A Continuous-time Multicast Network Model

Bomin Kim[*]

Department of Statistics, Pennsylvania State University,

Aaron Schein

College of Information and Computer Sciences, UMass Amherst,

Bruce Desmarais

Department of Political Science, Pennsylvania State University,

and

Hanna Wallach

Microsoft Research NYC

May 2, 2018

Abstract

We introduce the continuous-time multicast network model—a generative model for directed edges with multiple recipients or multiple senders. To define the model, we integrate a dynamic version of the exponential random graph model (ERGM) and generalized linear model approach to understand who communicates with whom, and when. We use the model to analyze emails sent between department managers in Montgomery county government in North Carolina, and international sanctions data concerning the threats and use of economic sanctions by states. Our applications demonstrate that the model is effective at predicting and explaining continuous-time networks involving edges with multiple recipients or senders.

Keywords: 3 to 6 keywords, that do not appear in the title

1   Introduction

In recent decades, real-time digitized textual communication has developed into a ubiquitous form of social and professional interaction (Kanungo & Jain 2008, Szóstek 2011, Burgess et al. 2004, Pew 2016). From the perspective of the computational social scientist, this has lead to a growing need for methods of modeling interactions that manifest as text exchanged in continuous time. A number of models that build upon topic modeling through Latent Dirichlet Allocation (Blei et al. 2003) to incorporate link data as well as textual content have been developed recently (McCallum et al. 2005, Lim et al. 2013, Krafft et al. 2012). These models are innovative in their extensions that incorporate network tie information. However, none of the models that are currently available in the literature integrate the rich random-graph structure offered by state of the art models for network structure—such as the exponential random graph model (ERGM) (Robins et al. 2007, Chatterjee et al. 2013, Hunter et al. 2008). The ERGM is the canonical model for modeling the structure of a static network. It is flexible enough to specify a generative model that accounts for nearly any pattern of tie formation (e.g., reciprocity, clustering, popularity effects) (Desmarais & Cranmer 2017). Several models have been developed that handle time-stamped ties in which tie formation is governed by structural dynamics similar to those used in ERGMs (Perry & Wolfe 2013, Butts 2008, Snijders 1996), however, none of the models explicitly allows multicast—edge with single sender multiple recipients—which is a common property of digitalized textual interactions such as emails and online messages. We develop the continuous-time multicast network model which simultaneously models the multiple recipient selection processs and the time-to-next interactions which jointly govern time-stamped tie formation.

2   A Continuous-time Multicast Network Model

Data generated under the model consists of D unique edges. A single edge, indexed by $d \in [D]$, is represented by the three components: the sender $a_d \in [A]$, an indicator vector of recipients $\boldsymbol{r}_d = \{u_{dr}\}_{r=1}^{A}$, and the timestamp $t_d \in (0, \infty)$. For simplicity, we assume that edges are ordered by time such that $t_d \leq t_{d+1}$.

2.1 Tie Generating Process

For every possible author–recipient pair $(a, r)_{a \neq r}$, we define the "recipient intensity", which is the likelihood of edge d being sent from a to r:

$$\lambda_{adr} = \boldsymbol{b}^\top \boldsymbol{x}_{adr}, \tag{1}$$

where $\boldsymbol{b}$ is P–dimensional vector of coefficients and $\boldsymbol{x}_{adr}$ is a set of network features which vary depending on the hypotheses regarding canonical processes relevant to network theory such as popularity, reciprocity, and transitivity. In addition, we include intercept term to account for the average (or baseline) number of recipients. We place a Normal prior $\boldsymbol{b} \sim N(\boldsymbol{\mu}_b, \Sigma_b)$.

Next, we hypothesize "If a were the sender of edge d, who would be the recipient(s)?" To do this, we draw each sender's set of recipients from a non-empty Gibbs measure Fellows & Handcock (2017)—a probability measure we defined in order to 1) allow multiple recipients or "multicast", 2) prevent from obtaining zero recipient, and 3) ensure tractable normalizing constant. Because we allows multicast, we draw a binary (0/1) vector $\boldsymbol{u}_{ad} = (u_{ad1}, \dots, u_{adA})$

$$\boldsymbol{u}_{ad} \sim \text{Gibbs}(\boldsymbol{\lambda}_{ad}), \tag{2}$$

where $\boldsymbol{\lambda}_{id} = \{\lambda_{adr}\}_{r=1}^A$. In particular, we define $\text{Gibbs}(\boldsymbol{\lambda}_{ad})$ as

$$p(\boldsymbol{u}_{ad}|\delta, \boldsymbol{\lambda}_{ad}) = \frac{\exp\left\{\log\left(\text{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\right) + \sum_{r \neq a} \lambda_{adr} u_{adr}\right\}}{Z(\boldsymbol{\lambda}_{ad})}, \tag{3}$$

where $Z(\boldsymbol{\lambda}_{ad}) = \prod_{r \neq a}\left(\exp(\lambda_{adr})+1\right)-1$ is the normalizing constant and $\|\cdot\|_1$ is the l_1–norm. This is equivalent to assume independent Bernoulli trial on each u_{adr} with probability of 1 being $\text{logit}(\lambda_{adr})$, excluding the case when all $u_{adr} = 0$. We provide the derivation of the normalizing constant as a tractable form in the supplementary material.

2.2 Time Generating Process

Similarly, we hypothesize "If a were the sender of edge d, when would it be sent?" and define the "timing rate" for sender a

$$\mu_{ad} = g^{-1}(\boldsymbol{\eta}^\top \boldsymbol{y}_{ad}), \tag{4}$$

where $\boldsymbol{\eta}$ is Q–dimensional vector of coefficients with a Normal prior $\boldsymbol{\eta} \sim N(\boldsymbol{\mu}_\eta, \Sigma_\eta)$, $\boldsymbol{y}_{ad}$ is a set of time-related covariates, e.g., any feature that could affect timestamps of the edge, and $g(\cdot)$ is the appropriate link function such as identity, log, or inverse.

In modeling "when", we do not directly model the timestamp t_d. Instead, we assume that each sender's the time-increment or "time to next edge" (i.e., $\tau_d = t_d - t_{d-1}$) is drawn from a specific distribution in the exponential family. We follow the generalized linear model framework:

$$
\begin{aligned}
E(\tau_{ad}) &= \mu_{ad}, \\
V(\tau_{ad}) &= V(\mu_{ad}),
\end{aligned}
\tag{5}
$$

where τ_{ad} is a positive real number. Possible choices of distribution include Exponential, Weibull, Gamma, and lognormal[1] distributions, which are commonly used in time-to-event modeling. Based on the choice of distribution, we may introduce any additional parameter (e.g., σ_τ^2) to account for the variance. We use $f_\tau(\cdot; \mu, \sigma_\tau^2)$ and $F_\tau(\cdot; \mu, \sigma_\tau^2)$ to denote the probability density function (p.d.f) and cumulative density function (c.d.f), respectively, with mean μ and variance σ^2.

2.3   Observed Data

Finally, we choose the sender, recipients, and timestamp—which will be observed—by selecting the sender–recipient-set pair with the smallest time-increment Snijders (1996):

$$
\begin{aligned}
a_d &= \operatorname{argmin}_a(\tau_{ad}), \\
\boldsymbol{r}_d &= \boldsymbol{u}_{a_d d}, \\
t_d &= t_{d-1} + \tau_{a_d d}.
\end{aligned}
\tag{6}
$$

Therefore, it is a sender-driven process in that the recipients and timestamp of an edge is determined by the sender's urgency to send the edge to chosen recipients. Note that our generative process accounts for tied events such that in case of tied events (i.e., multiple senders generated same time increments τ_d), we observe all of the tied events[2] .

---

[1]lognormal distribution is not exponential family but can be used via modeling of $\log(\tau_d)$.

[2]The order of tied events do not matter.

Algorithm 1 Generating Process
---
for *d=1 to D* **do**
    for *a=1 to A* **do**
        for *r=1 to A* **do**
            if $r \neq a$ **then**
                form $\boldsymbol{x}_{adr}$
                set $\lambda_{adr} = \boldsymbol{b}^\top \boldsymbol{x}_{adr}$
            end
        end
        draw $\boldsymbol{u}_{ad} \sim \text{Gibbs}(\boldsymbol{\lambda}_{ad})$
        form $\boldsymbol{y}_{ad}$
        set $\mu_{ad} = g^{-1}(\boldsymbol{\eta}^\top \boldsymbol{y}_{ad})$
        draw $\tau_{ad} \sim f_\tau(\mu_{ad}, \sigma_\tau^2)$
    end
    For no tied events,
    set $a_d = \text{argmin}_a(\tau_{ad})$
    set $\boldsymbol{r}_d = \boldsymbol{u}_{a_d d}$
    set $t_d = t_{d-1} + \min_a \tau_{ad}$

    For *n* **tied events,** (i.e., *n* senders generated same minimum time increments)
    set $a_d, \ldots, a_{d+n} = \text{argmin}_a(\tau_{ad})$
    set $\boldsymbol{r}_d = \boldsymbol{u}_{a_d d}, \ldots, \boldsymbol{r}_{d+n} =, \boldsymbol{u}_{a_{d+n} d}$
    set $t_d, \ldots, t_{d+n} = t_{d-1} + \min_a \tau_{ad}$
    jump to $d = d + n + 1$
end
---

3   Posterior Inference

Our inference goal is to invert the generative process to obtain the posterior distribution over the unknown parameters, conditioned on the observed data and hyperparamters $(\boldsymbol{\mu}_b, \Sigma_b, \boldsymbol{\mu}_\eta, \Sigma_\eta)$. We draw the samples using Markov chain Monte Carlo (MCMC) methods, repeatedly resampling the value of each parameter from its conditional posterior given the observed data, hyperparamters, and the current values of the other parameters. We express each parameters conditional posterior in a closed form using the data augmentation schemes in $\boldsymbol{u}$ Tanner & Wong (1987). In this section, we provide each latent variable's conditional posterior.

First, since u_{adr} is a binary random variable, new values may be sampled directly using

$$P(u_{adr} = 1 | \boldsymbol{u}_{ad\backslash r}, \boldsymbol{b}, \boldsymbol{x}) \propto \exp(\lambda_{adr});$$

$$P(u_{adr} = 0 | \boldsymbol{u}_{ad\backslash r}, \boldsymbol{b}, \boldsymbol{x}) \propto \mathrm{I}(\|\boldsymbol{u}_{ad\backslash r}\|_1 > 0), \tag{7}$$

where $I(\cdot)$ is the indicator function that is used to prevent from the instances where a sender chooses zero number of recipients.

New values for continuous variables $\boldsymbol{b}$, and $\boldsymbol{\eta}$ and σ_τ^2 (if applicable) cannot be sampled directly from their conditional posteriors, but may instead be obtained using the Metropolis–Hastings algorithm. With uninformative priors (i.e., $N(0, \infty)$), the conditional posterior over $\boldsymbol{b}$ is

$$P(\boldsymbol{b}|\boldsymbol{u}, \boldsymbol{x}) \propto \prod_{d=1}^{D} \prod_{a=1}^{A} \frac{\exp\left\{ \log\left(\mathrm{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\right) + \sum_{r \neq a} \lambda_{adr} u_{adr} \right\}}{Z(\boldsymbol{\lambda}_{ad})}, \tag{8}$$

where the two variables share the conditional posterior and thus can be jointly sampled. Likewise, assuming uninformative priors on $\boldsymbol{\eta}$ (i.e., $N(0, \infty)$) and σ_τ^2 (i.e., half-Cauchy(∞)), the conditional posterior for no-tied event case is

$$P(\boldsymbol{\eta}, \sigma_\tau^2 | \boldsymbol{u}, \boldsymbol{y}) \propto \prod_{d=1}^{D} \left(f_\tau(\tau_d; \mu_{a_d d}, \sigma_\tau^2) \times \prod_{a \neq a_d} \left(1 - F_\tau(\tau_d; \mu_{ad}, \sigma_\tau^2)\right) \right). \tag{9}$$

Since we assme the existence of tied-event, the conditional posterior of $\boldsymbol{\eta}$ and σ_τ^2 should be written as

$$P(\boldsymbol{\eta}, \sigma_\tau^2 | \boldsymbol{u}, \boldsymbol{y}) \propto \prod_{m=1}^{M} \left(\prod_{d:t_d = t_m^*} f_\tau(t_m^* - t_{m-1}^*; \mu_{a_d d}, \sigma_\tau^2) \times \prod_{a \notin \{a_d\}_{d:t_d = t_m^*}} \left(1 - F_\tau(t_m^* - t_{m-1}^*; \mu_{ad}, \sigma_\tau^2)\right) \right), \tag{10}$$

where $t_1^*, \ldots, t_M^*$ are the unique timepoints across D events ($M \leq D$). If $M = D$ (i.e., no tied events), Equation (10) reduces to Equation (9).

6

Algorithm 2 MCMC Algorithm

---

set initial values of $\boldsymbol{b}, \boldsymbol{\eta}, \sigma_\tau^2$
for *o=1 to outer* **do**
    for *d=1 to D* **do**
        for *a = 1 to A* **do**
            for *r = a to A* **do**
                if $r \neq a$ **then**
                    update u_{adr} using Gibbs update — Equation (7)
                end
            end
        end
    end
    for *n=1 to inner1* **do**
        update $\boldsymbol{b}$ using Metropolis-Hastings — Equation (8)
    end
    for *n=1 to inner2* **do**
        update $\boldsymbol{\eta}$ and σ_τ^2 using Metropolis-Hastings — Equation (10)
    end
    for *n=1 to inner3* **do**
        update σ_τ^2 using Metropolis-Hastings — Equation (10)
    end
end
Summarize the results with:
last chain of $\boldsymbol{b}$, and last chain of $\boldsymbol{\eta}$ and σ_τ^2

---

4 Case Studies

We now present two case studies applying our method to 1) county government emails (single sender multiple recipients) and 2) international sanctions (multiple senders single recipient). For each case study, we formulate the network statistics $\boldsymbol{x}$ and timestamp statistics $\boldsymbol{y}$ and ground them in illustrative examples. We then report a suite of experiments that test our methods ability to form the posterior distribution over latent variables for different types of data.

4.1 Case Study 1: County Government Emails

4.1.1 Montgomary County Data

Our data come from the North Carolina county government email dataset collected by ben Aaron et al. (2017) that includes internal email corpora covering the inboxes and outboxes of managerial-level employees of North Carolina county governments. Out of over twenty counties, we chose Montgomery County to 1) test our model using data with a lot of multicast edges (16.76%), and 2) limit the scope of this initial application. The Montgomery County email network contains 680 emails, sent and received by 18 department managers over a period of 3 months (March–May) in 2012.

4.1.2 Covariates

In the example of email networks, we form the covariate vector $\boldsymbol{x}_{adr}$ using time-varying network statistics based on the time interval prior to and including t_{d-1}. Along with intercept term, we compute the network statistics (Perry & Wolfe 2013), where the time interval tracks 7 days prior to the last email was sent $l_d = (t_{d-1} - 7\text{days}, t_{d-1}]$.

1. intercept $\boldsymbol{x}_{adr1} = 1$;

2. outdegree $\boldsymbol{x}_{adr2} = \sum\limits_{d':t_{d'} \in l_d} I(a_{d'} = a)$;

3. indegree $\boldsymbol{x}_{adr3} = \sum\limits_{d':t_{d'} \in l_d} I(u_{d'r} = 1)$;

4. send $\boldsymbol{x}_{adr4} = \sum\limits_{d':t_{d'} \in l_d} I(a_{d'} = a)I(u_{d'r} = 1)$;

5. receive $\boldsymbol{x}_{adr5} = \text{send}(r, a)$;

6. 2-send $\boldsymbol{x}_{adr6} = \sum\limits_{h \neq a,r} \text{send}(a, h)\text{send}(h, r)$;

7. 2-receive $\boldsymbol{x}_{adr7} = \sum\limits_{h \neq a,r} \text{send}(h, a)\text{send}(r, h)$;

8. sibling $\boldsymbol{x}_{adr8} = \sum\limits_{h \neq a,r} \text{send}(h, a)\text{send}(h, r)$;

9. cosibling $\boldsymbol{x}_{adr9} = \sum\limits_{h \neq a,r} \text{send}(a, h)\text{send}(r, h)$;

10. Nreceive $\boldsymbol{x}_{adr10} = \sum\limits_{d':t_{d'} \in l_d} \sum\limits_{r=1}^{A} I(a_{d'} = a)I(u_{d'r} = 1)$;

11. outdegree∗Nreceive $\boldsymbol{x}_{adr11} = \boldsymbol{x}_{adr2} \times \boldsymbol{x}_{adr10}$

where $I(\cdot)$ is an indicator function, and the last covariate is the interaction term between outdegree and Nreceive.

For time-related covariates $\boldsymbol{y}_{ad}$ in Section 2.2, we form the covariate vector $\boldsymbol{y}_{ad}$ using the seven statistics which may possibly effect "time to send the next document". Note that these statistics could depend on a only, d (or t_d) only, or both a and d. Specifically, the statistics are

1. intercept $\boldsymbol{y}_{ad1} = 1$;

2. outdegree $\boldsymbol{y}_{ad2} = \sum\limits_{d':t_{d'} \in l_d} I(a_{d'} = a)$;

3. indegree $\boldsymbol{y}_{ad3} = \sum\limits_{d':t_{d'} \in l_d} I(u_{d'a} = 1)$;

4. gender $\boldsymbol{y}_{ad4} = I(a = \text{female})$;

5. manager $\boldsymbol{y}_{ad5} = I(a = \text{County Manager})$;

6. weekend $\boldsymbol{y}_{ad6} = I(t_{d-1} = \text{weekend})$;

7. am/pm $\boldsymbol{y}_{ad7} = I(t_{d-1} = \text{pm})$;
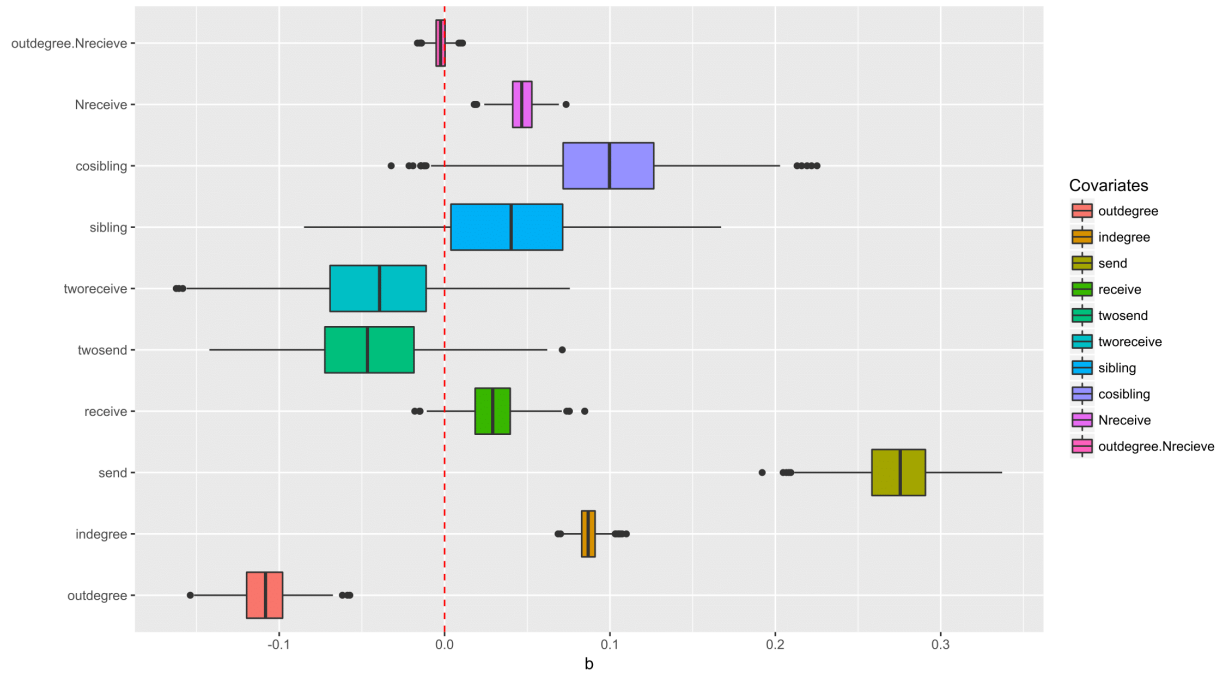
4.1.3 Parameter Estimation and Interpretation

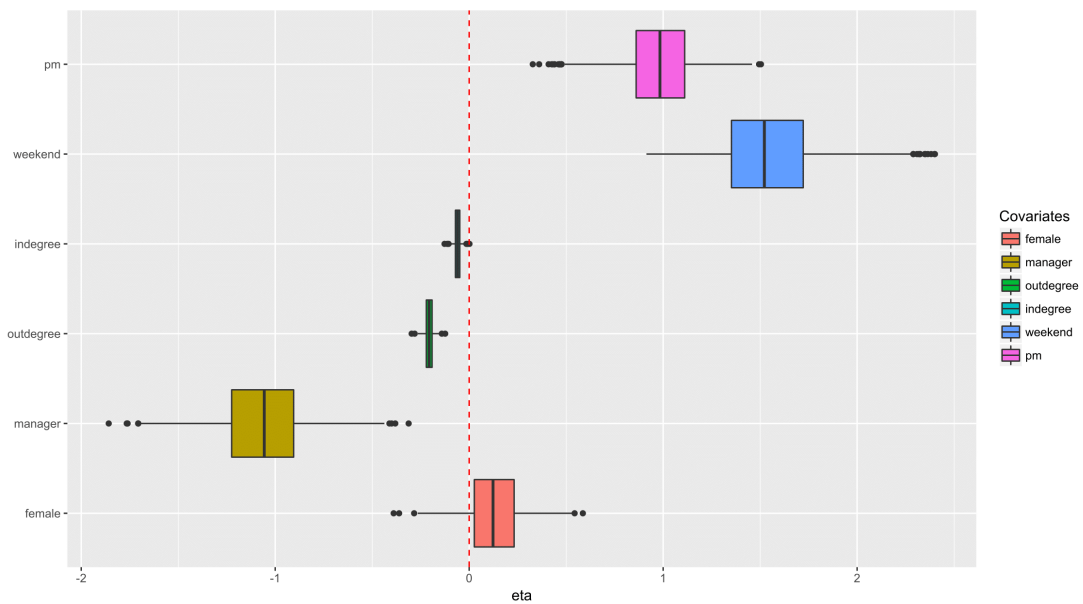Figure 1: Posterior distribution of $\boldsymbol{b}$ estimates.

Figure 2: Posterior distribution of $\boldsymbol{\eta}$ estimates.

4.2 Case Study 2: International Sanctions

4.2.1 Threat and Imposition of Sanctions (TIES) Data

We processed the threat and imposition of sanctions (TIES) data (Morgan et al. 2014) that includes the records of international sanctions—defined as actions that one or more countries take to limit or end their economic relations with a target country in an effort to persuade that country to change one or more of its policies—from 1945 to 2005.

4.2.2 Covariates

4.2.3 Parameter Estimation and Interpretation

4.3 Experiment

We evaluate the model's ability to predict ties from the Montgomery County email network and the threat and imposition of sanctions (TIES) data, conditioned on their "training" part of the data. We separately formed a test split of each three components—author, recipients, and timestamps—by randomly selecting "test" data with probability $p = 0.1$. Any missing variables were imputed by drawing samples from their conditional posterior distributions, given the observed data, estimates of latent variables, and current estimates of test data. The full conditional posterior distributions for "test" author, recipients, and timestamps are provided in the supplementary material. We then run inference to update the latent variables given the imputed and observed data. We iterate the two steps— imputation and inference—multiple times to obtain enough number of estimates for "test" data. Algorithm 3 outlines this procedure.

Algorithm 3 Out-of-Sample Tie Predictions

Input: data $\{(a_d, \boldsymbol{r}_d, t_d)\}_{d=1}^D$,
number of new data to generate R,
hyperparameters $(\boldsymbol{\mu}_b, \Sigma_b, \boldsymbol{\mu}_\eta, \Sigma_\eta)$

Test splits:
Draw test authors with $p = 0.1$ (out of D authors)
Draw test elements of recipient vector with $p = 0.1$ (out of $D \times (A - 1)$ receipient indicators $\{\{\boldsymbol{r}_{dr}\}_{r \in [A]\setminus a_d}\}_{d=1}^D$)
Draw test timestamps with $p = 0.1$ (out of D timestamps)
Set the "test" data as "missing" (NA)

Imputation and inference:
Initialize the parameters $(\boldsymbol{b}, \boldsymbol{\eta}, \boldsymbol{u}, \sigma_\tau^2)$
for $r = 1$ **to** R **do**
  for $d = 1$ **to** D **do**
    if $a_d = $ NA **then**
      for $a = 1$ **to** A **do**
        Compute π_a using $P(a_d = a|\cdot)$
      end for
      Draw $a_d \sim \text{Multinomial}(\pi)$
    end if
    for $r \in [A]_{\setminus a_d}$ **do**
      if $r_{dr} = $ NA **then**
        Draw r_{dr} using $P(r_{dr} = 1|\cdot)$ and $P(r_{dr} = 0|\cdot)$
      end if
    end for
    if $t_d = $ NA **then**
      Draw proposal $\boldsymbol{\tau}_d^{new} \sim f_\tau(\mu_{a_d d}, \sigma_\tau^2)$
      Use Metropolis-Hastings to decide accept or reject using the probability

$$\frac{P(\boldsymbol{\tau}_d^{new}|\mu_{a_d d}, \sigma_\tau^2) P(\boldsymbol{\tau}_d^{new}|\cdot)}{P(\boldsymbol{\tau}_d^{old}|\mu_{a_d d}, \sigma_\tau^2) P(\boldsymbol{\tau}_d^{old}|\cdot)},$$

      where $\boldsymbol{\tau}_d^{old}$ is from earlier iteration.
    end if
    Run inference and update $(\boldsymbol{b}, \boldsymbol{\eta}, \boldsymbol{u}, \sigma_\tau^2)$ given the imputed and observed data
  end for
  Store the estimates for "test" data
end for

NOTE: we assume independent Bernoulli trial on u_{adr}. Then, does it matter to have test 'elements' for the recipient vector instead of having the test 'recipient vectors'?

4.4 Posterior Predictive Checks

Finally, we perform posterior predictive checks Rubin et al. (1984) to evaluate the appropriateness of our model specification for the Montgomery County email data and the threat and imposition of sanctions (TIES) data. We formally generated entirely new data, by simulating ties and timestamps $\{(a_d, \boldsymbol{r}_d, t_d)\}_{d=1}^{D}$ from the genenerative process in Section 2, conditional upon a set of inferred parameter values from the inference in Section 3. For the test of goodness-of-fit in terms of network dynamics, we use multiple network statistics that summarize meaningful aspects of the Montgomery County email data and the threat and imposition of sanctions (TIES) data: indegree distribution, outdegree distribution, recipient size distribution, time-increments distribution, the edgewise shared partner distribution, and the geodesic distance distribution. We then generated 500 synthetic networks from the posterior predictive distribution.

Algorithm 4 Generate new data for PPC

  Input: number of new data to generate R, covariates $\boldsymbol{x}$ and $\boldsymbol{y}$
  estimated latent variables $(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{\eta}, \sigma_\tau^2)$

  for $r = 1$ **to** R **do**
    for $d = 1$ **to** D **do**
      Draw $(a_d, \boldsymbol{r}_d, t_d)$ following Section 2
    end for
    Store every r^{th} new data $\{(a_d, \boldsymbol{r}_d, t_d)\}_{d=1}^{D}$
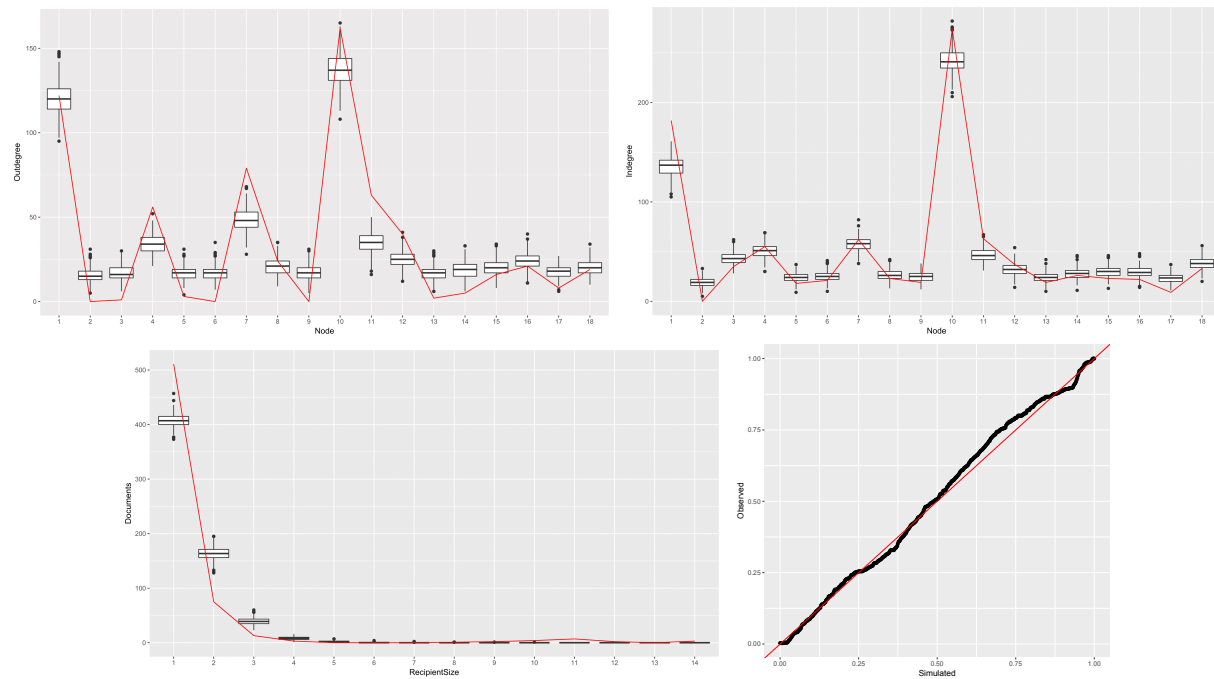 **end for**

4.5   Montgomery County Data

Figure 3: PPC results.

4.6   International Sanctions

5   Conclusion

References

ben Aaron, J., Denny, M., Desmarais, B. & Wallach, H. (2017), 'Transparency by conformity: A field experiment evaluating openness in local governments', *Public Administration Review* **77**(1), 68–77.

Blei, D. M., Ng, A. Y. & Jordan, M. I. (2003), 'Latent dirichlet allocation', *J. Mach. Learn. Res.* **3**, 993–1022.

Burgess, A., Jackson, T. & Edwards, J. (2004), Email overload: Tolerance levels of employees within the workplace, *in* 'Innovations Through Information Technology: 2004 Information Resources Management Association International Conference, New Orleans, Louisiana, USA, May 23-26, 2004', Vol. 1, IGI Global, p. 205.

Butts, C. T. (2008), 'A relational event framework for social action', *Sociological Methodology* **38**(1), 155–200.

Chatterjee, S., Diaconis, P. et al. (2013), 'Estimating and understanding exponential random graph models', *The Annals of Statistics* **41**(5), 2428–2461.

Desmarais, B. A. & Cranmer, S. J. (2017), Statistical inference in political networks research, *in* J. N. Victor, A. H. Montgomery & M. Lubell, eds, 'The Oxford Handbook of Political Networks', Oxford University Press.

Fellows, I. & Handcock, M. (2017), Removing phase transitions from gibbs measures, *in* 'Artificial Intelligence and Statistics', pp. 289–297.

Hunter, D. R., Handcock, M. S., Butts, C. T., Goodreau, S. M. & Morris, M. (2008), 'ergm: A package to fit, simulate and diagnose exponential-family models for networks', *Journal of statistical software* **24**(3), nihpa54860.

Kanungo, S. & Jain, V. (2008), 'Modeling email use: a case of email system transition', *System Dynamics Review* **24**(3), 299–319.

Krafft, P., Moore, J., Desmarais, B. & Wallach, H. M. (2012), Topic-partitioned multinetwork embeddings, *in* F. Pereira, C. Burges, L. Bottou & K. Weinberger, eds, 'Advances in Neural Information Processing Systems 25', Curran Associates, Inc., pp. 2807–2815.

Lim, K. W., Chen, C. & Buntine, W. (2013), Twitter-network topic model: A full bayesian treatment for social network and text modeling, *in* 'NIPS2013 Topic Model workshop', pp. 1–5.

McCallum, A., Corrada-Emmanuel, A. & Wang, X. (2005), The author-recipient-topic model for topic and role discovery in social networks, with application to enron and academic email, *in* 'Workshop on Link Analysis, Counterterrorism and Security', p. 33.

Morgan, T. C., Bapat, N. & Kobayashi, Y. (2014), 'Threat and imposition of economic sanctions 1945–2005: Updating the ties dataset', *Conflict Management and Peace Science* **31**(5), 541–558.

Perry, P. O. & Wolfe, P. J. (2013), 'Point process modelling for directed interaction networks', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **75**(5), 821–849.

Pew, R. C. (2016), 'Social media fact sheet', *Accessed on 03/07/17* .

Robins, G., Pattison, P., Kalish, Y. & Lusher, D. (2007), 'An introduction to exponential random graph (p*) models for social networks', *Social networks* **29**(2), 173–191.

Rubin, D. B. et al. (1984), 'Bayesianly justifiable and relevant frequency calculations for the applied statistician', *The Annals of Statistics* **12**(4), 1151–1172.

Snijders, T. A. (1996), 'Stochastic actor-oriented models for network change', *Journal of mathematical sociology* **21**(1-2), 149–172.

Szóstek, A. M. (2011), '?dealing with my emails?: Latent user needs in email management.', *Computers in Human Behavior* **27**(2), 723–729.

Tanner, M. A. & Wong, W. H. (1987), 'The calculation of posterior distributions by data augmentation', *Journal of the American statistical Association* **82**(398), 528–540.

Appendices

A Normalizing constant of Gibbs measure

The non-empty Gibbs measure Fellows & Handcock (2017) defines the probability of author a selecting the binary recipient vector $\boldsymbol{u}_{ad}$ as

$$P(\boldsymbol{u}_{ad}|\boldsymbol{\lambda}_{ad}) = \frac{\exp\left\{\log\big(\mathrm{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\big) + \sum_{r\neq a}\lambda_{adr}u_{adr}\right\}}{Z(\boldsymbol{\lambda}_{ad})}.$$

To use this distribution efficiently, we derive a closed-form expression for $Z(\boldsymbol{\lambda}_{id})$ that does not require brute-force summation over the support of $\boldsymbol{u}_{ad}$ (i.e. $\forall \boldsymbol{u}_{ad} \in [0,1]^A$). We recognize that if $\boldsymbol{u}_{ad}$ were drawn via independent Bernoulli distributions in which $P(u_{adr} = 1|\delta, \boldsymbol{\lambda}_{ad})$ was given by $\mathrm{logit}(\lambda_{adr})$, then

$$P(\boldsymbol{u}_{ad}|\boldsymbol{\lambda}_{ad}) \propto \exp\left\{\sum_{r\neq a}\lambda_{adr}u_{adr}\right\}.$$

This is straightforward to verify by looking at

$$P(u_{adr} = 1|\boldsymbol{u}_{ad[-r]}, \boldsymbol{\lambda}_{ad}) = \frac{\exp\left(\lambda_{adr}\right)}{\exp\left(\lambda_{adr}\right) + 1}.$$

We denote the logistic-Bernoulli normalizing constant as $Z^l(\boldsymbol{\lambda}_{ad})$, which is defined as

$$Z^l(\delta, \boldsymbol{\lambda}_{ad}) = \sum_{\boldsymbol{u}_{ad}\in[0,1]^A} \exp\left\{\sum_{r\neq a}\lambda_{adr}u_{adr}\right\}.$$

Now, since

$$\exp\left\{\log\Big(\mathrm{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\Big) + \sum_{r\neq a}\lambda_{adr}u_{adr}\right\}$$

$$= \exp\left\{\sum_{r\neq a}\lambda_{adr}u_{adr}\right\},$$

except when $\|\boldsymbol{u}_{ad}\|_1 = 0$, we note that

$$Z(\boldsymbol{\lambda}_{ad}) = Z^l(\boldsymbol{\lambda}_{ad}) - \exp\left\{\sum_{\forall u_{adr}=0}\lambda_{adr}u_{adr}\right\}$$

$$= Z^l(\lambda_a^{(d)}) - 1.$$

17

We can therefore derive a closed form expression for $Z(\delta, \boldsymbol{\lambda}_{ad})$ via a closed form expression for $Z^l(\delta, \boldsymbol{\lambda}_{ad})$. This can be done by looking at the probability of the zero vector under the logistic-Bernoulli model:

$$\frac{\exp\left\{\sum\limits_{\forall u_{adr}=0} \lambda_{adr} u_{adr}\right\}}{Z^l(\boldsymbol{\lambda}_{ad})} = \prod_{r\neq a}\left(1 - \frac{\exp\left(\lambda_{adr}\right)}{\exp\left(\lambda_{adr}\right)+1}\right).$$

Then, we have

$$\frac{1}{Z^l(\boldsymbol{\lambda}_{ad})} = \prod_{r\neq a}\frac{1}{\exp(\lambda_{adr})+1}.$$

Finally, the closed form expression for the normalizing constant under the non-empty Gibbs measure is

$$Z(\boldsymbol{\lambda}_{ad}) = \prod_{r\neq a}\left(\exp(\lambda_{adr})+1\right) - 1.$$