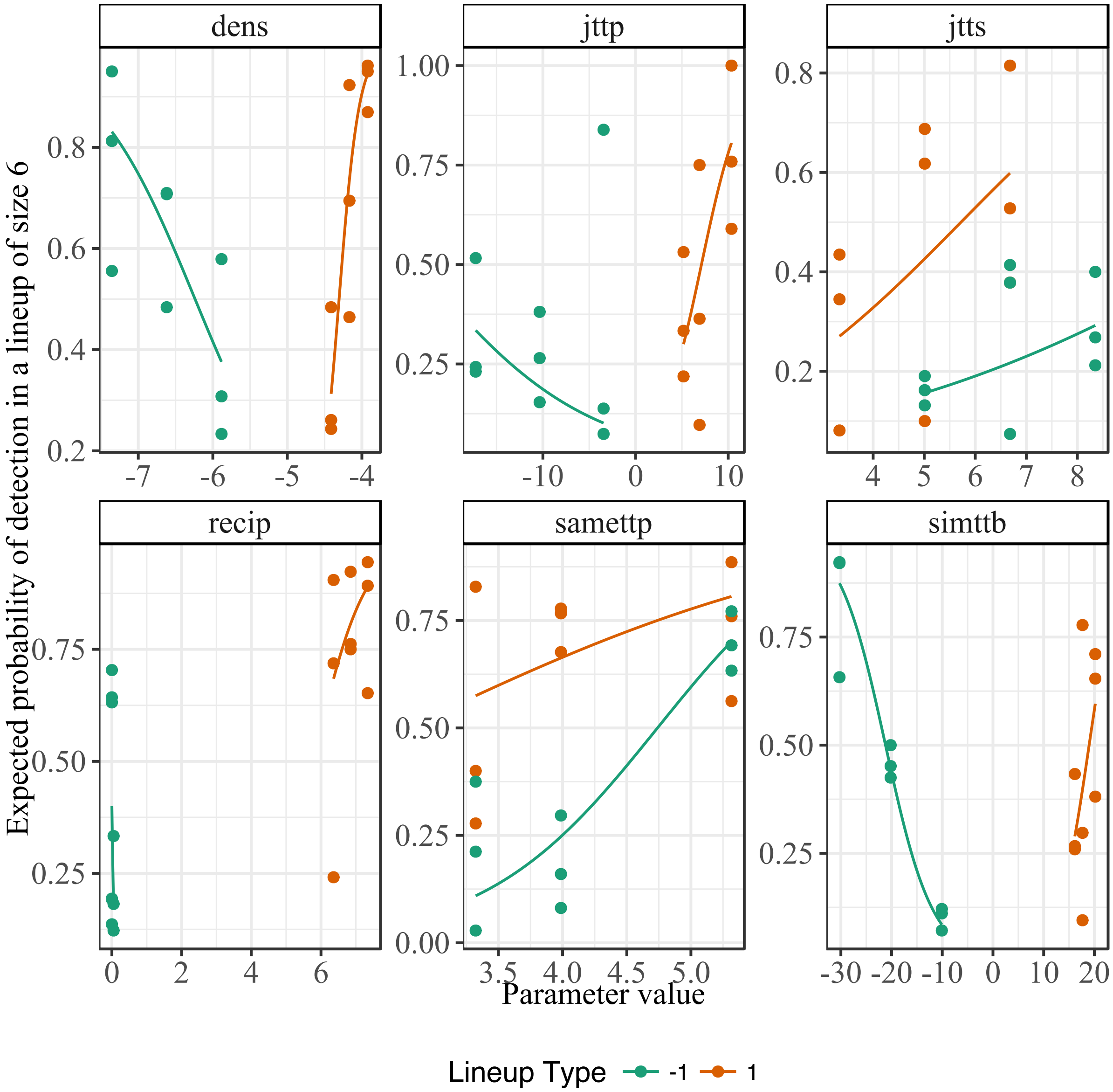


Figure 8: Predictions from our generalized linear mixed effects model given in Equation efeq:glmm. The lines show the expected probability of detecting the alternative data in a lineup of size 6 for new observers of new lineups is plotted on the y-axis, and the size of the parameter of interest is on the x-axis. The proportions detected by our Turk participants for each lineup group are shown by the points.

References

Amazon (2010).

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015), “Fitting Linear Mixed-Effects Models Using lme4,” *Journal of Statistical Software*, 67, 1–48.

Buja, A., Cook, D., Hofmann, H., Lawrence, M., Lee, E., Swayne, D., and Wickham, H. (2009), “Statistical Inference for Exploratory Data Analysis and Model Diagnostics,” *Royal Society Philosophical Transactions A*, 367, 4361–4383.

Chang, W., Cheng, J., Allaire, J., Xie, Y., and McPherson, J. (2017), *shiny: Web Application Framework for R*, r package version 1.0.3.

Gelman, A. (2004), “Exploratory Data Analysis for Complex Models,” *Journal of Computational and Graphical Statistics*, 13, 755–779.

Goldenberg, A., Zheng, A. X., Fienberg, S. E., and Airoldi, E. M. (2010), “A Survey of Statistical Network Models,” *Foundations and Trends in Machine Learning*, 2, 129–233.

Good, P. (2005), *Permutation, Parametric, and Bootstrap Tests of Hypotheses*, New York: Springer.

Gross, J. H., Kirkland, J. H., and Shalizi, C. R. (2008), “Cosponsorship in the U.S. Senate: A Multilevel Two-Mode Approach to Detecting Subtle Social Predictors of Legislative Support,” *Unpublished Manuscript*.

Hofmann, H. and Röttger, C. (2016), *vinference: Inference under the lineup protocol*, r package version 0.1.1.

Loy, A., Follett, L., and Hofmann, H. (2015), “Variations of Q-Q Plots – the Power of our Eyes!” *The American Statistician*, 2015, 1–36.

Majumder, M., Hofmann, H., and Cook, D. (2013), “Validation of Visual Statistical Inference, Applied to Linear Models,” *Journal of American Statistical Association*, 108, 942–956.

Nils Ringe, J. N. V. and Cho, W. T. (2016), *The Oxford Handbook of Political Networks*, Oxford University Press, chap. Legislative Networks.

Ripley, R., Boitmanis, K., and Snijders, T. A. (2013), *RSiena: Siena - Simulation Investigation for Empirical Network Analysis*, r packa ge version 1.1-232.

Schweinberger, M. (2012), "Statistical modelling of network panel data: Goodness of fit," *British Journal of Mathematical and Statistical Psychology*, 65, 262–281.

Snijders, T., Steglich, C., and Schweinberger, M. (2007), *Longitudinal Models in the Behavioral and Related Sciences*, Lawrence Erlbaum Associates, chap. Modeling the Coevolution of Networks and Behavior.

Snijders, T. A. (2001), "The Statistical Evaluation of Social Network Dynamics," *Sociological Methodology*, 31, 361–395.

— (2017), "Stochastic Actor-Oriented Models for Network Dynamics," *Annual Review of Statistics and Its Application*, 4, 343–63.

Snijders, T. A., van de Bunt, G. G., and Steglich, C. E. (2010a), "Introduction to stochastic actor-based models for network dynamics," *Social Networks*, 32, 44 – 60, dynamics of Social Networks.

Snijders, T. A. B. (1996), "Stochastic actor-oriented models for network change," *Journal of Mathematical Sociology*, 21, 149–172.

Snijders, T. A. B., Koskinen, J., and Schweinberger, M. (2010b), "Maximum likelihood estimation for social network dynamics," *The Annals of Applied Statistics*, 4, 567–588.

Vander Plas, S. and Hofmann, H. (2015), "Clusters beat Trend!? Testing feature hierarchy in statistical graphics," *Journal of Computational and Graphical Statistics*, submitted.