# General Linear Threshold Models with Application to Influence Maximization

Alexander Kagan, Elizaveta Levina, Ji Zhu

Department of Statistics University of Michigan {amkagan, elevina, jizhu}@umich.edu

#### Abstract

A number of models have been developed for information spread through networks, often for solving the Influence Maximization (IM) problem. IM is the task of choosing a fixed number of nodes to "seed" with information in order to maximize the spread of this information through the network, with applications in areas such as marketing and public health. Most methods for this problem rely heavily on the assumption of the known strength of connections between network members (edge weights), which is often unrealistic. In this paper, we develop a likelihoodbased approach to estimate edge weights from the fully and partially observed information diffusion paths. We also introduce a broad class of information diffusion models, the general linear threshold (GLT) model, which generalizes the well-known linear threshold (LT) model by allowing arbitrary distributions of node activation thresholds. We then show our weight estimator is consistent under the GLT and some mild assumptions. For the special case of the standard LT model, we also present a much faster expectation-maximization approach for weight estimation. Finally, we prove that for the GLT models, the IM problem can be solved by a natural greedy algorithm with standard optimality guarantees if all node threshold distributions have concave cumulative distribution functions. Extensive experiments on synthetic and real-world networks demonstrate that the flexibility in the choice of threshold distribution combined with the estimation of edge weights significantly improves the quality of IM solutions, spread prediction, and the estimates of the node activation probabilities.

**Keywords:** Social Networks, Influence Maximization, Information Diffusion Models, Linear Threshold Model

# 1 Introduction

The emergence of large-scale online social networks has led to a new surge of interest in information diffusion models. These networks supply incredibly rich data, which can include connections between users, user covariates, e.g., demographics, and the paths of information propagation between users, e.g., retweets or reposts. Information diffusion paths, also known as *propagation traces*, are especially valuable in modeling information spread since they provide direct data on the influence users have on their network neighbors. "Information" in this context can be interpreted broadly and refer to anything that can spread from node to node, be it a news item or a virus (and sometimes these are connected – Dinh and Parulian [2020] showed that the spread of Covid-19 through people's social networks can be well predicted by their interactions around Covid-related posts on Twitter). For example, Liu and Wu [2018] used propagation traces for fake news detection while Saito et al. [2008] proposed to estimate information diffusion probabilities from the same data. Goval et al. [2011] proposed a model assigning credits to users based on how well they propagate information and subsequently using these weights to solve the influence maximization (IM) problem, that is, identifying the best "seed nodes" for spreading information. While the IM problem and spread prediction are the main applications of the present paper, the method we propose for estimating edge weights from propagation traces is general and can be applied to any other downstream problem in network analysis after the weights are estimated, such as testing for network differences [Tantardini et al., 2019] or community detection in weighted graphs [Lancichinetti and Fortunato, 2009].

Since the IM problem was formulated by Richardson and Domingos [2002] and formalized by Kempe et al. [2003], many efficient approaches for identifying the best seed sets have been proposed.

All of them require a probability model for how information ("influence") spreads over the network, known as the *diffusion model*, usually with edge- or vertex-specific parameters governing information spread. Perhaps the most popular is the independent cascade (IC) model [Goldenberg et al., 2001], which assumes that each edge is associated with a transmission probability and all transmission events are independent. Another popular choice is the linear threshold (LT) model [Granovetter, 1978), which assumes that each edge has a deterministic weight and each node uses a uniformly distributed random threshold to decide whether to "accept" incoming information; this will be stated formally in Section 2. There is substantial literature on solving the IM problem given a specific diffusion model (see Banerjee et al. [2020] for a survey), and some on more efficient but less general approaches, for example, mixed integer programming for the IC model [Farnad et al., 2020]. However, in most applications, the underlying diffusion parameters are unknown and can significantly change the IM problem solution under misspecification. Work by Goyal et al. [2011] highlighted this issue, showing that the IM solutions improve significantly when the edge weights are estimated, for example using the EM approach of Saito et al. [2008] for the IC model. One contribution we make in this paper is to develop an EM approach analogous to Saito et al. [2008] for edge weight estimation under the LT model (Section 4.6). However, the LT and IC model classes are not especially rich, and the natural next step would be to estimate the parameters of more sophisticated and flexible network diffusion models. While there are well-known flexible generalizations of these two models, such as the triggering and general threshold (GT) models studied in Kempe et al. [2003], their additional flexibility is achieved at the cost of exponentially many additional parameters (see Proposition 3.1), making parameter estimation impossible with realistic amounts of data. To address this, we propose the general linear threshold (GLT) model, which has only O(|E| + |V|) parameters but is much more flexible than the LT model; importantly, it allows for heterogeneity in how readily users accept new information. Our generalization is a special case of the GT model but not of the triggering model. We propose a likelihood-based approach for estimating its parameters and show the estimator is consistent under mild regularity conditions. In Section A.6 of the Appendix, we extend this technique by showing the IC model parameters are identifiable and can be consistently estimated under similar conditions. While there have been several papers [He et al., 2016, Narasimhan et al., 2015] establishing Probably Approximately Correct (PAC) learnability guarantees for nodes' activation probabilities under the IC, LT, and several similar models, surprisingly, there has been very little work establishing theoretical guarantees for the diffusion model parameters. The only paper we are aware of that addresses these questions is the work by Rodriguez et al. [2014] where authors derive identifiability conditions and establish consistency for the parameters of several continuous-time diffusion models. In such a way, to the best of our knowledge, this work is the first one to prove consistency for the estimates of discrete diffusion model parameters based on the propagation traces. Finally, we show that under some additional conditions on the GLT model, the IM problem under the GLT assumption can be solved by the natural and widely used greedy strategy described in Section 2, with standard optimality guarantees.

The rest of this manuscript is organized as follows: in Section 2, we briefly introduce the background on the IM problem and the relevant diffusion models and fix notation. In Section 3 we propose the Generalized Linear Threshold (GLT) model, describe its relationship to other diffusion models, and derive the conditions under which the IM problem under the GLT model can be optimally solved by the greedy strategy. In Section 4, we establish identifiability conditions, derive a likelihood approach to weight estimation under the GLT model, and establish consistency. We also present an EM-based improvement of our estimation algorithm for the standard LT model. Finally, Section 5 presents experiments on synthetic and a real-world network showing that using the proposed weight estimation procedure outperforms standard heuristics and estimating the threshold distributions under the GLT model instead of using the standard LT model can substantially improve the IM solutions.

# 2 Background

In this section, we formally define the influence maximization problem and the most commonly used diffusion models. We start by setting up notation.

Let G = (V, E) be a graph where V is the set of nodes and  $E \subseteq V \times V$  is the set of edges. Unless otherwise stated, we assume throughout this manuscript that G is a directed graph with no loops and repeated edges. Denote the sets of parent and children nodes of a node v as, respectively,

$$P(v) = \{u : (u, v) \in E\}$$
 and  $C(v) = \{u : (v, u) \in E\}.$ 

Similarly, for any set of nodes  $S \subset V$ , let

$$P(S) := \bigcup_{v \in S} P(v) \setminus S$$
 and  $C(S) := \bigcup_{v \in S} C(v) \setminus S$ .

In words, P(S) and C(S) consist of nodes outside of S which have at least one child or parent in S, respectively.

A diffusion model associated with G controls the spread of information over the graph. In general, the information diffusion process starts with a given non-empty set  $D_0 \subset V$  of initially activated (influenced) nodes, also known as the seed set. Then, at each time step t = 1, 2, ..., a new subset of nodes  $D_t$ , which were not previously active, become activated following the diffusion model. We only consider progressive propagation, which means that once a node is activated, it remains activated (i.e., it never forgets the information it received). This assumption implies that all sets  $D_t$  are disjoint. Let  $A_t := \bigsqcup_{0 \le \tau \le t} D_{\tau}$  be the set of all nodes activated by time t. We also assume that if a node was not active at time t, it cannot be activated at time t + 1 unless one of its parent nodes was newly activated at time t, meaning that if a given subset of active parents was once not enough for activation, it will never be enough in the future. In particular, this means that a node cannot be influenced from outside of the network G or self-activated. The process stops when no new node is activated; that time point is denoted by  $T = \arg\min_{t>0} \{t: D_{t+1} = \emptyset\}$ . The entire diffusion process can be described by the set sequence  $\mathcal{D} := (D_0, \ldots, D_T)$ , which is called a *propagation trace*, or an information diffusion path. For convenience, we let  $D_{-1} := \emptyset$  and  $A_{-1} := \emptyset$  (no nodes are active before the information is seeded). Finally, we write  $A(\mathcal{D})$  to denote the set of all nodes activated by the end of the propagation described by  $\mathcal{D}$ .

The assumptions on the diffusion process imply not every sequence of node subsets can be a feasible propagation trace. The feasibility condition can be formulated as follows:

**Definition 1** (Feasible trace). We say that a set sequence  $\mathcal{D} = (D_0, \ldots, D_T)$  with  $D_t \subseteq V$  is a feasible propagation trace if

- 1.  $D_0$  is not an empty set;
- 2. All sets  $D_t$ ,  $t = 0, \ldots, T$  are disjoint;
- 3. For every t = 1, ..., T, each newly activated node  $v \in D_t$  has at least one parent in  $D_{t-1}$ , i.e.,  $v \in C(D_{t-1}) \setminus A_{t-1}$ .

The set of all feasible traces on G will be denoted by  $\mathcal{F}(G)$ . Now we are ready to state a formal definition of a diffusion model.

**Definition 2** (Diffusion model). A diffusion model on G = (V, E) is a mapping  $M_{G,\theta}$  which takes any feasible trace on G as input and outputs a probability distribution on the feasible sets of newly active nodes:

$$\mathcal{D} = (D_0, \dots, D_{t-1}) \in \mathcal{F}(G) \quad \stackrel{M_{G,\theta}}{\longmapsto} \quad \mathbb{P}^t_{\theta}(D_t = S \mid \mathcal{D}), \ S \in 2^{C(D_{t-1}) \setminus A_{t-1}}.$$

The mapping depends on parameters  $\boldsymbol{\theta}$ , which can depend on G.

When the context is clear, we will omit the subscript G and write  $M_{\theta}$ . This definition does not specify the distribution that generates the seed set  $D_0$ ; instead, it conditions all output distributions on it. If we additionally define a distribution  $\mathbb{P}^0$  over the subsets of V which does not depend on  $\theta$ , satisfies  $\mathbb{P}^0(\emptyset) = 0$ , and assume that the seed set is drawn from  $\mathbb{P}^0$  and then passed on to the diffusion model, it will be convenient to refer to the pair  $(M_{G,\theta}, \mathbb{P}^0)$  as a *seeded diffusion model* with the seed distribution  $\mathbb{P}^0$ . When G and  $\mathbb{P}^0$  are clear from the context we will write  $M_{\theta}^{\theta}$  instead.

A seeded diffusion model naturally implies a distribution on all the feasible traces:

**Definition 3** (Trace distribution under the seeded diffusion model). Consider a simple directed graph G = (V, E) and a seeded diffusion model  $M^0_{\theta} = (M_{G,\theta}, \mathbb{P}^0)$  on it. For each feasible trace  $\mathcal{D} = (D_0, \ldots, D_T) \in \mathcal{F}(G)$  consider the transition probabilities  $\mathbb{P}^t_{\theta} = M^0_{\theta}(D_0, \ldots, D_{t-1}), t = 1, \ldots, T+1$ . Then the following probability mass function defines the distribution of feasible traces on G induced by  $M^0_{\theta}$ :

$$\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}) := \mathbb{P}^0(D_0) \prod_{t=1}^T \mathbb{P}_{\boldsymbol{\theta}}^t(D_t | D_0, \dots, D_{t-1}) \mathbb{P}_{\boldsymbol{\theta}}^{T+1}(\boldsymbol{\emptyset} | \mathcal{D}), \qquad \mathcal{D} \in \mathcal{F}(G).$$

On the other hand, the trace distribution also uniquely defines the diffusion model. Indeed, for any feasible trace  $(D_0, \ldots, D_{T-1})$  and  $S \in 2^{C(D_{T-1}) \setminus A_{T-1}}$ , we can write:

$$M^{0}_{\boldsymbol{\theta}}(D_{0},\ldots,D_{T-1})(S) = \frac{\sum_{\mathcal{D}'\in\mathcal{F}(G)}\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}')\mathbf{1}(D'_{T}=S;D'_{t}=D_{t},\ \forall t=0,\ldots,T-1)}{\sum_{\mathcal{D}'\in\mathcal{F}(G)}\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}')\mathbf{1}(D'_{t}=D_{t},\ \forall t=0,\ldots,T-1)},$$
(1)

where  $\mathbf{1}(\cdot)$  denotes the indicator of an event.

Next, we give definitions of the two most popular diffusion models, the Linear Threshold (LT) and the Independent Cascade (IC) models. Conceptually, under the LT model, all of a node's activated parents influence it additively, whereas the IC model gives each active parent node one individual chance to influence its children.

**Definition 4** (Linear Threshold (LT) Model). Assume that each edge  $(u, v) \in E$  has a weight  $b_{u,v} \geq 0$ , and the in-degrees satisfy, for all  $v \in V$ ,

$$\sum_{e \in P(v)} b_{u,v} \le 1. \tag{2}$$

Each node  $v \in V$  has an activation threshold  $U_v$ , with thresholds sampled i.i.d. from the Unif[0, 1] distribution at the outset. At every time step  $t \ge 1$ , each non-active node v becomes activated if the sum of the edge weights from all its previously activated parents exceeds its threshold  $U_v$ , that is,  $v \in D_t$  if

u

$$v \notin A_{t-1}$$
 and  $\sum_{u \in P(v) \cap A_{t-1}} b_{u,v} \ge U_v.$ 

**Definition 5** (Independent Cascade (IC) Model). Assume that each edge  $(u, v) \in E$  is associated with a propagation probability  $p_{u,v} \in [0, 1]$ . At every time step  $t \ge 1$ , each newly active node  $u \in D_{t-1}$ independently tries to activate all its not yet active children  $v \in C(u) \setminus A_{t-1}$  with probability  $p_{u,v}$ . That is,  $v \in D_t$  if  $v \notin A_{t-1}$  and at least one  $Z_{u,v} \sim \text{Bernoulli}(p_{u,v})$ , for  $u \in P(v) \cap D_{t-1}$ , takes on the value 1.

The vector of parameters  $\boldsymbol{\theta}$  in the general definition 2 corresponds to  $\{b_{u,v} : (u,v) \in E\}$  for the LT model, and to  $\{p_{u,v} : (u,v) \in E\}$  for the IC model.

Given a graph G and an arbitrary diffusion model M on it, the Influence Maximization (IM) problem poses a natural question: which nodes are the most effective spreaders of information in the network, that is, where should a given number of seeds be placed to start the diffusion process so that the resulting expected number of activated nodes is maximized? This question was first introduced by Richardson and Domingos [2002] and further formalized by Kempe et al. [2003]. Before stating the definition of the IM problem, we introduce the notion of the influence function:

**Definition 6** (Influence function). Given a simple directed graph G = (V, E) and a diffusion model M on it, let the influence function  $\sigma_{G,M} : 2^V \to \mathbb{R}_{\geq 0}$  denote a set function that maps each subset of nodes  $S \subset V$  to the expected number of nodes influenced if propagation is seeded at  $D_0 = S$ :

$$S \xrightarrow{\sigma_{G,M}} \mathbb{E}_{\mathcal{D}|D_0=S}|A(\mathcal{D})|, \qquad S \subset V.$$

**Definition 7** (The Influence Maximization (IM) problem). Given a simple directed graph G = (V, E), a diffusion model M on it, and a budget  $k \in \mathbb{N}$ , the IM problem is to find a subset  $S^* \subset V$  which would maximize the influence function across all k-element subsets of V:

$$S^* = \operatorname*{argmax}_{S \subset V: |S| \le k} \sigma_{G,M}(S).$$

For the rest of the paper, we omit the subscripts G and M of  $\sigma_{G,M}$  whenever they are clear from the context.

In Kempe et al. [2003], it was shown that the IM problem is NP-hard under both LT and IC models and hence intractable unless P = NP. However, when the influence function has certain properties, the optimal solution can be well approximated by a greedy strategy (see Algorithm 1). These properties are monotonicity and submodularity. Monotonicity means that adding more nodes to a seed set cannot decrease the value of the influence function, and submodularity means that adding a node to a given seed set is at least as efficient as adding it to a bigger seed set containing this given set, both very reasonable and mild assumptions.

**Definition 8** (Monotonicity). An influence function  $\sigma(\cdot)$  is monotone if  $\sigma(S') \leq \sigma(S)$  for any  $S' \subset S \subseteq V$ .

**Definition 9** (Submodularity). An influence function  $\sigma(\cdot)$  is submodular if  $\sigma(\{v\} \cup S') - \sigma(S') \ge \sigma(\{v\} \cup S) - \sigma(S)$  for any  $S' \subset S \subseteq V$  and  $v \in V \setminus S$ .

Though Kempe et al. [2003] were the first to study the greedy algorithm behavior in the context of the IM problem, the analysis of its worst-case performance under monotonicity and submodularity assumptions dates back to the works of Nemhauser, Wolsey, and Fisher who proved the following result:

**Theorem 2.1** (Cornuéjols et al. [1977] and Nemhauser et al. [1978]). Consider a universal set  $\Omega$ and a monotone and submodular function  $\sigma : 2^{\Omega} \to \mathbb{R}_{\geq 0}$ . Let  $\hat{S} \subset \Omega$  of size k be the set obtained by selecting elements from  $\Omega$  one at a time, when at each step one chooses an element that provides the largest marginal increase in the value of  $\sigma$ . Let  $S^*$  be the true maximizer of  $\sigma$  over all k-element subsets of  $\Omega$ . Then

$$\sigma(\hat{S}) \ge \left(1 - \frac{1}{e}\right)\sigma(S^*) \tag{3}$$

or, in other words,  $\hat{S}$  provides a  $1-\frac{1}{e}$  approximation to the optimal  $S^*$ .

Kempe et al. [2003] showed that under the LT and IC models, the influence function has the monotonicity and submodularity properties and thus the IM problem under these models can be solved by Algorithm 1 with the optimality guarantee in (3).

Algorithm 1 The greedy algorithm for the IM problem

```
Input: graph G = (V, E), diffusion model M, and seed budget k

S \leftarrow \emptyset

while |S| < k do

v \leftarrow \arg \max_{v \in V \setminus S} (\sigma_{G,M}(S \cup \{v\}) - \sigma_{G,M}(S))

S \leftarrow S \cup \{v\}

return S
```

# 3 The general linear threshold model

In this section, we introduce a generalization of the LT model which we call the General Linear Threshold (GLT) model. It generalizes the standard LT model by allowing each node v to have an individual and possibly non-uniform threshold distribution  $F_v$ . We show that for GLT models with concave  $F_v$ 's, Algorithm 1 solves the IM problem with the optimality guarantees (3). We further show that despite having only O(|E| + |V|) parameters, the GLT model is surprisingly rich. In particular, in Proposition 3.2, we show that it is not a special case of the Triggering model, a well-known overparametrized generalization of the LT model. We begin by giving a formal definition of the GLT model.

**Definition 10** (General Linear Threshold (GLT) Model). Assume each node  $v \in V$  is assigned a non-negative threshold  $U_v$  from a distribution with cumulative distribution function (cdf)  $F_v$ , s.t.  $F_v^{-1}(0) = 0$  and  $h_v := F_v^{-1}(1) \leq +\infty$ . Also, assume that each edge  $(u, v) \in E$  is assigned a weight  $b_{u,v} \geq 0$ , such that for all  $v \in V$ ,

$$\sum_{u \in P(v)} b_{u,v} \le h_v$$

The propagation process is then the same as for the LT model.

An even more general model was introduced by Kempe et al. [2003], allowing the activation probability to be an arbitrary, not necessarily linear, function of the active parent set.

**Definition 11** (General Threshold (GT) model). Assume each node v in the graph is assigned a threshold function  $f_v$  that maps subsets of v's parents to [0, 1], such that  $f_v(\emptyset) = 0$ . As in the LT model, each node v chooses an activation threshold  $U_v \sim \text{Unif}[0, 1]$ . A node v is activated at step  $t \ge 1$  if  $f_v(A_{t-1} \cap P(v)) \ge U_v$ .

The GLT model is an instance of the GT model with  $f_v(S) := F_v(\sum_{u \in S} b_{u,v})$ . The GT model does not automatically guarantee the desirable submodularity property, which implies the IM problem cannot be solved by Algorithm 1 with the optimality guarantee (3). However, Mossel and Roch [2010] showed that a relatively mild extra assumption enforces submodularity on the GT model.

**Theorem 3.1** (Mossel and Roch [2010]). The GT model on a graph G = (V, E) is monotone and submodular if the threshold function  $f_v$  is monotone and submodular for each node  $v \in V$ .

We will denote the subclass of Monotone Submodular GT models by MSGT. Theorem 3.1 implies a sufficient submodularity condition for the GLT model, stated next.

**Theorem 3.2.** The GLT model on a graph G = (V, E) is monotone and submodular if the threshold  $cdf F_v$  is concave for each node  $v \in V$ .

The proof can be found in Section A.3 of the Appendix. This general theorem makes it easy to establish conditions on a particular threshold distribution that guarantee submodularity. For example, if the thresholds are sampled from a beta distribution (to which we will return later), it is easy to check submodularity, as shown next in Corollary 3.1.

**Corollary 3.1.** A GLT model with  $F_v \sim \text{Beta}(\alpha_v, \beta_v)$  is submodular if  $\alpha_v \leq 1, \alpha_v + \beta_v \geq 2$  for all  $v \in V$ .

*Proof.* The density  $q(x; \alpha, \beta)$  of the Beta $(\alpha, \beta)$  distribution is non-increasing on its support whenever for each  $x \in (0, 1)$ 

$$q'(x|\alpha,\beta) \propto x^{\alpha-2}(1-x)^{\beta-2}[(\alpha-1)(1-x)-(\beta-1)x] \le 0,$$

which is equivalent to  $(\alpha + \beta - 2)x \ge \alpha - 1$ .

The following example shows submodularity may fail for a network instance under the GLT model when the threshold cdfs are not concave. Consider the 3-node graph shown in Figure 1. The influence function satisfies  $\sigma(\{1\}) = 1 + F_3(x), \sigma(\{2\}) = 1 + F_3(y)$ , and  $\sigma(\{1,2\}) = 2 + F_3(x+y)$ . Then if  $F_3 \sim \text{Beta}(2,1)$  and both edge weights x and y are positive, the submodularity condition implies a contradiction:

$$\sigma(\{1,2\}) - \sigma(\{1\}) \le \sigma(\{2\}) - \sigma(\emptyset) \quad \Leftrightarrow \quad F_3(x+y) \le F_3(x) + F_3(y) \Leftrightarrow \quad (x+y)^2 \le x^2 + y^2.$$

On the other hand, if at least one of x and y is zero, submodularity will hold.

Next, we compare the GLT model to the LT model and its other widely used generalization from the MSGT family known as the Triggering model. For formal proof showing that the LT model is a special case of the Triggering model, see Theorem 2.5 of Kempe et al. [2003].

**Definition 12** (Triggering Model). At the outset, each node v independently chooses a random triggering set  $\Gamma_v$  according to some distribution over subsets of its parents. An inactive node v becomes active at time  $t \geq 1$  if its triggering set  $\Gamma_v$  contains a node in  $D_{t-1}$ .

The Triggering model allows for a different value of  $\mathbb{P}(\Gamma_v = S)$  for each non-empty subset S of v's parents, making the number of unknown parameters  $\sum_{v \in V} (2^{|P(v)|} - 1)$ , which is not feasible to fit in most cases. On the other hand, the LT model has only |E| parameters, so computationally it is an attractive option. A major disadvantage of the LT model is its assumption that all nodes behave identically when receiving equal amounts of influence from their neighbors, and does not allow for heterogeneity in readiness to be influenced. In reality, some people are conservative and need a lot to be convinced, while some may be trusting and easily influenced; this can be estimated from data on propagation traces through that network. The GLT model allows us to account for differences in users' attitudes to new information while not significantly increasing the number of unknown parameters.

A natural approach to making the number of parameters manageable is to choose threshold distributions from a parametric family. For example, if  $F_v \sim \text{Beta}(\alpha_v, \beta_v)$ , the model has only |E| + 2|V| parameters. If we further assume that the network can be partitioned into communities and nodes within one community follow the same distribution (see Figure 2), we can further reduce this number. However, even if each node has its own individual set of r parameters, the following proposition demonstrates that the GLT almost always has fewer parameters than the Triggering and the MSGT models, which both have  $\sum_{v \in V} (2^{|P(v)|} - 1)$  parameters.



Figure 1: Example: a three-node graph with positive weights  $b_{13} = x > 0, b_{23} = y > 0$  and the threshold cdf  $F_3(t) = t^2 \sim \text{Beta}(2, 1)$  has a non-submodular influence function.



Figure 2: Example: three communities with different levels of conservativeness, all modeled with the Beta distribution.

**Proposition 3.1.** Consider a directed graph G = (V, E) with average node in-degree d. Then for any  $r \leq 2^d - d - 1$ ,

$$|E| + r|V| \le \sum_{v \in V} \left( 2^{|P(v)|} - 1 \right).$$

*Proof.* Noticing that |E| = d|V| and applying Jensen's inequality to the convex function  $f(x) = 2^x$ , we obtain

$$\frac{|E|+r|V|}{\sum_{v \in V} \left(2^{|P(v)|} - 1\right)} \le \frac{(d+r)|V|}{(2^d-1)|V|} = \frac{d+r}{2^d-1} \le 1.$$

The next proposition shows that despite a much smaller number of parameters, there are submodular instances of the GLT model that can not be represented as an instance of the Triggering model. The proof can be found in Section 3.2 of the Appendix.

**Proposition 3.2.** There exists a GLT model  $M_{G,\theta}$  with a submodular influence function for which there is no instance of the Triggering model  $\tilde{M}_{G,\tilde{\theta}}$  such that the induced trace distributions of  $M_{G,\theta}$ and  $\tilde{M}_{G,\tilde{\theta}}$  coincide on each feasible trace  $\mathcal{D} \in \mathcal{F}(G)$ .

Figure 3 summarizes the relationships between the various diffusion models. As in Proposition 3.2, we represent one diffusion model family as contained in another model family if for each instance  $M_{G,\theta}$  of the smaller family there is an instance  $\tilde{M}_{G,\tilde{\theta}}$  of the bigger family such that their induced trace distributions coincide.



Figure 3: Relationship between various generalizations of the LT model.

### 4 Estimation and theoretical properties under the GLT model

### 4.1 Identifiability for the GLT model

In this section, we study identifiability of the weights  $\boldsymbol{\theta} := \{b_{u,v} : (u,v) \in E\}$  with respect to the trace distribution  $\mathbb{P}_{\boldsymbol{\theta}}$  induced by the GLT model, and then present conditions for consistency of the MLE of the edge weights. To simplify further narration, we would like to fix the edge ordering in E to work with  $\boldsymbol{\theta}$  as a vector, not as a set. To achieve this, we assume that the nodes are enumerated from 1 to |V| and the edges are sorted in the lexicographical order with the child node having priority. For example, a triangle graph with each edge going in both directions corresponds to the ordered set E = [(2, 1), (3, 1), (1, 2), (3, 2), (1, 3), (2, 3)].

By Definition 3 and independence of node activation thresholds, the probability of a feasible trace  $\mathcal{D} = (D_0, \ldots, D_T)$  is given by

$$\mathbb{P}_{\theta}(\mathcal{D}) = \mathbb{P}^{0}(D_{0}) \prod_{v \in C(A_{T})} \left\{ 1 - F_{v} \left[ B\left(v, A_{T}\right) \right] \right\} \times \prod_{t=0}^{T-1} \prod_{v \in D_{t+1}} \left\{ F_{v} \left[ B\left(v, A_{t}\right) \right] - F_{v} \left[ B\left(v, A_{t-1}\right) \right] \right\}, \quad (4)$$

where B(v, S) denotes the influence node v receives from a set S, i.e.,

$$B(v,S) = \sum_{S \cap P(v)} b_{u,v}.$$
(5)

The first term in (4) does not depend on  $\theta$  by definition of the seeded diffusion model. The second term represents nodes that were not activated but have at least one active parent in the trace. The third term captures information about activated nodes, i.e., nodes in  $A_T \setminus D_0$ .

The parameter space includes all weights satisfying the GLT model assumptions,

$$\Theta = \left\{ \boldsymbol{\theta} \in \mathbb{R}^{|E|} \text{ s.t. for all } v \in V, \sum_{w \in P(v)} b_{w,v} \le h_v, \text{ and } b_{u,v} \ge 0 \text{ for all } (u,v) \in E \right\}.$$
(6)

This space has a nice block structure arising from grouping edges by their terminal node. Denote the set of (child) vertices with at least one parent as

$$V_c = \{ v \in V : P(v) \neq \emptyset \}$$
(7)

and for every  $v \in V_c$ , define its *individual parameter subspace* as

$$\Theta_v = \{ \boldsymbol{\theta}_v \in \mathbb{R}^{|P(v)|} : \ \boldsymbol{\theta}_v \ge 0, \ \|\boldsymbol{\theta}_v\|_1 \le h_v \},$$
(8)

where  $\|\cdot\|_1$  denotes the  $\ell_1$ -norm of a vector and inequality between a number and vector denotes element-wise comparison. Then, the parameter space  $\Theta$  can be written as a Cartesian product of individual parameter spaces:

$$\Theta = \underset{v \in V_c}{\otimes} \Theta_v. \tag{9}$$

This decomposition will be crucial for deriving the MLE of weights since it allows decomposing the log-likelihood optimization problem into at most  $|V_c|$  small sub-problems.

To ensure identifiability of  $\{\mathbb{P}_{\theta}, \theta \in \Theta\}$ , we need to impose some constraints on the threshold cdfs  $F_v$ , which up to this point were not restricted.

Assumption 1. For each  $v \in V$ , the threshold  $cdf F_v$  is continuous, strictly monotone (thus invertible), and  $h_v = F_v^{-1}(1) < \infty$ .

Together with  $F_v^{-1}(0) = 0$  stated in the GLT model definition,  $h_v < \infty$  implies the threshold distributions have bounded support. This will be crucial in proving the MLE consistency since it makes the parameter space  $\Theta$  compact. Additionally, we will assume that all the weights are separated from zero and each node in-degree is separated from the upper bound of the corresponding threshold distribution:

**Assumption 2.** There exists a universal  $\varepsilon_0$ ,  $0 < \varepsilon_0 < \min_{v \in V} \frac{h_v}{|P(v)|+1}$  such that the true weights satisfy  $\varepsilon_0 \leq b_{u,v}$  for all  $(u, v) \in E$  and  $\sum_{u \in P(v)} b_{u,v} \leq h_v - \varepsilon_0$  for all  $v \in V$ .

Intuitively, Assumption 1 means that any node has a positive chance to activate its child even if none of the child's other parents are activated. Assumption 2 means that even if all parents of a node are activated, there is a positive probability the node will not be activated. We require  $\varepsilon_0 < h_v/(|P(v)|+1)$  because if all  $b_{u,v} \ge \varepsilon_0$  then it should also hold

$$|P(v)|\varepsilon_0 \le \sum_{u \in P(v)} b_{u,v} \le h_v - \varepsilon_0$$
, therefore  $\varepsilon_0(|P(v)| + 1) \le h_v$ .

We rule out equality because it would force  $b_{u,v} = \varepsilon_0$  for all  $u \in P(v)$ . Assumption 2 implies that similarly to (9), we can define the truncated parameter space as  $\tilde{\Theta} = \bigotimes_{v \in V_c} \tilde{\Theta}_v$  where

$$\hat{\Theta}_{v} = \{ \boldsymbol{\theta}_{v} : \| \boldsymbol{\theta}_{v} \|_{1} \le h_{v} - \varepsilon_{0}, \ \boldsymbol{\theta}_{v} \ge \varepsilon_{0} \}.$$

$$(10)$$

The following property of  $\tilde{\Theta}$  is crucial for identifiability.

**Lemma 4.1.** Under Assumption 1, any feasible trace  $\mathcal{D} \in \mathcal{F}(G)$  with  $\mathbb{P}^0(D_0) > 0$  has a positive probability w.r.t.  $\mathbb{P}_{\theta}$  for each  $\theta \in \tilde{\Theta}$ .

*Proof.* Fix an arbitrary  $\boldsymbol{\theta} \in \tilde{\Theta}$ . For any  $v \in V$  with parent subsets  $S' \subset S \subseteq P(v)$ , the strict monotonicity of  $F_v$  implies

$$0 < F_v\left(\sum_{u \in S'} b_{u,v}\right) < F_v\left(\sum_{u \in S} b_{u,v}\right) \le F_v\left(\sum_{u \in P(v)} b_{u,v}\right) \le F_v(h_v - \varepsilon_0) < 1.$$

Therefore, each term of the product in (4) is strictly positive. As the number of terms in the likelihood is bounded by |V| (each node can appear in a trace at most once), we have  $\mathbb{P}_{\theta}(\mathcal{D}) > 0$ .

The next proposition shows that even if all weights are separated from zero, identifiability is not guaranteed if there are source nodes that have zero probability of appearing in the trace.

**Proposition 4.1.** If  $\{\mathbb{P}_{\theta}, \theta \in \tilde{\Theta}\}$  is identifiable, then for each  $u \in V$  with  $C(u) \neq \emptyset$ , the following must hold for any  $\theta \in \tilde{\Theta}$ :

$$\mathbb{P}_{\theta}(u \in A(\mathcal{D})) > 0$$

If additionally,  $P(u) = \emptyset$  it must hold that

$$\mathbb{P}^0(u \in D_0) > 0.$$

Proof. Assume there is  $\boldsymbol{\theta} \in \tilde{\Theta}$  and  $u \in V$  with a child v such that  $\mathbb{P}_{\boldsymbol{\theta}}(u \in A(\mathcal{D})) = 0$ . Since u cannot be activated, the edge (u, v) will never participate in a trace, therefore if we change  $b_{u,v}$  while keeping other weights in  $\boldsymbol{\theta}$  fixed,  $\mathbb{P}_{\boldsymbol{\theta}}$  will remain the same. Contradiction. Now, if u also does not have a parent, it cannot be activated by other nodes, so it may appear in  $A(\mathcal{D})$  only if it lies in a seed set.

Next, we show that the necessary condition in Proposition 4.1 has an equivalent more intuitive formulation.

**Proposition 4.2.** Define a set  $R := R(G, \mathbb{P}^0)$  of "reachable" nodes in G that consists of  $u \in V$  s.t. either  $\mathbb{P}^0(u \in D_0) > 0$  or there is a directed path to u from at least one  $w \in V$  with  $\mathbb{P}^0(w \in D_0) > 0$ . Then the following conditions are equivalent:

- (a) For all  $\boldsymbol{\theta} \in \tilde{\Theta}$ , it holds  $\mathbb{P}_{\boldsymbol{\theta}}(u \in A(\mathcal{D})) > 0$ ,
- (b) There is  $\boldsymbol{\theta} \in \tilde{\Theta}$  such that  $\mathbb{P}_{\boldsymbol{\theta}}(u \in A(\mathcal{D})) > 0$ .
- (c)  $u \in R$ .

Proof. Statement (a) trivially implies (b). To show (b) implies (c), note that (b) implies that there is a feasible trace  $\mathcal{D} = (D_0, \ldots, D_t, \ldots, D_T)$  with  $\mathbb{P}^0(D_0) > 0$  and  $u \in D_t$ . We prove by induction over  $\tau \geq 1$  that there is a path to u from a node  $w_{t-\tau} \in D_{t-\tau}$ . First, for  $\tau = 1$ , since  $\mathcal{D}$  is feasible, ushould have a parent  $w_{t-1}$  in  $D_{t-1}$ . Now suppose the induction hypothesis holds for  $\tau$ . If there is a path to u from  $w_{t-\tau} \in D_{t-\tau}$ , there is also a path from  $D_{t-\tau-1}$  since  $w_{t-\tau}$  should have a parent in  $D_{t-\tau-1}$  by feasibility. This results in a path  $(w_0, \ldots, w_{t-1}, u)$  connecting  $D_0$  and u which implies that  $u \in R$ .

To prove (c) implies (a), consider an arbitrary  $u \in R$ . If  $\mathbb{P}^0(u \in D_0) > 0$ , then (a) holds. Assume now there is a sequence of nodes  $(w_0, w_1, \ldots, w_T = u)$  such that  $(w_{t-1}, w_t) \in E$  for all  $t = 1, \ldots, T$ and  $w_0 \in D_0$  with  $\mathbb{P}^0(D_0) > 0$ . Then,  $\mathcal{D} = (D_0, \{w_1\}, \ldots, \{w_T\})$  is a feasible trace that has a positive probability for any  $\boldsymbol{\theta} \in \tilde{\Theta}$  according to Lemma 4.1.

Combining Propositions 4.2 and 4.1, we deduce that weights are identifiable only if all edges have sources either appearing in or reachable from a seed set having positive probability, i.e., if  $V = R(G, \mathbb{P}^0)$ . Unfortunately, the following example demonstrates this condition is insufficient for identifiability.

**Example.** Consider the graph in Figure 1 and assume that the seed set is fixed at nodes 1 and 2, i.e.,  $\mathbb{P}^{0}(\{1,2\}) = 1$ . Then the distribution of possible traces is a function of  $b_1 + b_2$ ,

$$\mathbb{P}(\mathcal{D}) = \begin{cases} F_3(b_1 + b_2), & \text{node } 3 \in D_1 \\ 1 - F_3(b_1 + b_2) & \text{node } 3 \notin D_1 \end{cases},$$

and therefore  $b_1$  and  $b_2$  are not identifiable, only their sum is. It is easy to verify that if the support of  $\mathbb{P}^0$  includes at least two distinct subsets of  $\{1, 2\}$ , the weights are identifiable.

This example provides some intuition on why identifiability of  $\{\mathbb{P}_{\theta}, \theta \in \tilde{\Theta}\}$  imposes some requirements on the seed set distribution  $\mathbb{P}^0$ . This intuition is formalized in the following theorem. Its proof is given in Section A.4 of the Appendix.

**Theorem 4.1.** Under Assumption 1,  $\{\mathbb{P}_{\theta}, \theta \in \tilde{\Theta}\}$  is identifiable if and only if for each child node  $v \in V_c$  with  $P(v) = \{u_1, \ldots, u_m\}$ , there exist  $S_1, \ldots, S_m \subseteq P(v)$  such that

- 1. For each j = 1, ..., m, there is a feasible trace  $(D_0^{(j)}, ..., D_{t_j}^{(j)}) \in \mathcal{F}(G)$  with  $\mathbb{P}^0(D_0^{(j)}) > 0$ ,  $v \notin A_{t_i}^{(j)}$ , and  $D_{t_i}^{(j)} \cap P(v) = S_j$ .
- 2. The matrix  $X_v = [\mathbf{1}(u_i \in S_j)]_{i,j=1}^m$  is invertible.

The identifiability condition in Theorem 4.1 implies the necessary reachability condition from Proposition 4.1: if a source node u of an edge (u, v) is unreachable, the matrix  $X_v$  will not be invertible, since it will have a row of zeros  $[\mathbf{1}(u \in S_j)]_{j=1}^m$  for any choice of  $S_j$ .

### 4.2 Weight estimation under GLT models

Next, we propose a likelihood-based approach to estimating the weights in the GLT model from a collection  $\mathcal{T}$  of N observed (and therefore feasible) propagation traces,

$$\mathcal{T} = \{ \mathcal{D}_n := (D_1^{(n)}, \dots, D_{T_n}^{(n)}) \mid n = 1, \dots, N \},$$
(11)

where  $T_n$  is the number of time steps in trace  $\mathcal{D}_n$ . For now, we assume that all threshold distributions  $F_v$  are known and postpone the discussion of estimating the threshold distribution to Section 4.5.

We assume that the trace collection  $\mathcal{T} = \{\mathcal{D}_n \mid n = 1, \dots, N\}$  is i.i.d., by which we mean

(a) Seed sets  $\{D_0^{(n)} | n = 1, ..., N\}$  are generated independently from the seed distribution  $\mathbb{P}^0$ ;

(b) Node thresholds are generated independently for each trace and for each node, with

$$\mathcal{U}_n := \left( U_1^{(n)}, \dots, U_{|V|}^{(n)} \right) \stackrel{\text{iid}}{\sim} (F_1, \dots, F_{|V|}), \quad n = 1, \dots, N.$$

For an i.i.d. trace collection, the parameters can be estimated by solving the problem

$$\max_{\boldsymbol{\theta}\in\Theta}\sum_{n=1}^{N}L(\mathcal{D}_{n}|\boldsymbol{\theta}),\tag{12}$$

where  $L(\mathcal{D}_n|\boldsymbol{\theta})$ , the log-likelihood of the trace  $\mathcal{D}_n$ , by (4), takes the form

$$L(\mathcal{D}_{n}|\boldsymbol{\theta}) = \sum_{v \in C(A_{T_{n}}^{(n)})} \log \left\{ 1 - F_{v} \left[ B\left(v, A_{T_{n}}^{(n)}\right) \right] \right\}$$

$$+ \sum_{t=1}^{T_{n}} \sum_{v \in D_{t}^{(n)}} \log \left\{ F_{v} \left[ B\left(v, A_{t-1}^{(n)}\right) \right] - F_{v} \left[ B\left(v, A_{t-2}^{(n)}\right) \right] \right\}.$$
(13)

Here, we omitted the  $\log \mathbb{P}^0(D_0)$  term as it does not depend on  $\boldsymbol{\theta}$ . Note that in (12), we optimize over untruncated space  $\Theta$  instead of  $\tilde{\Theta}$ , as in practice the true "slack" variable  $\varepsilon_0$  from Assumption 2 is unknown.

Examining (13), we see that the likelihood depends only on weights  $b_{u,v}$  for  $v \in V_c(\mathcal{T})$ , the subset of child nodes defined by

$$V_{c}(\mathcal{T}) := \bigcup_{n=1}^{N} \left\{ \left[ A_{T_{n}}^{(n)} \setminus D_{0}^{(n)} \right] \cup C(A_{T_{n}}^{(n)}) \right\}.$$
 (14)

Here, each set in the union consists of the nodes that either were activated after the seed set within a given trace or failed to be activated while having an active parent.

This is still a large number of parameters to optimize over, but fortunately, the problem has a block structure we can use to speed up computations. By changing the order of summation, we can rewrite the objective of (12) as

$$\sum_{n=1}^{N} L(\mathcal{D}_n | \boldsymbol{\theta}) = \sum_{v \in V_c(\mathcal{T})} L_v(\boldsymbol{\theta}_v)$$

where

$$L_{v}(\boldsymbol{\theta}_{v}) = \sum_{n: v \in C(A_{T_{n}}^{(n)})} \log \left\{ 1 - F_{v} \left[ B\left(v, A_{T_{n}}^{(n)}\right) \right] \right\} + \sum_{n: v \in A_{T_{n}}^{(n)} \setminus D_{0}^{(n)}} \log \left\{ F_{v} \left[ B\left(v, A_{t(v,n)-1}^{(n)}\right) \right] - F_{v} \left[ B\left(v, A_{t(v,n)-2}^{(n)}\right) \right] \right\},$$
(15)

and  $t(v,n) := \arg\min_{t\geq 0} \{t : v \in A_t^{(n)}\}$  is the activation time of node v in trace n. Therefore, solving (12) is equivalent to maximizing  $L_v(\boldsymbol{\theta}_v)$  over  $\boldsymbol{\theta}_v \in \Theta_v$  defined in (8), for each  $v \in V_c(\mathcal{T})$ , an optimization problem with only |P(v)| variables and |P(v)| + 1 affine constraints:

$$\max_{\boldsymbol{\theta}_{v}\in\boldsymbol{\Theta}_{v}}L_{v}(\boldsymbol{\theta}_{v}), \quad v\in V_{c}(\mathcal{T}).$$
(16)

The next natural question is whether this optimization problem is convex. The feasible set  $\Theta_v$  is a convex simplex, and the arguments of  $F_v$  in (15) depend linearly on  $\theta_v$ . Thus  $L_v$  is a concave function of  $\theta_v$  if  $\log[F_v(x) - F_v(y)]$  is a concave function on x > y. For example, if  $U_v$  is uniformly distributed on [0, 1], as in the standard LT model, then  $\log[F_v(x) - F_v(y)] = \log(x - y)$  is concave. This turns out to be true for all distributions with log-concave densities, and in particular when  $F_v$  is the Beta distribution with parameters  $\alpha_v \ge 1$  and  $\beta_v \ge 1$ .

**Proposition 4.3.** The function  $L_v(\theta_v)$  in (15) is concave in  $\theta_v$  if  $F_v$  has a log-concave density.

The proof is given in Section A.1 of the Appendix.

#### 4.3 Consistency

If the identifiability conditions hold, MLE consistency follows almost immediately from standard results.

**Theorem 4.2.** Under Assumptions 1, 2, identifiability condition in Theorem 4.1, and the assumption of *i.i.d.* traces, the MLE  $\hat{\theta}_N$  of  $\theta^*$  obtained by solving the optimization problem (12) is consistent, *i.e.*,  $\hat{\theta}_N \to \theta^*$  in probability as  $N \to \infty$ .

*Proof.* By Theorem 9.9 in Keener [2010], MLE  $\hat{\boldsymbol{\theta}}_N$  is consistent if the parameter space  $\tilde{\Theta}$  is compact,  $\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D})$  is a continuous function of  $\boldsymbol{\theta}$  for each  $\mathcal{D}$ ,  $\{\mathbb{P}_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \tilde{\Theta}\}$  is identifiable, and

$$\mathbb{E}_{\mathcal{D}\sim\mathbb{P}_{\theta^*}}\sup_{\theta\in\tilde{\Theta}}|\log\mathbb{P}_{\theta}(\mathcal{D}) - \log\mathbb{P}_{\theta^*}(\mathcal{D})| < \infty.$$
(17)

The parameter space  $\tilde{\Theta}$  is compact as a closed subset of compact  $\Theta$ . Continuity of  $\mathbb{P}_{\theta}$  holds since each term in (13) is a composition of continuous functions. The last condition holds since the quantity under the supremum (17) is a continuous function of  $\theta$ , therefore it achieves its maximum at some  $\tilde{\theta} \in \tilde{\Theta}$ . Finally,  $\mathbb{P}_{\tilde{\theta}}$  and  $\mathbb{P}_{\theta^*}$  have the same support by Lemma 4.1, therefore the quantity under the expectation is finite for any  $\mathcal{D}$  in the support of  $\mathbb{P}_{\theta^*}$ .

As the weight identifiability condition in Theorem 4.1 is very restrictive, Theorem 4.2 becomes hardly applicable in practice. Therefore, we conclude this section by extending the consistency result to the case when some edges have source nodes unreachable from a seed set in the support of  $\mathbb{P}^0$ . One possible way to deal with this issue is to restrict the identifiability question to a subgraph where all source nodes are reachable. Instead of the full graph G, we can define a restriction G' = (V', E'), where V' is the set of nodes reachable from or belonging to a seed set in the support of  $\mathbb{P}^0$  and  $E' := \{(u, v) \in E : u, v \in V'\}$ , and restrict the parameter space  $\tilde{\Theta}$  to E' as follows:

$$\tilde{\Theta}' = \left\{ \boldsymbol{\theta}' \in \mathbb{R}^{|E'|} \mid v \in V' : \sum_{w \in P(v) \cap V'} b_{w,v} \le h_v - \varepsilon_0, \ (u,v) \in E' : b_{u,v} \ge \varepsilon_0 \right\}.$$

Then Theorems 4.1 and 4.2 are both applicable to the subgraph G'.

We note that the identifiability conditions stated in Theorem 4.1 are also necessary and sufficient for identifying the IC model parameters, assuming all edge probabilities  $p_{u,v}$  are bounded away from zero and one. Given identifiability, the proof of consistency is identical to Theorem 4.2. Since the IC model lies outside of the main scope of this paper, we present and prove the corresponding theorems in Section A.6 of the Appendix.

#### 4.4 Extension to partially-observed traces.

In many real-life applications, we do not observe a full propagation trace but only have access to a node's active parents it had before activating itself. For example, suppose a person (node) gets infected given their other family members (parent nodes) are currently infected. In that case, we can infer that the virus transmission from one of the family members occurred while we may not know the full virus propagation path leading to their own infection. We will encode each such observation for node v as a pair  $(A_v, y)$  where  $A_v \subset P(v)$  is a set of v's active (infected) parents and  $y \in \{0, 1\}$  is an indicator of the event that  $A_v$  together managed to activate (infect) v. We will refer to such pair as a *pseudo-trace*.

Suppose that for each child node  $v \in V_c$ , we observe a possibly empty collection of pseudo-traces

$$\mathcal{T}_{v} = \{ (A_{v}^{(n)}, y_{v}^{(n)}), \ n = 1, \dots, N_{v} \}$$
(18)

that we would like to use to estimate the GLT weights  $\theta_v$ . A natural question is if we can assume some pseudo-trace generating distribution for  $\mathcal{T}_v$  so that the corresponding likelihood optimization problem and consistency theory directly follow from what we previously established for the fully observed traces. The most straightforward way to do that is to treat a pseudo-trace  $(A_v, y)$  as a trace seeded at  $A_v$  and propagating in a graph  $G_v$  with nodes  $\{v\} \cup P(v)$  and edges  $\{(u, v) : u \in P(v)\}$ . Given that  $A_v \subset P(v)$ , the feasible traces on  $G_v$  can only be of two types: those who stopped immediately at the seed set  $A_v$  and those who activated v at time one and then stopped. The first case corresponds to a pseudo-trace  $(A_v, 0)$  and the second one to  $(A_v, 1)$ . Assuming that the sets  $A_v^{(n)}$ ,  $n = 1, ..., N_v$  are independently generated from a parameter-free seed distribution  $\mathbb{P}_v^0$  supported on the subsets of P(v), the pseudo-trace likelihood will have the form

$$\mathbb{P}_{\theta_{v}}(A_{v}, y) = \mathbb{P}_{v}^{0}(A_{v}) \left\{ 1 - F_{v} \left[ B(v, A_{v}) \right] \right\}^{1-y} F_{v} \left[ B(v, A_{v}) \right]^{y}.$$
(19)

Aggregation of these probabilities across all pseudo-traces in  $\mathcal{T}_{v}$  results in the log-likelihood:

$$L_{v}^{pt}(\boldsymbol{\theta}_{v}) = \sum_{n: \ y^{(n)}=0} \log\left\{1 - F_{v}\left[B\left(v, A_{v}^{(n)}\right)\right]\right\} + \sum_{n: \ y^{(n)}=1} \log F_{v}\left[B\left(v, A_{v}^{(n)}\right)\right],$$
(20)

where we omitted the parameter-free  $\log \mathbb{P}_{v}^{0}(A_{v})$  terms. The assumption that the seed distribution  $\mathbb{P}_{v}^{0}$  does not depend on any diffusion model parameters may seem strong, as a seed node in  $A_{v}$  could have been activated by its own network parents, making the probability of  $A_{v}$  indirectly dependent on the seed's incoming edge weights. However, by carefully comparing (20) with its counterpart for fully observed traces in (15), we observe that the only difference is that, in the latter case, we always subtract  $F_{v}\left[B\left(v, A_{t(v,n)-2}^{(n)}\right)\right]$  under the logarithm for traces where v was activated. This term represents the probability that the active parent set preceding the one that eventually activated v was not enough for v's activation. Thus, the only information lost in a pseudo-trace, compared to a fully observed trace, is which parent subset of an influenced node v was not sufficient to activate it.

With pseudo-trace distribution formulated as a trace distribution on a sub-graph  $G_v$ , identifiability and consistency theory are extended straightforwardly:

**Theorem 4.3.** Under Assumption 1,  $\{\mathbb{P}_{\theta_v}, \theta_v \in \tilde{\Theta}_v\}$  is identifiable for v with  $P(v) = \{u_1, \ldots, u_m\}$  if and only if there exist  $S_1, \ldots, S_m \subseteq P(v)$  such that

- 1. For each j = 1, ..., m, it holds  $\mathbb{P}_{v}^{0}(S_{j}) > 0$ .
- 2. The matrix  $X_v = [\mathbf{1}(u_i \in S_j)]_{i,j=1}^m$  is invertible.

**Theorem 4.4.** Under Assumptions 1, 2, identifiability condition in Theorem 4.3, and the assumption of *i.i.d.* pseudo-traces, the MLE  $\hat{\theta}_v$  of  $\theta_v^*$  obtained by solving the optimization problem (16) is consistent, *i.e.*,  $\hat{\theta}_v \to \theta_v^*$  in probability as  $N_v \to \infty$ .

Proofs of these theorems can be found in Section A.5 of Appendix.

### 4.5 Estimation of threshold parameters

So far we have treated threshold distributions known, while they are unlikely to be observed in reality. While we cannot estimate these parameters without assuming something about their distribution, we can easily obtain an estimate if we assume each  $F_v, v \in V$  comes from the same parametric family with parameters  $\varphi_v \in \Phi_v \subset \mathbb{R}^{r_v}$ . For example, assuming  $F_v \sim \text{Beta}(1, \beta_v)$ , we can define  $\varphi_v = \beta_v$  with  $\Phi_v = [1, +\infty)$  to satisfy conditions in both Corollary 3.1 and Proposition 4.3. Then we can estimate  $(\theta_v, \varphi_v)$  for each  $v \in V_c(\mathcal{T})$  by solving the following optimization problem:

$$\max_{\boldsymbol{\rho}_{v} \in \Phi_{v}, \ \boldsymbol{\theta}_{v} \in \Theta_{v}} L_{v}\left(\mathcal{T} | \boldsymbol{\theta}_{v}, \boldsymbol{\varphi}_{v}\right)$$
(21)

where the individual node likelihood  $L_v$  is the one from (15) with the Beta distribution cdfs plugged in. Allowing  $F_v$  to vary within the feasible set makes the optimization problem non-convex even in the simplest case of a one-parameter Beta family. Thus finding even a local optimum of (21) requires careful tuning of the gradient steps since  $\theta$  and  $\varphi$  might have very different magnitudes. A natural way to deal with that problem is to switch to coordinate gradient descent, alternating between fixing one set of variables ( $\theta_v$  or  $\varphi_v$ ) and optimizing over the other one. Our numerical experiments, however, showed that this type of coordinate gradient descent converges reliably only if the initial values are sufficiently close to the truth (not included in the paper but available in the GitHub repository). Therefore, unless the dimension  $r_v$  of  $\Phi_v$  is very high, we choose  $\Phi_v$  from a discrete grid, optimize (21) over  $\theta$  for each  $\varphi_v \in \Phi_v$  and choose the one resulting in the highest log-likelihood. An example of such optimization when all node thresholds follow the same Beta( $\alpha^*, \beta^*$ ) distribution (with  $\alpha^*, \beta^*$  unknown) can be found in the Section B.1 of Appendix. While in our further experiments, we also assume that all nodes' thresholds follow Beta distribution, the parametric family as well as the parameter grid  $\Phi_v$  in (21) are not required to be the same across  $v \in V_c(\mathcal{T})$ .

Unfortunately, if we add more threshold parameters to the model, the grid search approach quickly becomes computationally infeasible. A possible alternative is the method of Turnbull [1976], who

proposed a non-parametric estimator of the cdf from a set of intervals within which the corresponding (censored) random variable was independently observed. In the GLT model case, for each node v, we observe independent realizations  $U_v^{(n)}$ ,  $n = 1, \ldots, N$  of its activation threshold, each within a weight-dependent interval. This interval is  $\left(B(v, A_{t(v,n)-2}^{(n)}), B(v, A_{t(v,n)-1}^{(n)})\right)$  for traces where v was activated, and  $\left(B(v, A_{T_n}^{(n)}) \infty\right)$  for traces where it was not. This suggests a natural iterative procedure that alternates solving (16) separately for each  $\boldsymbol{\theta}_v$  and estimating  $F_v$  nonparametrically. We leave investigation of this method for future work.

### 4.6 EM-algorithm for the LT model

Though each problem in (16) has only |P(v)| optimized variables, it may still be a large number when the network is sufficiently dense. In this case, efficient optimization may require significant computational resources. Thus, we present a much faster Expectation-Maximization (EM) algorithm for weight estimation under the LT model that uses explicit parameter updates avoiding numerical gradient computation. This method is analogous to the approach of Saito et al. [2008], who derived an EM algorithm under the independent cascade (IC) model. A simulation study comparing the run time and estimation accuracy of the EM algorithm with problem (16) can be found in Section B.2 of Appendix.

By Theorem 2.5 in Kempe et al. [2003], the LT model can be equivalently reformulated in terms of the Triggering model as follows.

**Definition 13** (Equivalent definition of the LT model). Assume we are given a graph G = (V, E) with non-negative edge weights  $b_{u,v}$  satisfying the LT model constraint  $\sum_{u \in P(v)} b_{u,v} \leq 1$  for all  $v \in V$ . Suppose that at the start each node v independently chooses its *triggering node set*  $\Gamma_v$  which contains at most one parent node  $u \in P(v)$ . Each parent u is chosen to be in  $\Gamma_v$  with probability  $b_{u,v}$ , and  $\Gamma_v = \emptyset$  with probability  $1 - \sum_{u \in P(v)} b_{u,v}$ . Given the initial seed set  $D_0$ , at each time step  $t \geq 1$ , a node v is activated  $(v \in D_t)$  if  $\Gamma_v \subset D_{t-1} \cap P(v)$ .

This formulation has the advantage of a natural latent structure which can be leveraged to derive an EM algorithm. For each trace, define latent variables

$$\mathcal{Z}_n = \{ Z_{u,v}^{(n)} = \mathbf{1}(\Gamma_v^{(n)} = \{u\}), \text{ for } (u,v) \in E \},\$$

where  $Z_{u,v}^{(n)}$  is the indicator of the event that v chooses u as its activator in trace n. Then the full joint log-likelihood for  $\mathcal{Z}_n$  and the trace  $\mathcal{D}_n$  can be written as

$$L(\mathcal{Z}_n, \mathcal{D}_n | \boldsymbol{\theta}) = \sum_{t=0}^{T_n - 1} \sum_{v \in D_{t+1}^{(n)}} \sum_{u \in D_t^{(n)} \cap P(v)} Z_{u,v}^{(n)} \log b_{u,v} + \sum_{v \in C(A_{T_n}^{(n)})} \left[ \sum_{u \in P(v) \setminus A_{T_n}^{(n)}} Z_{u,v}^{(n)} \log b_{u,v} + Z_{\emptyset,v}^{(n)} \log \left( 1 - \sum_{u \in P(v)} b_{u,v} \right) \right],$$

where

$$Z_{\emptyset,v}^{(n)} := \mathbf{1} \left[ \Gamma_v^{(n)} = \emptyset \right] = \mathbf{1} \left[ Z_{u,v}^{(n)} = 0 \text{ for all } u \in P(v) \right].$$

To perform the *E*-step, we need to compute the conditional expectation of  $Z_{u,v}^{(n)}$  given the trace  $\mathcal{D}_n$  and the current estimate of the weights  $\hat{\boldsymbol{\theta}} = \{\hat{b}_{u,v} : (u,v) \in E\}$ . First, consider the case when  $u \in D_t^{(n)}$  and  $v \in D_{t+1}^{(n)}$ . Then

$$\mathbb{E}\Big[Z_{u,v}^{(n)}|\mathcal{D}_n, \hat{\boldsymbol{\theta}}\Big] = \mathbb{P}\Big[Z_{u,v}^{(n)} = 1 \ \Big| \bigcup_{w \in P(v) \cap D_t^{(n)}} \{Z_{w,v}^{(n)} = 1\}, \hat{\boldsymbol{\theta}}\Big] = \frac{b_{u,v}}{\sum_{w \in P(v) \cap D_t^{(n)}} \hat{b}_{w,v}} := \hat{I}_{u,v}^{(n)}, \hat{I}_{u,v}^{$$

where the denominator gives the probability that v is activated at time t along the trace n, i.e., the probability that v's one chosen activating node is in the set  $D_t^{(n)}$ . Next, for  $v \in C(A_{T_n}^{(n)})$ , we have

$$\mathbb{E}\left[Z_{\emptyset,v}^{(n)}\Big|\mathcal{D}_{n},\hat{\boldsymbol{\theta}}\right] = \mathbb{P}\left[\bigcap_{w\in P(v)} \{Z_{w,v}^{(n)}=0\} \Big| \bigcap_{w\in A_{T_{n}}^{(n)}\cap P(v)} \{Z_{w,v}^{(n)}=0\}, \hat{\boldsymbol{\theta}}\right] = \frac{1-\sum_{w\in P(v)} b_{w,v}}{1-\sum_{w\in P(v)\cap A_{T_{n}}^{(n)}} \hat{b}_{w,v}} := \hat{J}_{\emptyset,v}^{(n)}.$$

Finally, if v chose an activating node that was not activated along the trace, i.e., if  $v \in C(A_{T_n}^{(n)})$  and  $u \in P(v) \setminus A_{T_n}^{(n)}$ , we have

$$\mathbb{E}\left[Z_{u,v}^{(n)}\Big|\mathcal{D}_{n},\hat{\boldsymbol{\theta}}\right] = \mathbb{P}\left[Z_{u,v}^{(n)} = 1 \Big| \bigcap_{w \in A_{T_{n}}^{(n)} \cap P(v)} \{Z_{w,v}^{(n)} = 0\}, \hat{\boldsymbol{\theta}}\right] = \frac{b_{u,v}}{1 - \sum_{w \in P(v) \cap A_{T_{n}}^{(n)}} \hat{b}_{w,v}} := \hat{J}_{u,v}^{(n)} \cdot \hat{J}_{u,v}^{(n)} + \hat{J}_{u,v}^{(n)} \cdot \hat{J}_{u,v}^{(n)} + \hat{J}_{u,v}^{(n)} \cdot \hat{J}_{u,v}^{(n)} + \hat{J}_{u,v}^{(n)} + \hat{J}_{u,v}^{(n)} \cdot \hat{J}_{u,v}^{(n)} + \hat{J}_{u$$

Summing over all the traces  $n = 1 \dots N$ , we obtain the following target function for the *M*-step of the EM algorithm:

$$Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}) = \sum_{n=1}^{N} \left\{ \sum_{t=0}^{T_n-1} \sum_{v \in D_{t+1}^{(n)}} \sum_{u \in D_t^{(n)} \cap P(v)} \hat{I}_{u,v}^{(n)} \log b_{u,v} + \sum_{v \in C(A_{T_n}^{(n)})} \left[ \hat{J}_{\emptyset,v}^{(n)} \log \left(1 - \sum_{u \in P(v)} b_{u,v}\right) + \sum_{u \in P(v) \setminus A_{T_n}^{(n)}} \hat{J}_{u,v}^{(n)} \right] \right\}.$$
(22)

To perform the *M*-step, we solve the system of linear equations given by the optimality condition  $dQ/db_{u,v} = 0$ . The system can be split into blocks of equations, with each block only involving  $\theta_v, v \in V_c(\mathcal{T})$ . Solving this for each v explicitly leads to the update for the weight

$$b_{u,v} = \frac{\hat{H}_{u,v}}{\hat{H}_{\emptyset,v} + \sum_{w \in P(v)} \hat{H}_{w,v}} , \qquad (23)$$

where

$$\hat{H}_{u,v} = \sum_{n: \ u \in D_{t(u,n)}^{(n)}, v \in D_{t(u,n)+1}^{(n)}} \hat{I}_{u,v}^{(n)} + \sum_{n: \ u \notin A_{T_n}^{(n)}, \ v \in C(A_{T_n}^{(n)})} \hat{J}_{u,v}^{(n)} ,$$
(24)

$$\hat{H}_{\emptyset,v} = \sum_{n: \ v \in C(A_{T_{v}}^{(n)})} \hat{J}_{\emptyset,v}^{(n)} \ .$$
(25)

Note that  $\hat{H}_{\emptyset,v} = 0$  if v was activated in each trace where it had at least one active parent. In this case, (23) implies  $\sum_{u \in P(v)} b_{u,v} = 1$ , as expected (in-degree achieves the maximum possible value).

The resulting EM optimization procedure can be summarized as Algorithm 2. Similarly to problem (16), this algorithm can be parallelized w.r.t. the child nodes  $v \in V_c(\mathcal{T})$  since the update rule for  $\theta_v$  depends only on its previous iteration estimate but not on the whole  $\theta$ .

#### Algorithm 2 The EM algorithm for the LT model

**Require:** graph G, traces  $\mathcal{T}$ , number of update steps M, initial estimate  $\hat{\boldsymbol{\theta}}^{(0)}$ for  $v \in V_c(\mathcal{T})$  do for  $m = 1, \ldots, M$  do Compute  $\hat{J}_{\emptyset,v}^{(n)}$  using current estimate  $\hat{\boldsymbol{\theta}}_v^{(m-1)}$ Compute  $\hat{H}_{\emptyset,v}$  from (25) using  $\hat{J}_{\emptyset,v}^{(n)}$ for  $u \in P(v)$  do Compute  $\hat{J}_{u,v}^{(n)}$  and  $\hat{I}_{u,v}^{(n)}$  using current estimate  $\hat{\boldsymbol{\theta}}_v^{(m-1)}$ Compute  $\hat{J}_{u,v}^{(n)}$  and  $\hat{I}_{u,v}^{(n)}$  using  $\hat{J}_{u,v}^{(n)}$  and  $\hat{I}_{u,v}^{(n)}$ Compute  $\hat{\theta}_{u,v}^{(m)}$  from (24) using  $\hat{J}_{u,v}^{(n)}$  and  $\hat{I}_{u,v}^{(n)}$ Compute  $\hat{b}_{u,v}^{(m)}$  from (23) using  $\hat{H}_{u,v}$  and  $\hat{H}_{\emptyset,v}$ return final estimate  $\hat{\boldsymbol{\theta}}^{(M)}$ 

## 5 Experiments

In this section, we present simulation studies and a real-world data example to illustrate the effectiveness of the proposed method. The code for these analyses is available at https://github.com/AlexanderKagan/gltm\_experiments. The parameter estimation algorithms for the IC (Saito et al. [2008]), LT (Algorithm 2), and GLT (16) models, as well as trace sampling, spread estimation, and greedy algorithm under these models, are implemented as a Python package InfluenceDiffusion,

available at https://github.com/AlexanderKagan/InfluenceDifusion. Whenever we fit the GLT model by solving the optimization problem in equation (16), we use the SciPy implementation of the SLSQP solver [Jones et al., 2001]. For all synthetic graphs in subsequent experiments, seed sets are sampled uniformly from the node sets of sizes between 1 and 5:

$$\mathbb{P}^{0} = \text{Unif}\{D_{0} \subset V : 1 \le |D_{0}| \le 5\}.$$
(26)

#### 5.1 Weight estimation under the LT model

We begin by demonstrating that Algorithm 2 outperforms existing heuristics for weight estimation when the LT model is used as the ground truth for trace generation. While solving problem (16) with all  $F_v$  set as uniform distributions is a possible alternative, we do not include it in the comparison because its weight estimates are identical to those of the EM algorithm, up to numerical precision. Therefore, we only report the results for the EM algorithm method (denoted as **LT-EM** in the plots).

As benchmarks, we use two heuristics from Goyal et al. [2011] and the EM-based method from Saito et al. [2008], which estimates the edge probabilities for the IC model:

- 1. Weighted Cascade (**WC**): The weight of an edge  $(u, v) \in E$  is estimated as the inverse of the in-degree of v, i.e.,  $\hat{b}_{u,v} = 1/|P(v)|$ .
- 2. Propagated Trace Proportion (**PTP**): The weight of edge (u, v) is estimated based on the ratio between the number of traces where u is activated before v and the number of traces where u is activated. Normalization is used to ensure that the in-degree of each node equals 1:

$$\hat{b}_{u,v} \propto \frac{|\{n : u \in D_{t_u}^{(n)}, v \in D_{t_v}^{(n)}, t_u < t_v\}|}{|\{n : u \in A(\mathcal{D}_n)\}|}, \qquad \sum_{u \in P(v)} \hat{b}_{u,v} = 1$$

3. IC probabilities estimated with the EM method from Saito et al. [2008] (**IC-EM**): Since the estimated edge probabilities  $\hat{p}_{u,v}$  in the IC model do not necessarily satisfy the in-degree constraint of the LT model, we normalize them to ensure unit in-degrees:

$$\hat{b}_{u,v} = \frac{\hat{p}_{u,v}}{\sum_{w \in P(v)} \hat{p}_{w,v}}$$

It is important to note that the listed benchmarks can only produce weight estimates where each child node v has a unit in-degree. In contrast, Algorithm 2 is not constrained by this restriction due to the  $\hat{H}_{\emptyset,v}$  term in the update rule, which allows it to fit arbitrary node in-degrees. In our first scenario, we demonstrate that this flexibility significantly improves estimation accuracy when the true node in-degrees differ from one. To create heterogeneous in-degrees, the parent edge weights  $\theta_v^*$  for each child node v are sampled uniformly from the interior of the simplex  $\Theta_v$ :

$$\boldsymbol{\theta}_{v}^{*} \sim \operatorname{Unif}\{\boldsymbol{\theta}_{v} \in \mathbb{R}^{|P(v)|} : \boldsymbol{\theta}_{v} \ge 0, \ \|\boldsymbol{\theta}_{v}\|_{1} \le 1\},$$

$$(27)$$

which allows us to evaluate the performance of Algorithm 2 under conditions of varying node in-degrees.

In our second scenario, we show that even when the true node in-degrees are all one, the LT-EM algorithm still significantly outperforms the heuristics PTP and WC, while performing comparably to IC-EM. This experiment also highlights that although Theorem 4.2 guarantees consistency only when node in-degrees are separated from one (as per Assumption 2), in practice, the estimation error decreases towards zero as the number of traces increases, even when this assumption is violated. To generate the ground-truth LT model weights with unit in-degree, we sample  $\theta_v^*$  uniformly from the surface of the simplex  $\Theta_v$ :

$$\boldsymbol{\theta}_{v}^{*} \sim \operatorname{Unif}\{\boldsymbol{\theta}_{v} \in \mathbb{R}^{|P(v)|} : \boldsymbol{\theta}_{v} \ge 0, \ \|\boldsymbol{\theta}_{v}\|_{1} = 1\},$$

$$(28)$$

which ensures that the generated weights adhere to the unit in-degree constraint for this experimental setting.

In both scenarios, we use a synthetic graph with 100 nodes generated from the directed Erdös-Renyi (ER) model [Erdös and Rényi, 1959], where each directed edge has a probability of p = 0.1. Then given the ground-truth LT model weights, we sample N traces, where N takes values from  $\{250, 500, 1000, 2000, 4000\}$ , and estimate the weights using all candidate methods based on these traces. For each value of N, we calculate the Relative Mean Absolute Error (RMAE) between the true weights  $\theta^*$  and estimated weights  $\hat{\theta}$ :

$$\operatorname{RMAE}(\boldsymbol{\theta}^*, \hat{\boldsymbol{\theta}}) := \frac{\|\boldsymbol{\theta}^* - \hat{\boldsymbol{\theta}}\|_1}{\|\boldsymbol{\theta}^*\|_1}.$$
(29)

This process is repeated 10 times for each value of N. The average RMAEs, along with the corresponding two standard errors, are presented in Figure 4. As shown, the error for the proposed LT-EM approach decreases toward zero as the number of traces increases, and LT-EM consistently outperforms the benchmark methods.



Figure 4: RMAE for the weight estimators as a function of the number of traces. Networks are generated using the directed ER model with p = 0.1. Traces are sampled from the LT model with edge weights generated based on (27) (Left) and (28) (Right). The error bars represent two standard errors and are calculated from 10 repetitions of each experiment.



Figure 5: (Left) RMAE of the LT-EM estimator as a function of the ER graph size for different edge probabilities p. The LT model weights are generated according to (27) and estimated using N = 2000 traces. (Right) RMAE as a function of the number of traces for different maximum node in-degrees, as defined in (30). The error bars represent two standard errors and are calculated from 10 repetitions of each experiment.

Further, we evaluate the performance of LT-EM under varying ER edge probabilities and different magnitudes of LT model weights. Specifically, in the ER edge probability experiment, we fix the number of traces at N = 2000 while varying the number of graph nodes and the edge probability p. In the LT model weight magnitude experiment, we fix the graph size to 100 nodes and p = 0.1, and vary the number of traces and the maximum weighted in-degree of the nodes. The maximum in-degree is controlled by sampling LT model weights using the following procedure:

$$\boldsymbol{\theta}_{v}^{*} \sim \mathrm{Unif}\{\boldsymbol{\theta}_{v} \in \mathbb{R}^{|P(v)|} : \boldsymbol{\theta}_{v} \geq 0, \ \|\boldsymbol{\theta}_{v}\|_{1} \leq d_{max}\},\tag{30}$$

where  $d_{max} \in \{0.2, 0.4, 0.6, 0.8, 1\}$ . The results are presented in Figure 5. We did not include the benchmark methods in this figure, as their behaviors are similar to those in Figure 4. In the left panel of Figure 5, we observe that the estimation error increases as the edge probability p or the network size grows. This is expected, as higher network density or larger network size increases the number of edge weights to estimate, requiring more traces for accurate estimation. The right panel

of Figure 5 shows that lower ground-truth weight magnitudes lead to higher estimation errors. This is because larger weight magnitudes result in higher node activation probabilities, producing longer traces with more edge weights contributing to the likelihood estimation.

#### 5.2 The impact of threshold distribution on IM problem solutions

In this section, we demonstrate that selecting an accurate diffusion model can significantly improve the quality of the seed set obtained by solving the IM problem using Algorithm 1.

As in the previous section, we use the directed ER model with p = 0.1 to generate graphs with 100 nodes. However, this time we aim to explore the behavior of IM solutions across different network instances, so we sample 10 networks  $G_{\ell}, \ell = 1, ..., 10$  instead of using a fixed network. For each network, we generate N = 2000 traces from the ground-truth GLT model with weights sampled according to (27), and the threshold distributions are as follows:

$$F_v \sim \text{Beta}(1, \beta_v), \quad \beta_v \sim \text{Unif}\{1, 2, 3, 4, 5\}.$$
 (31)

To examine how diffusion model misspecification impacts IM solutions, we compare the following methods: the LT model with weights estimated by the two heuristics from the previous section (WC and PTP) and Algorithm 2 (LT-EM), the IC model with weights estimated using the method from Saito et al. [2008] (still denoted as IC-EM, though we do not normalize the estimated probabilities here), and the GLT model with both weights and threshold distributions estimated by solving problem (21), where the threshold distribution for each node v is assumed to be  $\text{Beta}(1, \beta_v)$ , with  $\beta_v$  estimated via a grid search over  $\Phi_v = \{1, 2, ..., 10\}$ .



Figure 6: Comparison of the average spread across different seed set sizes, where the seeds are selected by a greedy algorithm under five candidate diffusion models and the ground truth. Networks with 100 nodes are generated from the directed ER model with p = 0.1. The ground truth is the GLT model with weights sampled according to (27), and the threshold distributions  $F_v \sim \text{Beta}(1, \beta_v)$ , where  $\beta_v \sim \text{Unif}\{1, 2, 3, 4, 5\}$ . Each box plot represents the estimated spread across 10 networks, where the spread is averaged over five fitted models (each trained on a separate set of 2000 traces). All values of the spread function  $\sigma(\cdot)$  are estimated using 1000 Monte Carlo simulations.

For each of the diffusion models described above, we follow these steps:

- 1. For each network  $G_{\ell}$ , we estimate the model weights (and in the case of GLT, the  $\beta_v$  values as well) using the corresponding estimation procedure. The estimated diffusion model is denoted  $\hat{M}_{\ell}$ ,  $\ell = 1, ..., 10$ .
- 2. For each network  $G_{\ell}$ , we run Algorithm 1 with seed set size  $k = \{1, 4, 7, 10, 13\}$  under the estimated diffusion model to obtain a seed set  $\hat{S}_{\ell,k}$ . The influence function  $\sigma_{G_{\ell},\hat{M}_{\ell}}(\cdot)$  in Algorithm 1 is approximated using 1000 Monte Carlo simulations.
- 3. For each network  $G_{\ell}$ , repeat the above steps five times and calculate the average spread for each seed set size k, denoted by  $\hat{\sigma}_{\ell,k}$ .

As a benchmark, we also compute the average spread under the ground truth GLT model, referred to as the **Oracle**.

Figure 6 presents box-plots (across the 10 networks) of  $\hat{\sigma}_{\ell,k}$  for different values of k, ranging from 1, 4, 7, 10, to 13. The results indicate that as the seed set size k increases, the choice of diffusion model has a greater impact on the spread. This is likely because, for smaller seed sets, the greedy algorithm tends to select the most central nodes regardless of the fitted model. However, as more non-central nodes are considered for the seed set, significant differences in spread emerge due to the incorrect estimation of their influence by certain models. As expected, LT-EM and IC-EM show inferior performance compared to the ground truth and the GLT model. Notably, the GLT model achieves performance comparable to that of the Oracle.

#### 5.3 Spread estimation

In the previous experiment, we only evaluated the spreads from seed sets that were optimal according to the greedy Algorithm 1. However, in practice, we might also be interested in assessing the spread of information, such as fake news or a virus, initiated from a given seed set that has not been optimally selected. In this section, we show that, for propagation under the GLT model, selecting an accurate threshold distribution can significantly improve the accuracy of spread estimation.

As in the previous experiment, we consider a 100-node graph generated from the directed ER model with p = 0.1 and the associated GLT model, where weights are sampled according to (27) and the threshold distributions are set to  $F_v^* \sim \text{Beta}(2,2)$  for all nodes v. We then generate a set of 2500 traces from this GLT model and randomly split them into a training set of 2000 traces and a test set of 500 traces. For candidate threshold distributions Beta(2,2), Beta(2,1), Unif[0,1], and Beta(3,1), we estimate the corresponding GLT model using the training set by solving problem (16) with all  $F_v$  set as this distribution. Finally, for each seed set in the test traces, we compute the estimated spread using 1000 Monte Carlo simulations.



Figure 7: Comparison between the actual spreads from 500 test sets and the estimated spreads for several candidate GLT models with a fixed Beta threshold distribution for all the nodes. A network of 100 nodes is generated from the directed ER model with p = 0.1. The trace-generating model is the GLT with  $F_v^* \sim \text{Beta}(2,2)$  for all nodes, and weights generated according to (27). Weights for each candidate GLT model are estimated using 2000 training traces. For each estimated GLT model and each set in the test traces, the spread is estimated using 1000 Monte Carlo simulations.

In Figure 7, we observe that the model estimated with the ground-truth Beta(2, 2) distribution performs best maintaining relatively low errors for both low and high true spread values. For example, when the true spread is low, models with  $\alpha > \beta$  (referred to as "conservative" in Figure 2) tend to underestimate the spread, while their "anti-conservative" counterparts tend to overestimate it. For large true spread values, none of the models performs exceptionally well; however, the less conservative models, such as the LT model (red) and the ground truth Beta(2, 2), show relatively smaller errors. We note that this severe underestimation of spread is not specific to the diffusion models considered in this experiment; rather, it represents a general pattern in spread estimation, as observed in other studies (e.g., Figure 2 in Goyal et al. [2011]). High spread is an extreme event in the tail of the spread distribution and, therefore, cannot be accurately predicted by the mean spread, even when computed under the ground-truth model.

### 5.4 Real-world data example

In this section, we apply the proposed weight estimation procedures to the Flixster dataset, collected from www.flixster.com, a popular social platform for movie ratings. The dataset contains an undirected, unweighted social network of approximately 1 million users and over 8 million records of the times when users rated specific movies, spanning from 2005 to 2009. Following the notations in Goyal et al. [2011], these rating actions can be encoded as an *action log*, which is a collection of triples (u, a, t), where u represents the user ID, a the movie ID, and t the time when the user rated the movie. The authors assumed that if user u rated movie a before user v (a connected child in the social graph), the influence likely propagated from u to v. Of course, this assumption can be questioned, as a user may rate a movie based solely on personal interest or recommendations from someone outside the observed social network. However, the authors' experiments show that incorporating the information in the action log can significantly improve the estimation of influence propagation. In our experiment, we aim to compare the performance of different diffusion models discussed in this manuscript in terms of their accuracy in predicting node activation events.

To preprocess the data, we first removed all users who rated fewer than 20 movies, as it is difficult to estimate the influence of their social network neighbors on them with limited activity. Next, we apply the periphery-cleaning algorithm described in [Kojaku and Masuda, 2018] to extract the core sub-graph of the remaining users. Finally, we isolate the largest connected component of this core graph. This preprocessing results in a network of 8174 nodes and approximately 50,000 undirected edges. By restricting the initial action log to only the users within this network, we reduce the size of the log to roughly 2.1 million tuples. With the truncated action log in hand, the next step is to transform it into trace or pseudo-trace data that can be used to fit the diffusion models, such as through optimization problem (21) or Algorithm 2. As all diffusion models discussed throughout this manuscript assume a directed graph, we add reverse edges, resulting in 100k directed edges in our final network.

Constructing full propagation traces proves challenging, even in simple scenarios. For example, consider a network with three users who follow each other pairwise and watch the same movie at distinct times  $t_1 < t_2 < t_3$ . There are already five possible ways to construct the corresponding trace:  $(\{1,2,3\}), (\{1\},\{2,3\}), (\{1,2\},\{3\}), (\{1\},\{2,3\}), or (\{1\},\{2\},\{3\})$ . This complexity arises because the trace data we considered earlier in this manuscript has a discrete-time structure, while the Flixster dataset involves continuous time. To reduce ambiguity in trace construction, we propose using the pseudo-trace framework described in Section 4.4.

For each node v, we extract pseudo-traces where v was activated (y = 1) by identifying all actions a it performed and noting the set  $A_v^{(a)} \subset P(v)$  of v's parents who performed a before v. For pseudo-traces where v was not activated (y = 0), we consider all actions a that v did not perform but at least one of its parents did, noting the set  $A_v^{(a)}$  of all its active parents at the last recorded time point in the action log. In such a way, we construct the pseudo-trace collection for each node v in the graph. We then randomly split the data into training and test sets in a 4:1 ratio, stratified by the activation status y. Using the same candidate diffusion models as in Section 5.2, we estimate the models based on the training pseudo-traces and compute the predicted node activation probabilities on the test pseudo-traces. The only difference is that when fitting the GLT model, we use a larger parameter grid for the Beta threshold distribution:

$$F_v \sim \text{Beta}(\alpha_v, \beta_v), \text{ where } (\alpha_v, \beta_v) \in \{1, 2, \dots, 10\}^2.$$

Note that for the fitting problem (21), we only require  $F_v$  to be log-concave, which is satisfied when both  $\alpha$  and  $\beta$  are greater than or equal to 1 for the Beta distribution. This differs from solving the IM problem under the estimated model, which would require  $F_v$  to be concave, as indicated by Corollary 3.1. Figure 8 presents the resulting ROC curves and AUC scores for the estimated diffusion models. To check the robustness of these results to the choice of the training-test split, we repeat the process five times with different random seeds. The obtained AUC scores differ from those in Figure 8 by less than 0.001, so we do not report them here. The results show that GLT-Opt has a slight



Figure 8: ROC curves and AUC scores for the activation probabilities computed on the test pseudo-trace set by various candidate diffusion models estimated on the training pseudo-trace set. The test and training sets consist of 20% and 80% of each node's pseudo-trace collection, as defined in (18). The split is stratified by the activation status y.

advantage over LT-EM and IC-EM, while all three models significantly outperform the heuristics PTP and WC.

# 6 Discussion

In this paper, we have proposed the general linear threshold (GLT) model for information diffusion on networks, derived identifiability conditions for edge weights, developed a likelihood-based method to estimate them from fully or partially observed traces, and proved that these estimates are consistent. An important assumption needed for consistency is the compactness of the parameter space, which prevents us from using threshold distributions with unbounded support, such as the exponential distributions. While consistency could likely still be established in these cases using standard parameter space compactification techniques, we found empirically that MLE solutions obtained by solving problem (16) are not always stable for such distributions (experiments are not included in the paper but are available on GitHub). This is likely because there are some special cases, for example, if for an edge (u, v) we observe only traces where v is activated immediately after u or not at all, the weight estimate diverges. We leave the investigation of threshold distributions with unbounded support for future work.

To allow for easy use with arbitrary distribution of thresholds, our implementation of the likelihood optimization problem (16) uses the SLSQP solver, which accommodates both convex and non-convex problems. This choice was made to allow for fitting the GLT model with non-log-concave threshold densities, which may violate the convexity condition in Proposition 4.3. When convexity is guaranteed, however, optimization efficiency can be significantly improved by using convex solvers, such as Gurobi [Gurobi Optimization, LLC, 2024] or Mosek [ApS, 2019]. Incorporating alternative solver options will be incorporated in future work.

In (21), we used a grid search to estimate the parameters of threshold distributions. While this approach is reasonable for distributions with a small number of parameters, it does not scale well. A possible alternative is nonparametric estimation, as discussed at the end of Section 4.5.

Another extension we leave for future work is establishing identifiability and consistency conditions for the GLT model where both the parent edge weights  $\theta_v$  and the threshold distribution parameters  $\varphi_v$  vary with v. We should still be able to estimate the parameters of the nodes that appear in a sufficient number of traces, but additionally allowing for threshold parameters would reduce the effective sample size, and limit the usefulness of the method to applications with a larger number of traces available.

Finally, it should be straightforward to establish asymptotic normality of the MLE for both the IC and the GLT models with appropriate regularity conditions on the weight distribution. That could be used to conduct inference, constructing confidence intervals or hypothesis tests, for example, to compare influence of parent nodes on a particular user, though we did not pursue this direction as it is rarely the purpose of information spread modeling.

# References

- MOSEK ApS. The MOSEK optimization toolbox for MATLAB manual. Version 9.0., 2019. URL http://docs.mosek.com/9.0/toolbox/index.html.
- Suman Banerjee, Mamata Jenamani, and Dilip Kumar Pratihar. A survey on influence maximization in a social network. *Knowledge and Information Systems*, 62:3417–3455, 2020.
- Gérard Cornuéjols, Marshall Fisher, and George Nemhauser. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management Science*, 23:789–810, 04 1977. doi: 10.1287/mnsc.23.8.789.
- Ly Dinh and Nikolaus Nova Parulian. Covid-19 pandemic and information diffusion analysis on twitter. Proceedings of the Association for Information Science and Technology. Association for Information Science and Technology, 57, 2020.
- P. Erdös and A. Rényi. On random graphs i. Publicationes Mathematicae Debrecen, 6:290, 1959.
- Golnoosh Farnad, Behrouz Babaki, and Michel Gendreau. A unifying framework for fairness-aware influence maximization. In *Companion Proceedings of the Web Conference 2020*, WWW '20, page 714–722, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450370240. doi: 10.1145/3366424.3383555. URL https://doi.org/10.1145/3366424.3383555.
- Eleftherios Gkioulekas. On equivalent characterizations of convexity of functions. International Journal of Mathematical Education in Science and Technology, 44(3):410–417, 2013. doi: 10.1080/ 0020739X.2012.703336. URL https://doi.org/10.1080/0020739X.2012.703336.
- Jacob Goldenberg, Barak Libai, and Eitan Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters*, 2001.
- Amit Goyal, Francesco Bonchi, and Laks V. S. Lakshmanan. A data-based approach to social influence maximization. *Proc. VLDB Endow.*, 5(1):73–84, sep 2011. ISSN 2150-8097. doi: 10.14778/2047485.2047492. URL https://doi.org/10.14778/2047485.2047492.
- Mark Granovetter. Threshold models of collective behavior. *American journal of sociology*, 83(6): 1420–1443, 1978.
- Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2024. URL https://www.gurobi.com.
- Xinran He, Ke Xu, David Kempe, and Yan Liu. Learning influence functions from incomplete observations, 2016. URL https://arxiv.org/abs/1611.02305.
- J. L. W. V. Jensen. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. Acta Mathematica, 30:175–193, 1906. URL https://api.semanticscholar.org/CorpusID:120669169.
- Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001. URL http://www.scipy.org/.
- R.W. Keener. Theoretical statistics: Topics for a core course. 2010. URL https://books.google.co.in/books?id=aVJmcega44cC.
- David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. page 137-146, 2003. doi: 10.1145/956750.956769. URL https://doi.org/10.1145/956750.956769.

- Sadamori Kojaku and Naoki Masuda. Core-periphery structure requires something else in the network. New Journal of Physics, 20(4):043012, apr 2018. doi: 10.1088/1367-2630/aab547. URL https://dx.doi.org/10.1088/1367-2630/aab547.
- Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: A comparative analysis. *Physical Review E*, 80(5), nov 2009. doi: 10.1103/physreve.80.056117. URL https: //doi.org/10.1103%2Fphysreve.80.056117.
- Yang Liu and Yi-Fang Brook Wu. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.
- Elchanan Mossel and Sebastien Roch. Submodularity of infuence in social networks: From local to global. *SIAM Journal on Computing*, 39:2176–2188, 03 2010. doi: 10.1137/080714452.
- Harikrishna Narasimhan, David C Parkes, and Yaron Singer. Learnability of influence in networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper\_files/paper/2015/file/ 4a2ddf148c5a9c42151a529e8cbdcc06-Paper.pdf.
- George Nemhauser, Laurence Wolsey, and M. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14:265–294, 12 1978. doi: 10.1007/ BF01588971.
- András Prékopa. On logarithmic concave measures and functions. 1973.
- Matthew Richardson and Pedro Domingos. Mining knowledge-sharing sites for viral marketing. In Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '02, page 61–70, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 158113567X. doi: 10.1145/775047.775057. URL https://doi.org/10.1145/ 775047.775057.
- Manuel Rodriguez, Jure Leskovec, David Balduzzi, and Bernhard Schölkopf. Uncovering the structure and temporal dynamics of information propagation. *Network Science*, 2:26–65, 04 2014. doi: 10.1017/nws.2014.3.
- Kazumi Saito, Ryohei Nakano, and Masahiro Kimura. Prediction of information diffusion probabilities for independent cascade model. KES '08, page 67–75, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 9783540855668. doi: 10.1007/978-3-540-85567-5\_9. URL https://doi.org/10.1007/ 978-3-540-85567-5\_9.
- Mattia Tantardini, Francesca Ieva, Lucia Tajoli, and Carlo Piccardi. Comparing methods for comparing networks. *Scientific Reports*, 9, 11 2019. doi: 10.1038/s41598-019-53708-y.
- Bruce W. Turnbull. The empirical distribution function with arbitrarily grouped, censored, and truncated data. *Journal of the royal statistical society series b-methodological*, 38:290-295, 1976. URL https://api.semanticscholar.org/CorpusID:122426592.

# 7 Appendix

### A Proofs

### A.1 Proof of Proposition 4.3

Let F be an arbitrary cumulative distribution function with density f. We need to show that

$$F(x) - F(y) = \int_{\mathbb{R}} \mathbf{1}(y < t \le x) f(t) dt$$
(32)

is concave on  $\{(x, y) : x > y\}$ . Notice that  $g(x, y, u) := \mathbf{1}(y < t \le x)$  is a log-concave function since for any  $\lambda \in [0, 1]$  and points  $A_1 = (x_1, y_1, u_1), A_2 = (x_2, y_2, u_2)$  with  $x_1 > y_1, x_2 > y_2$ , it holds:

$$g(\lambda A_1 + (1 - \lambda)A_2) = \mathbf{1} [\lambda x_1 + (1 - \lambda)x_2 < \lambda u_1 + (1 - \lambda)u_2 \le \lambda y_1 + (1 - \lambda)y_2]$$
  

$$\geq \mathbf{1} [x_1 < u_1 \le y_1] \mathbf{1} [x_2 < u_2 \le y_2]$$
  

$$= \mathbf{1} [x_1 < u_1 \le y_1]^{\lambda} \mathbf{1} [x_2 < u_2 \le y_2]^{1 - \lambda}$$
  

$$= g(A_1)^{\lambda} g(A_2)^{1 - \lambda}.$$

Thus, the expression under the integral in (32) is log-concave as a product of log-concave functions. Finally, by Theorem 6 in Prékopa [1973], the integral of a multivariate log-concave function w.r.t. any of its arguments is also log-concave. This completes the proof.

#### A.2 Proof of Proposition 3.2

Consider a GLT model on the directed graph G = (V, E) with nodes  $V = \{1, 2, 3, 4\}$  and edges  $E = \{(1, 4), (2, 4), (3, 4)\}$  having equal weights  $b_{u,v} = 1/3$ . Let also node 4 have a threshold cdf F satisfying

$$F(0) = 0, F(1/3) = 0.5, F(2/3) = 0.85, F(1) = 1.$$

An example of such F can be constructed using Lagrange interpolation:

$$F(x) = \begin{cases} 0, & x < 0\\ -0.225x^3 - 0.45x^2 + 1.675x, & x \in [0, 1]\\ 1, & x > 1 \end{cases}$$

Notice that due to Theorem 3.2, the influence function of this GLT model is submodular since F is concave on [0, 1]:

$$F''(x) = -1.35x - 0.9 < 0, \quad x \in [0, 1]$$

and threshold distributions of nodes 1, 2, and 3 do not affect the diffusion model as other nodes cannot activate them. Due to the same reason, the Triggering model on this graph will have only 8 parameters representing probabilities of node 4 to choose each possible subset of  $\{1, 2, 3\}$  as its triggering set. We will denote this probability distribution as

$$\mathcal{P} = \{P_{\emptyset}, P_1, P_2, P_3, P_{12}, P_{13}, P_{23}, P_{123}\}$$

If the GLT model was a special case of the Triggering model, we could find an appropriate distribution  $\mathcal{P}$  such that the activation probability of node 4 is the same for the two models given any seed set, i.e. the following linear system should hold:

$$\begin{cases} P_1 + P_{12} + P_{13} + P_{123} &= F(b_{1,4}) = 0.5 \\ P_2 + P_{12} + P_{23} + P_{123} &= F(b_{2,4}) = 0.5 \\ P_3 + P_{13} + P_{23} + P_{123} &= F(b_{3,4}) = 0.5 \\ P_1 + P_2 + P_{12} + P_{13} + P_{23} + P_{123} &= F(b_{1,3} + b_{2,4}) = 0.85 \\ P_1 + P_3 + P_{12} + P_{13} + P_{23} + P_{123} &= F(b_{1,3} + b_{3,4}) = 0.85 \\ P_2 + P_3 + P_{12} + P_{13} + P_{23} + P_{123} &= F(b_{1,3} + b_{2,4}) = 0.85 \\ P_1 + P_2 + P_3 + P_{12} + P_{13} + P_{23} + P_{123} &= F(b_{1,3} + b_{2,4}) = 0.85 \\ P_1 + P_2 + P_3 + P_{12} + P_{13} + P_{23} + P_{123} &= F(b_{1,4} + b_{2,4} + b_{3,4}) = 1 \end{cases}$$

Solving this system we obtain:

$$\begin{pmatrix} P_1 \\ P_2 \\ P_3 \\ P_{12} \\ P_{13} \\ P_{123} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 1 & 1 & -1 \\ 0 & -1 & 0 & 1 & 0 & 1 & -1 \\ -1 & 0 & 0 & 1 & 1 & 0 & -1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 0.5 \\ 0.5 \\ 0.85 \\ 0.85 \\ 0.85 \\ 1 \end{pmatrix}$$

But expanding the last row of the matrix we obtain that probability  $P_{123} = 0.5 \cdot 3 - 0.85 \cdot 3 + 1 = -0.05$  is a negative number. Contradiction.

#### A.3 Proof of Theorem 3.2

We start with the following technical lemma:

**Lemma 7.1.** Consider cdf F of the distribution supported on [0,h]. Then the condition

$$F(x+b) - F(x) \ge F(y+b) - F(y)$$
 (33)

holds for all triples (x, y, b) with  $0 \le x \le y \le y + b \le h$  if and only if F is concave on [0, h].

Proof of Lemma 7.1. (Necessity) It is enough to verify the "midpoint" concavity condition:

$$F\left(\frac{x'+y'}{2}\right) \ge \frac{F(x')+F(y')}{2}, \quad 0 \le x' \le y' \le C$$

since for bounded functions (cdf is bounded between 0 and 1), it is equivalent to concavity (see Section 5 of Jensen [1906]). Plugging x = x', b = (y' - x')/2, y = (y' + x')/2 into (33) and rearranging the terms implies the needed inequality.

(Sufficiency) First notice that WLOG we can assume that  $x + b \le y$ . Indeed, if x + b > y we can make a substitution x' = x, y' = x + b, b' = y - x and rearrange the terms in the target inequality to obtain:

$$F(y) - F(x) \ge F(y+b) - F(x+b) \iff F(x'+b') - F(x') \ge F(y'+b') - F(y')$$

where clearly  $x' + b' \le y'$ . Now we are left to demonstrate that for  $x \le x + b \le y \le y + b$  and concave cdf F, it holds:

$$F(x+b) - F(x) \ge F(y+b) - F(y)$$

This, in turn, follows from the equivalent definition of a concave function (Lemma 2.1 in Gkioulekas [2013]) which states that for any  $x_1 < x_2 < x_3$  it should hold:

$$\frac{F(x_2) - F(x_1)}{x_2 - x_1} \ge \frac{F(x_3) - F(x_2)}{x_3 - x_2}$$

Indeed, applying this inequality to (x, x + b, y) and (x + b, y, y + b) we deduce what was needed:

$$\frac{F(x+b) - F(x)}{b} \ge \frac{F(y) - F(x+b)}{y - x - b} \ge \frac{F(y+b) - F(y)}{b}.$$

The proof is complete.

Proof of Theorem 3.2. Consider a graph G = (V, E) and an arbitrary GLT model on it. By Theorem 3.1, it is enough to show that all threshold functions  $f_v(S) = F_v\left(\sum_{u \in S} b_{u,v}\right)$  are monotone and submodular. Monotonicity holds trivially since all edge weights are nonnegative and  $F_v$  is non-decreasing. To establish submodularity, we need to check for  $S' \subset S \subset P(v)$  and  $w \notin S$  that

$$F_{v}\left(\sum_{u\in S\cup\{w\}}b_{u,v}\right) - F_{v}\left(\sum_{u\in S}b_{u,v}\right) \le F_{v}\left(\sum_{u\in S'\cup\{w\}}b_{u,v}\right) - F_{v}\left(\sum_{u\in S'}b_{u,v}\right).$$

This follows by applying Lemma 7.1 to  $b := b_{w,v}, x := \sum_{u \in S'} b_{u,v}, y := \sum_{u \in S} b_{u,v}$ . The condition  $0 \le x \le y \le y + b \le h_v$  follows from weights' non-negativity:

$$0 \le \sum_{u \in S'} b_{u,v} \le \sum_{u \in S} b_{u,v} \le \sum_{u \in S \cup \{w\}} b_{u,v} \le \sum_{u \in P(v)} b_{u,v} \le h_v.$$

#### A.4 Proof of Theorem 4.1

(Sufficiency) Assume there are distinct vectors of parameters  $\boldsymbol{\theta} = \{b_{u,v} : (u,v) \in E\}$  and  $\boldsymbol{\theta} = \{b_{u,v} : (u,v) \in E\}$  in  $\tilde{\Theta}$  for which  $\mathbb{P}_{\boldsymbol{\theta}} = \mathbb{P}_{\boldsymbol{\tilde{\theta}}}$ . Since  $\boldsymbol{\theta} \neq \boldsymbol{\tilde{\theta}}$ , there is an edge  $(u,v) \in E$  such that  $b_{u,v} \neq \tilde{b}_{u,v}$ . Define the vectors of v's parent edges as

$$\boldsymbol{\theta}_v = \{b_{w,v}: w \in P(v)\} \text{ and } \boldsymbol{\theta}_v = \{b_{w,v}: w \in P(v)\}.$$

For a subset  $A \subset P(v)$ , we will also denote

$$B(v,A) := \sum_{u \in P(v) \cap A} b_{u,v} \quad \text{and} \quad \tilde{B}(v,A) := \sum_{u \in P(v) \cap A} \tilde{b}_{u,v}.$$

Consider the subsets  $S_j, j = 1, ..., m$  of P(v) together with corresponding traces  $\mathcal{D}_j$  satisfying conditions of the theorem. Notice that equality of distributions  $\mathbb{P}_{\theta}$  and  $\mathbb{P}_{\tilde{\theta}}$  implies equality of diffusion models by (1), thus

$$\mathbb{P}_{\boldsymbol{\theta}}(v \in D_{t_j+1}|\mathcal{D}_j) = \mathbb{P}_{\tilde{\boldsymbol{\theta}}}(v \in D_{t_j+1}|\mathcal{D}_j) \Rightarrow$$

$$F_v\left(B(v, A_{t_j}^{(j)})\right) - F_v\left(B(v, A_{t_j-1}^{(j)})\right) = F_v\left(\tilde{B}(v, A_{t_j}^{(j)})\right) - F_v\left(\tilde{B}(v, A_{t_j-1}^{(j)})\right). \tag{34}$$

Notice that here we used the fact that  $v \notin A_{t_j}^{(j)}$  since otherwise both probabilities would be zero. In case  $t_j = 0$ , i.e. if  $A_{t_j}^{(j)} = D_0^{(j)}$ , the negative term becomes zero on both sides due to  $A_{-1}^{(j)} = \emptyset$ . Thus, by invertibility of  $F_v$  we deduce  $B(v, D_0^{(j)}) = \tilde{B}(v, D_0^{(j)})$ . But the set of v's parents lying in  $D_0^{(j)}$  is exactly  $S_j$ , thus:

$$B(v, S_j) = \tilde{B}(v, S_j) \tag{35}$$

In case  $t_j > 0$ , the trace  $\mathcal{D}_j^{-1} := (\mathcal{D}_0^{(j)}, \dots, \mathcal{D}_{t_j-1}^{(j)})$  becomes also feasible by Lemma 4.1. Therefore, we have:

$$\mathbb{P}_{\boldsymbol{\theta}}(v \notin D_{t_j} | \mathcal{D}_j^{-1}) = \mathbb{P}_{\tilde{\boldsymbol{\theta}}}(v \notin D_{t_j} | \mathcal{D}_j^{-1}) \Rightarrow$$
$$F_v \left( B(v, A_{t_j-1}^{(j)}) \right) = F_v \left( \tilde{B}(v, A_{t_j-1}^{(j)}) \right).$$

Summing this equality with (34) and inverting  $F_v$  implies that both  $B(v, A_{t_j-1}^{(j)}) = \tilde{B}(v, A_{t_j-1}^{(j)})$  and  $B(v, A_{t_j}^{(j)}) = \tilde{B}(v, A_{t_j}^{(j)})$  should hold. But the difference between the two equations again gives (35) since

$$S_j = D_{t_j}^{(j)} \cap P(v) = \left(A_{t_j}^{(j)} \cap P(v)\right) \setminus \left(A_{t_j-1}^{(j)} \cap P(v)\right).$$

Combining (35) across all  $S_j, j = 1, ..., m$ , we obtain a linear system which in matrix form can be written as  $X_v^T \boldsymbol{\theta}_v = X_v^T \tilde{\boldsymbol{\theta}}_v$ . But according to our assumption,  $X_v$  is invertible, thus  $\boldsymbol{\theta}_v = \tilde{\boldsymbol{\theta}}_v$  and  $b_{u,v} = \tilde{b}_{u,v}$  in particular. Contradiction.

(Necessity) Assume  $\{\mathbb{P}_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \tilde{\Theta}\}\$  is identifiable but there is  $v \in V_c$  with  $P(v) = \{u_1, \ldots, u_m\}$  for which conditions of the theorem do not hold. Take arbitrary  $\boldsymbol{\theta} \in \tilde{\Theta}$  such that  $\boldsymbol{\theta}_v = \frac{h_v}{m+1} \mathbf{1}_m$ (it is in  $\tilde{\Theta}_v$  since  $\varepsilon_0 < \frac{h_v}{m+1}$ ). Our further goal is to obtain a contradiction by constructing  $\tilde{\boldsymbol{\theta}} \in \tilde{\Theta}$  coinciding with  $\boldsymbol{\theta}$  everywhere except for  $\boldsymbol{\theta}_v$  so that  $\mathbb{P}_{\tilde{\boldsymbol{\theta}}} = \mathbb{P}_{\boldsymbol{\theta}}$ . Consider all possible subsets  $S_j \subset P(v), j = 1, \ldots, k$  which satisfy condition 1 of the theorem. Then, our assumption implies that the matrix  $X_v = [\mathbf{1}(u_i \in S_j)] \in \{0, 1\}^{m \times k}$  has  $\operatorname{rank}(X_v) < m$ . In other words, there is a non-zero vector  $z \in \mathbb{R}^m$  such that  $X_v^T z = \mathbf{0}_k$ . Thus, for any scalar  $\delta > 0$ , we can define  $\tilde{\boldsymbol{\theta}}_v = \boldsymbol{\theta}_v + \delta z$ to make it different from  $\boldsymbol{\theta}_v$  while preserving

$$X_v^T \boldsymbol{\theta}_v = X_v^T \tilde{\boldsymbol{\theta}}_v. \tag{36}$$

We pick  $\delta := \left(\frac{h_v}{m+1} - \varepsilon_0\right) / \|z\|_1$  so that  $\tilde{\theta}_v \in \tilde{\Theta}_v$ . Indeed, by triangular inequality,  $\tilde{\theta}_v$  satisfies all the needed constraints:

$$\|\tilde{\boldsymbol{\theta}}_{v}\|_{1} \leq \|\boldsymbol{\theta}_{v}\|_{1} + \delta \|z\|_{1} = \frac{mh_{v}}{m+1} + \frac{h_{v}}{m+1} - \varepsilon_{0} = h_{v} - \varepsilon_{0},$$
$$\tilde{\boldsymbol{\theta}}_{v} \geq \boldsymbol{\theta}_{v} - \delta \|z\|_{1} \mathbf{1}_{m} \geq \varepsilon_{0}.$$

Notice that by changing  $\boldsymbol{\theta}_v$  alone, we could only change the probability of a feasible trace  $\mathcal{D} = (D_0, \ldots, D_T)$  with  $\mathbb{P}^0(D_0) > 0$  and  $P(v) \cap D_t \neq \emptyset$  for some  $t \leq t_v := \arg \max_{\tau \leq T} \{\tau : v \notin A_\tau\}$  where  $t_v$  is the last time v is not activated. Indeed, if  $\mathbb{P}^0(D_0) = 0$  then  $\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}) = \mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}) = 0$  and if  $P(v) \cap D_t = \emptyset$  for any  $t \leq t_v$ , then by (4), trace probability does not depend on  $\boldsymbol{\theta}_v$ . Take arbitrary such trace and consider all times  $s_j \leq t_v, j = 1 \dots, r$  for which  $P(v) \cap D_{s_j} \neq \emptyset$ . Notice that for each time  $s_j$  the trace  $(D_0, \ldots, D_{s_j})$  is also feasible and satisfies condition 1 of the theorem. Therefore, there is a corresponding column  $x_j := [\mathbf{1}(u_i \in D_{s_j})]_{i=1}^m$  in matrix  $X_v$  which according to (36) satisfies:

$$B(v, D_{s_j}) = \langle x_j, \boldsymbol{\theta}_v \rangle = \langle x_j, \boldsymbol{\theta}_v \rangle = B(v, D_{s_j}).$$

Summing these equations over  $j \leq r$  and j < r, we obtain respectively:

 $B(v, A_{t_v}) = \tilde{B}(v, A_{t_v})$  and  $B(v, A_{t_v-1}) = \tilde{B}(v, A_{t_v-1}).$ 

But from (4), trace probability is either a function of  $B(v, A_T) = B(v, A_{t_v})$  if  $v \notin A_T$  or of  $B(v, A_{t_v})$ and  $B(v, A_{t_v-1})$  if  $v \in A_T$ . Therefore, we deduce  $\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}) = \mathbb{P}_{\tilde{\boldsymbol{\theta}}}(\mathcal{D})$  and finally get the contradiction with identifiability of  $\tilde{\Theta}$ .

#### A.5 Identifiability and consistency for pseudo-traces.

(Proof of Theorem 4.3.) It is enough to check that for  $G_v$ , condition 1 in this theorem and Theorem 4.1 are equivalent. First, consider a subset  $S \subset P(v)$  with  $\mathbb{P}_v^0(S) > 0$ . Then clearly the trace that stopped at the seed set S without propagating further satisfies condition 1 of Theorem 4.1. Now, assume there is a trace  $(D_0, \ldots, D_t) \in \mathcal{F}(G_v)$  with  $\mathbb{P}_v^0(D_0) > 0$ ,  $v \notin A_t$ , and  $D_t \cap P(v) = S$ . Notice that given a seed set  $D_0 \subset P(v)$ , the only feasible traces on  $G_v$  we can get are either  $(D_0, )$  or  $(D_0, \{v\})$ . But  $v \notin A_t$ , thus t = 0 and  $D_0 = S$ . The proof is complete.

(Proof of Theorem 4.4). Follows directly from Theorem 4.2 applied to  $G_v$ .

#### A.6 Identifiability and consistency under the IC model.

By Definition 5, the likelihood of a feasible trace  $\mathcal{D} = (D_0, \ldots, D_T)$  sampled from the IC model with edge probabilities  $\boldsymbol{\theta} := \{p_{u,v} : (u,v) \in E\}$  and the seed distribution  $\mathbb{P}^0$  can be written as

$$\mathbb{P}_{\theta}(\mathcal{D}) = \mathbb{P}^{0}(D_{0}) \times \prod_{t=0}^{T-1} \left\{ \prod_{u \in D_{t}} \prod_{v \in C(D_{t}) \setminus A_{t+1}} (1 - p_{u,v}) \prod_{v \in D_{t+1}} \left[ 1 - \prod_{u \in D_{t} \cap P(v)} (1 - p_{u,v}) \right] \right\}.$$
 (37)

Similarly to Assumption 2, we will assume that the ground-truth probabilities are separated from zero and one for identifiability:

**Assumption 3.** There is a universal constant  $\varepsilon_0 \in (0, 0.5)$  such that the true edge probabilities satisfy  $\varepsilon_0 \leq p_{u,v} \leq 1 - \varepsilon_0$  for all  $(u, v) \in E$ .

Then, for the truncated parameter space

$$\tilde{\Theta}_{IC} = \{ p_{u,v} : \varepsilon_0 \le p_{u,v} \le 1 - \varepsilon_0 \text{ for all } (u,v) \in E \}$$
(38)

we can formulate the same identifiability condition as for the GLT model:

**Theorem 7.1.** Parametric family  $\{\mathbb{P}_{\theta}, \theta \in \tilde{\Theta}_{IC}\}$  is identifiable if and only if for each child node  $v \in V_c$  with  $P(v) = \{u_1, \ldots, u_m\}$ , there exist  $S_1, \ldots, S_m \subseteq P(v)$  such that

- 1. For each j = 1, ..., m, there is a feasible trace  $(D_0^{(j)}, ..., D_{t_j}^{(j)}) \in \mathcal{F}(G)$  with  $\mathbb{P}^0(D_0^{(j)}) > 0$ ,  $v \notin A_{t_j}^{(j)}$ , and  $D_{t_j}^{(j)} \cap P(v) = S_j$ .
- 2. The matrix  $X_v = [\mathbf{1}(u_i \in S_j)]_{i,j=1}^m$  is invertible.

*Proof.* (Sufficiency) Assume there are distinct vectors of parameters  $\boldsymbol{\theta} = \{p_{u,v} : (u,v) \in E\}$  and  $\tilde{\boldsymbol{\theta}} = \{\tilde{p}_{u,v} : (u,v) \in E\}$  in  $\tilde{\Theta}_{IC}$  for which  $\mathbb{P}_{\boldsymbol{\theta}} = \mathbb{P}_{\tilde{\boldsymbol{\theta}}}$ . Since  $\boldsymbol{\theta} \neq \tilde{\boldsymbol{\theta}}$ , there is an edge  $(u,v) \in E$  such that  $p_{u,v} \neq \tilde{p}_{u,v}$ . Define the vectors of v's parent edge probabilities

$$\boldsymbol{\theta}_v = \{ p_{w,v} : w \in P(v) \} \text{ and } \boldsymbol{\theta}_v = \{ \tilde{p}_{w,v} : w \in P(v) \}$$

as well as vectors

$$\ell_v = \{ \log(1 - p_{w,v}) : w \in P(v) \}$$
 and  $\tilde{\ell}_v = \{ \log(1 - \tilde{p}_{w,v}) : w \in P(v) \}.$ 

For a subset  $A \subset P(v)$ , we will also denote

$$Q(v,A) := \sum_{u \in P(v) \cap A} \log(1-p_{u,v}) \quad \text{and} \quad \tilde{Q}(v,A) := \sum_{u \in P(v) \cap A} \log(1-\tilde{p}_{u,v}).$$

Consider the subsets  $S_j, j = 1, ..., m$  of P(v) together with corresponding traces  $\mathcal{D}_j$  satisfying conditions of the theorem. Notice that equality of distributions  $\mathbb{P}_{\theta}$  and  $\mathbb{P}_{\tilde{\theta}}$  implies equality of diffusion models by (1), thus

$$1 - \prod_{u \in D_{t_j}^{(j)} \cap P(v)} (1 - p_{u,v}) = \mathbb{P}_{\theta}(v \in D_{t_j+1} | \mathcal{D}_j) = \mathbb{P}_{\tilde{\theta}}(v \in D_{t_j+1} | \mathcal{D}_j) = 1 - \prod_{u \in D_{t_j}^{(j)} \cap P(v)} (1 - \tilde{p}_{u,v}) + C_{t_j}^{(j)} \cap P(v)$$

Notice that here we used the fact that  $v \notin A_{t_j}^{(j)}$  since otherwise both probabilities would be zero. But by definition of  $\mathcal{D}_j$ , we have  $D_{t_j}^{(j)} \cap P(v) = S_j$ , therefore subtracting one and taking logarithm from both sides implies

$$Q(v, S_j) = \tilde{Q}(v, S_j).$$

Combining these equations across all  $S_j, j = 1, ..., m$ , we obtain a linear system which in matrix form can be written as  $X_v^T \boldsymbol{\ell}_v = X_v^T \tilde{\boldsymbol{\ell}}_v$ . But according to our assumption,  $X_v$  is invertible, thus  $\boldsymbol{\ell}_v = \tilde{\boldsymbol{\ell}}_v$ . Element-wise exponentiation implies  $p_{u,v} = \tilde{p}_{u,v}$ . Contradiction.

(Necessity) Assume  $\{\mathbb{P}_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \tilde{\Theta}_{IC}\}\$  is identifiable but there is  $v \in V_c$  with  $P(v) = \{u_1, \ldots, u_m\}$ for which conditions of the theorem do not hold. Take arbitrary  $\boldsymbol{\theta} \in \tilde{\Theta}_{IC}$  such that  $\boldsymbol{\theta}_v = \mathbf{1}_m/2$  (it is in  $\tilde{\Theta}$  since  $\varepsilon_0 < 0.5$ ). Our further goal is to obtain a contradiction by constructing  $\tilde{\boldsymbol{\theta}} \in \tilde{\Theta}$ coinciding with  $\boldsymbol{\theta}$  everywhere except for  $\boldsymbol{\theta}_v$  so that  $\mathbb{P}_{\tilde{\boldsymbol{\theta}}} = \mathbb{P}_{\boldsymbol{\theta}}$ . Consider all possible subsets  $S_j \subset P(v), j = 1, \ldots, k$  which satisfy condition 1 of the theorem. Then, our assumption implies that the matrix  $X_v = [\mathbf{1}(u_i \in S_j)] \in \{0, 1\}^{m \times k}$  has  $\operatorname{rank}(X_v) < m$ . In other words, there is a non-zero vector  $z \in \mathbb{R}^m$  such that  $X_v^T z = \mathbf{0}_k$ . Thus, for any scalar  $\delta > 0$ , we can define  $\tilde{\ell}_v = \ell_v + \delta z$ to make it different from  $\ell_v$  while preserving

$$X_v^T \boldsymbol{\ell}_v = X_v^T \tilde{\boldsymbol{\ell}}_v. \tag{39}$$

We pick  $\delta := \log (2 - 2\varepsilon_0) / ||z||_1$  so that  $\tilde{\theta}_v$  satisfies the parameter space constraints:

$$\varepsilon_0 \mathbf{1}_m \le \frac{1}{4(1-\varepsilon_0)} \mathbf{1}_m \le \exp(\boldsymbol{\ell}_v - \delta \|\boldsymbol{z}\|_1) \mathbf{1}_m \le \tilde{\boldsymbol{\theta}}_v \le \exp(\boldsymbol{\ell}_v + \delta \|\boldsymbol{z}\|_1) \mathbf{1}_m \le (1-\varepsilon_0) \mathbf{1}_m$$

Notice that by changing  $\boldsymbol{\theta}_v$  alone, we could only change the probability of a feasible trace  $\mathcal{D} = (D_0, \ldots, D_T)$  with  $\mathbb{P}^0(D_0) > 0$  and  $P(v) \cap D_t \neq \emptyset$  for some  $t \leq t_v := \arg \max_{\tau \leq T} \{\tau : v \notin A_\tau\}$  where  $t_v$  is the last time v is not activated. Indeed, if  $\mathbb{P}^0(D_0) = 0$  then  $\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}) = \mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}) = 0$  and if  $P(v) \cap D_t = \emptyset$  for any  $t \leq t_v$ , then by (37), trace probability does not depend on  $\boldsymbol{\theta}_v$ . Take arbitrary such trace and consider all times  $s_j \leq t_v, j = 1 \dots, r$  for which  $P(v) \cap D_{s_j} \neq \emptyset$ . Notice that for each time  $s_j$  the trace  $(D_0, \ldots, D_{s_j})$  is also feasible and satisfies condition 1 of the theorem. Therefore, there is a corresponding column  $x_j := [\mathbf{1}(u_i \in D_{s_j})]_{i=1}^m$  in matrix  $X_v$  which according to (39) satisfies:

$$Q(v, D_{s_j}) = \langle x_j, \boldsymbol{\ell}_v \rangle = \langle x_j, \boldsymbol{\ell}_v \rangle = \tilde{Q}(v, D_{s_j}).$$

Summing these equations over  $j \leq r$  and j < r, we obtain respectively:

$$Q(v, A_{t_v}) = \tilde{Q}(v, A_{t_v})$$
 and  $Q(v, A_{t_v-1}) = \tilde{Q}(v, A_{t_v-1}).$ 

But from (37), trace probability is either a function of  $Q(v, A_T) = Q(v, A_{t_v})$  if  $v \notin A_T$  or of  $Q(v, D_{t_v})$ and  $Q(v, A_{t_v-1})$  if  $v \in A_T$ . Therefore, we deduce  $\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{D}) = \mathbb{P}_{\tilde{\boldsymbol{\theta}}}(\mathcal{D})$  and finally get the contradiction with identifiability of  $\tilde{\Theta}_{IC}$ .

Now, assume that as in (11), we observe a collection of N independent traces  $\mathcal{T}$  sampled from the IC model, by which we mean that  $D_0^{(n)}$ ,  $n = 1, \ldots, N$  are independently sampled from  $\mathbb{P}^0$  and the edge activation random variables

$$\mathcal{Z}_n = \{Z_{u,v}^{(n)} : (u,v) \in E\}$$

are also independent across traces. Then, we can estimate  $\boldsymbol{\theta}$  by solving

$$\max_{\boldsymbol{\theta}\in\tilde{\Theta}_{IC}}\sum_{n=1}^{N}L(\mathcal{D}_{n}|\boldsymbol{\theta})$$
(40)

where the log-likelihood of the trace  $\mathcal{D}_n$ , by (37), takes the form:

$$L(\mathcal{D}_{n}|\boldsymbol{\theta}) = \sum_{t=0}^{T_{n}-1} \left\{ \sum_{u \in D_{t}^{(n)}} \sum_{v \in C(D_{t}^{(n)}) \setminus A_{t+1}^{(n)}} \log(1-p_{u,v}) + \sum_{v \in D_{t+1}^{(n)}} \log\left[1-\prod_{u \in D_{t}^{(n)} \cap P(v)} (1-p_{u,v})\right] \right\}.$$
(41)

Notice that problem (40) is non-convex which motivated Saito et al. [2008] to use the EM-algorithm for estimating the IC model parameters. However, we still can state the following consistency theorem for the global optimum of problem (40):

**Theorem 7.2.** Under Assumptions 3, identifiability condition in Theorem 7.1, and the assumption of *i.i.d.* traces, the MLE  $\hat{\theta}_N$  of  $\theta^*$  obtained by solving the optimization problem (12) is consistent, *i.e.*,  $\hat{\theta}_N \to \theta^*$  in probability as  $N \to \infty$ .

*Proof.* Similarly to Theorem 4.2, the result follows from compactness of (38), continuity of (41), and the fact that under assumption 3, every trace with  $\mathbb{P}^0(D_0) > 0$  has positive probability (37).

# **B** Additional Experiments

#### B.1 Grid Search for Beta distribution

Let G be a 100-node graph generated from the directed version of the ER model [Erdös and Rényi, 1959] with p = 0.1. For this graph, consider an instance of the GLT model with  $F_v^* \sim \text{Beta}(2,2)$  for all nodes v and weights generated as in (27). From this GLT model, we generate 2500 traces and split them randomly into 2000 train and 500 test traces. Finally, for each pair of threshold parameters in the grid

$$(\alpha, \beta) \in \{1, 1.25, \dots, 3\}^2$$

we set all threshold distributions  $F_v$  as  $\text{Beta}(\alpha, \beta)$ , estimate the weights  $\hat{\theta}$  by solving (16) using the train traces, and compute the log-likelihood on the test set under  $\hat{\theta}$ . The following heatmap shows that the highest log-likelihood is achieved for the true  $(\hat{\alpha}, \hat{\beta}) = (2, 2)$ . Moreover, relatively high log-likelihood values for  $(\alpha, \beta)$  pairs close to the  $\alpha = \beta$  line show that guessing the mean structure of the distribution ( $\mathbb{E} \operatorname{Beta}(\alpha, \beta) = \frac{\alpha}{\alpha+\beta} \approx 0.5$ ) is already enough for accurate model estimation.



Figure 9: Each cell of the heatmap represents the test log-likelihood of a GLT model with all  $F_v \sim \text{Beta}(\alpha, \beta)$  for the corresponding pair of  $\alpha$  (x-axis) and  $\beta$  (y-axis). Each diffusion model is estimated by fitting problem (16) with 2000 train traces, and log-likelihood is evaluated on 500 test traces. The ground-truth trace generating model is the GLT model with  $F_v \sim \text{Beta}(2, 2)$ , edge weights sampled as in (27), and seed sets sampled as in (26).

#### B.2 Estimation of the LT model using problem (16) vs Algorithm 2

To compare problem (16) (LT-Opt) and Algorithm 2 (LT-EM) in terms of the speed and accuracy of fitting the LT model, we conduct two experiments. In the first one, we generate two directed ER graphs with n = 100 and  $p \in \{0.1, 0.4\}$ . For each of them, we sample 4000 traces from the LT model with weights generated as in (27) and seed sets as in (26). Then for each trace set size  $N \in \{250, 500, 1000, 2000, 4000\}$ , we fit the model with the first N traces out of these 4000 and note the RMAE (defined in (29)) between the estimated and ground-truth weights. In the second experiment, we fix the number of train traces N = 2000 and for two sizes of the ER network  $n \in \{100, 200\}$ , check how their density affects the run time of the estimation procedures. We change the density by varying the ER edge probability  $p \in \{0.05, 0.1, 0.2, 0.4\}$ . The ground-truth LT model again has weights generated as in (27) and seed sets as in (26). Both experiments are run on 1 CPU to compare the methods in a scenario when no parallelization resources are available.

Results are presented in Figure 10. In the left plot, we can see that LT-Opt has a small edge in terms of the estimation accuracy which becomes less apparent with the increase of the network density (parameter p). On the other hand, the right plot demonstrates that with no parallelization resources at hand, the run time of LT-Opt is catastrophic compared to LT-EM, especially at higher values of the network density.



Figure 10: (Left) Run time comparison of problem (16) with all  $F_v \sim \text{Unif}[0, 1]$  and Algorithm 2 for different number of traces in the train set. Graphs are generated from the directed ER model with n = 100 and  $p \in \{0.1, 0.4\}$ . Confidence bars correspond to 1 standard deviation computed across 5 trace sets. (Right) Comparison of the RMAE between the estimated and true LT weights across graphs generated from the directed ER model with different values of p. The train set size is fixed to 2000 traces for all p. In both experiments, the ground-truth diffusion model is the LT model with weight initialization following (27) and seed sets sampled as in (26).