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

March 21, 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

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 $t_d < t_{d+1}$.

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. 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 $\boldsymbol{u}_{ad} = (u_{ad1}, \ldots, u_{adA})$

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

where δ is a real number controlling the average number of recipients and $\boldsymbol{\lambda}_{id} = \{\lambda_{adr}\}_{r=1}^A$. We place a Normal prior $\delta \sim N(\mu_\delta, \sigma_\delta^2)$. In particular, we define $\text{Gibbs}(\delta, \boldsymbol{\lambda}_{ad})$ as

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

where $Z(\delta, \boldsymbol{\lambda}_{ad}) = \prod_{r \neq a}(\exp\{\delta + \lambda_{adr}\} + 1) - 1$ is the normalizing constant and $\|\cdot\|_1$ is the l_1–norm. 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:

$$E(\tau_{ad}) = \mu_{ad},$$
$$V(\tau_{ad}) = V(\mu_{ad}), \tag{5}$$

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 $\phi_\tau(\cdot; \mu, \sigma_\tau^2)$ and $\Phi_\tau(\cdot; \mu, \sigma_\tau^2)$ 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):

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

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.

---

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

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

        form $\boldsymbol{y}_{ad}$

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

        draw $\tau_{ad} \sim \phi_\tau(\mu_{ad}, \sigma_\tau^2)$

    end

    set $a_d = \text{argmin}_a(\tau_{ad})$

    set $\boldsymbol{r}_d = \boldsymbol{u}_{a_d d}$

    set $t_d = t_{d-1} + \min_a \tau_{ad}$

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 $(\boldsymbol{\mu}_b, \Sigma_b, \boldsymbol{\mu}_\eta, \Sigma_\eta, \mu_\delta, \sigma_\delta^2)$. 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

$$
\begin{aligned}
P(u_{adr} = 1 | \boldsymbol{u}_{ad \backslash r}, \boldsymbol{b}, \delta, \boldsymbol{x}) &\propto \exp\{\delta + \lambda_{adr}\}; \\
P(u_{adr} = 0 | \boldsymbol{u}_{ad \backslash r}, \boldsymbol{b}, \delta, \boldsymbol{x}) &\propto \text{I}(\|\boldsymbol{u}_{ad \backslash r}\|_1 > 0),
\end{aligned}
\tag{7}
$$

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 $\delta, \boldsymbol{b}$, 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 δ and $\boldsymbol{b}$ is

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

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 is

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

Algorithm 2 MCMC Algorithm

---

set initial values of $\boldsymbol{b}, \delta, \boldsymbol{\eta}, \sigma_\tau^2$
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
        for *n=1 to inner1* **do**
            update $\boldsymbol{b}$ and δ using Metropolis-Hastings — Equation (8)
        end
        for *n=1 to inner2* **do**
            update $\boldsymbol{\eta}$ and σ_τ^2 using Metropolis-Hastings — Equation (9)
        end
        for *n=1 to inner3* **do**
            update σ_τ^2 using Metropolis-Hastings — Equation (9)
        end
    end
end
Summarize the results with:
last chain of $\boldsymbol{b}$ and δ, 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 dynamic network statistics on three time intervals prior to and including t_{d-1}. We compute eight network statistics within each time interval Perry & Wolfe (2013), where the intervals are $(t_{d-1} - 384h, t_{d-1} - 96h], (t_{d-1} - 96h, t_{d-1} - 24h]$ and $(t_{d-1} - 24h, t_{d-1}]$. We define the intervals to have equal length in the log-scale, and use $i = 1$ to denote the earliest interval—i.e., $(t_{d-1} - 384h, t_{d-1} - 96h]$—and i = 3 to denote the latest.

1. $\mathrm{outdegree}(a, i) = \sum\limits_{d':t_{d'} \in i} I(a_{d'} = a);$

2. $\mathrm{indegree}(r, i) = \sum\limits_{d':t_{d'} \in i} I(u_{d'r} = 1);$

3. $\mathrm{send}(a, r, i) = \sum\limits_{d':t_{d'} \in i} I(a_{d'} = a)I(u_{d'r} = 1);$

4. $\text{receive}(a, r, i) = \text{send}(r, a, i);$

5. $2\text{-send}(a, r, i) = \displaystyle\sum_{\substack{i', i'' \geq i: \\ i'=i \text{ or } i''=i}} \sum_{h \neq a, r} \text{send}(a, h, i')\text{send}(h, r, i'');$

6. $2\text{-receive}(a, r, i) = \displaystyle\sum_{\substack{i', i'' \geq i: \\ i'=i \text{ or } i''=i}} \sum_{h \neq a, r} \text{send}(h, a, i')\text{send}(r, h, i'');$

6. $\text{sibling}(a, r, i) = \displaystyle\sum_{\substack{i', i'' \geq i: \\ i'=i \text{ or } i''=i}} \sum_{h \neq a, r} \text{send}(h, a, i')\text{send}(h, r, i'');$

6. $\text{cosibling}(a, r, i) = \displaystyle\sum_{\substack{i', i'' \geq i: \\ i'=i \text{ or } i''=i}} \sum_{h \neq a, r} \text{send}(a, h, i')\text{send}(r, h, i'');$

where $I(\cdot)$ is an indicator function. Note that in order to obtain two-path statistics (i.e., 2-send, 2-receive, sibling, and cosibling) within a single time interval i, we compute the number of two-paths from a to r by summing over the pairs of intervals (i', i'') where the earlier email in the path was sent during interval i.

For time-related covariates $\boldsymbol{y}_{ad}$,

4.1.3 Experiment

4.1.4 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. 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 network: 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 3 Generate new data for PPC

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

for $r = 1$ **to** R **do**
  for $d = 1$ **to** D **do**
    Compute $\boldsymbol{x}_d$ and $\boldsymbol{y}_d$ given $\{(a_d, \boldsymbol{r}_d, t_d)\}_{[1:(d-1)]}$
    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

4.2 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 Experiment

4.2.4 Posterior Predictive Checks

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.

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

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.

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.

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

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

To use this distribution efficiently, we derive a closed-form expression for $Z(\delta, \boldsymbol{\lambda}_{id})$ 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 $\mathrm{logit}(\delta + \lambda_{adr})$, then

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

This is straightforward to verify by looking at

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

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

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

Now, since

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

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

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

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

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

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:

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

Then, we have

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

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

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