Convex Optimization

Prof. Amiri

Electrical Engineering Department

Naser Kazemi 99102059
Mohammad Moshtaghi 99109047

Project

February 2, 2024

Convex Optimization

Naser Kazemi 99102059
Mohammad Moshtaghi 99109047

A Review of Uncovering the Temporal Dynamics of Diffusion Networks

Abstract

In the field of Social Computation, Time plays an essential role in the diffusion of information, influence and disease over networks. Mostly, we have no information about connection of a network and we only know about when a node copies information, makes a decision or becomes infected. The paper we are reviewing, tries to find the structure of the network and transmission rates between nodes using a simple idea, Maximum Likelihood. Finally, prior to some assumptions we obtain a convex optimization problem.

1. Introduction

This paper addresses the challenge of understanding diffusion processes, such as information propagation, virus spread, and product adoption, by introducing a method to infer hidden mechanisms based on observed cascades. The proposed model assumes static but unknown networks, binary infections, independent edge infections, and varying infection times. By formulating a generative probabilistic model, the paper presents a scalable algorithm capable of reconstructing network connectivity and temporal dynamics. The innovation lies in modeling diffusion as a discrete network of independent temporal processes, avoiding the need to explicitly model underlying mechanisms and enabling large-scale analyses. The focus is on continuous temporal dynamics, a novel approach in diffusion modeling.

2. Problem Definition

Data

Observations are recorded on a fixed population of N nodes and consist of a set $\mathcal{C}$ of cascades $\{\boldsymbol{t}^1, ..., \boldsymbol{t}^{|C|}\}$. each cascade $\boldsymbol{t}^c$ is a N-dimensional vector which is as follows:

$$\boldsymbol{t}^c = (t_1^c, \ldots, t_N^c)$$

In this formulation, t_i^c represent the infection time of node i in cascade c and we assume that this data is given to your model. we consider our network as a fully connected graph and the edge between nodes i, j has a transmission rate, called $\alpha_{i,j}$. our goal is to find this transmission rates using only the data we have talked about.

Pairwise transmission likelihood

The analysis of diffusion dynamics starts with the consideration of pairwise interactions, assuming that infections can occur at varying rates across the network's edges. The objective is to deduce the transmission rates between node pairs.

The transmission likelihood from node j to node i, denoted as $f(t_i|t_j; \alpha_{j,i})$, is contingent upon the infection times t_j and t_i, and the transmission rate $\alpha_{j,i}$. A node j can infect node

i only if $t_j < t_i$. The paper discusses three parametric models to represent this likelihood: exponential, power-law, and Rayleigh.

Model	Transmission likelihood	Log survival function	Hazard function
Exponential (Exp)	$f(t_i\|t_j;\alpha_{j,i}) = \begin{cases} \alpha_{j,i}\cdot e^{-\alpha_{j,i}(t_i-t_j)} & \text{if } t_i > t_j \\ 0 & \text{otherwise} \end{cases}$	$\log S(t_i\|t_j;\alpha_{j,i}) = -\alpha_{j,i}(t_i - t_j)$	$H(t_i\|t_j;\alpha_{j,i}) = \alpha_{j,i}$
Power law (Pow)	$f(t_i\|t_j;\alpha_{j,i}) = \begin{cases} \frac{\alpha_{j,i}}{\delta}\cdot\left(\frac{t_i-t_j}{\delta}\right)^{-1-\alpha_{j,i}} & \text{if } t_i > t_j+\delta \\ 0 & \text{otherwise} \end{cases}$	$\log S(t_i\|t_j;\alpha_{j,i}) = -\alpha_{j,i}\log\left(\frac{t_i-t_j}{\delta}\right)$	$H(t_i\|t_j;\alpha_{j,i}) = \frac{\alpha_{j,i}}{t_i-t_j}$
Rayleigh (RAY)	$f(t_i\|t_j;\alpha_{j,i}) = \begin{cases} \alpha_{j,i}\cdot(t_i-t_j)\cdot e^{-\frac{1}{2}\alpha_{j,i}(t_i-t_j)^2} & \text{if } t_i > t_j \\ 0 & \text{otherwise} \end{cases}$	$\log S(t_i\|t_j;\alpha_{j,i}) = -\frac{1}{2}\alpha_{j,i}(t_i-t_j)^2$	$H(t_i\|t_j;\alpha_{j,i}) = \alpha_{j,i}\cdot(t_i-t_j)$

Table 1: Transmission likelihood, log survival function, and hazard function for different models.

Exponential model is appropriate for standard infections, characterized by a constant hazard rate. The Power-Law model describes infections with long-tail distributions. The Rayleigh model is a non-monotonic parametric model, traditionally used in epidemiology, and is well-suited for scenarios where infection likelihood rapidly ascends to a peak and then declines just as quickly.

Network Inference Problem

Our final problem would be as follows:

$$\begin{aligned} \min_{A} \quad & -\sum_{c\in\mathcal{C}} \log f(\boldsymbol{t}^c; A) \\ \text{s.t.} \quad & \alpha_{i,j} \geq 0 \qquad i,j = 1,\ldots,N, i \neq j \end{aligned}$$

Where A is the matrix with $\alpha_{i,j}$ in its ith row and jth column. Now we prove that this problem is a convex optimization problem under some assumptions.

We recall some additional standard notation. The cumulative density function, denoted $F(t_i|t_j;\alpha_{j,i})$, is computed from the transmission likelihoods. Given that node j was infected at time t_j, the *survival function* of edge $j \to i$ is the probability that node i is *not* infected by node j by time t_i:

$$S(t_i|t_j;\alpha_{j,i}) = 1 - F(t_i|t_j;\alpha_{j,i}).$$

The *hazard function*, or instantaneous infection rate, of edge $j \to i$ is the ratio

$$H(t_i|t_j;\alpha_{j,i}) = \frac{f(t_i|t_j;\alpha_{j,i})}{S(t_i|t_j;\alpha_{j,i})}.$$

The hazard functions of our models are simple, Table 1.

Next, we obtain the $f(\boldsymbol{t}^c; A)$ in terms of these two functions. Consider a cascade $\boldsymbol{t} = (t_1,\ldots,t_N)$. Lets define two new cascades in this way:

$$\boldsymbol{t}^{\geq T} = \{t_i | t_i \geq T\} \qquad \boldsymbol{t}^{\leq T} = \{t_i | t_i \leq T\}$$

Consider a cascade $t = (t_1,\ldots,t_N)$ and a node i not infected during the observation window, $t_i > T$. Since each infected node k may infect i independently, the probability that nodes $1,\ldots,N$ do not infect node i by time T is the product of the survival functions of the infected nodes $1,\ldots,N|t_k \leq T$ targeting i,

$$\prod_{t_k\leq T} S(T|t_k;\alpha_{k,i})$$

Therefore, using the assumption that infections are conditionally independent given the parents of the infected nodes, the likelihood factorizes over nodes as:

$$f(\boldsymbol{t}; A) = f(\boldsymbol{t}^{\geq T}; A) \times f(\boldsymbol{t}^{\leq T}; A) = \prod_{t_i \leq T} \prod_{t_k > T} S(T|t_i; \alpha_{i,k}) \times \prod_{t_i \leq T} f(t_i|t_1, \ldots, \hat{t_i}, \ldots, t_N; A) \quad (1)$$

Now we are going to compute the second part of the above equation. we assume that a node gets infected once the first parent infects the node. Given an infected node i, we compute the likelihood of a potential parent i to be the first parent in this way:

$$\begin{aligned} f(t_i|t_1, \ldots, \hat{t_i}, \ldots, t_N; A) &= \sum_{j:t_j<t_i} f(t_i|t_j; \alpha_{j,i}) \times \prod_{j \neq k, t_k < t_i} S(t_i|t_k; \alpha_{k,i}) \\ &= \sum_{j:t_j<t_i} \frac{f(t_i|t_j; \alpha_{j,i})}{S(t_i|t_j; \alpha_{j,i})} \times \prod_{t_k<t_i} S(t_i|t_k; \alpha_{k,i}) \\ &= \prod_{t_k<t_i} S(t_i|t_k; \alpha_{k,i}) \times \sum_{j:t_j<t_i} H(t_i|t_j; \alpha_{j,i}) \end{aligned}$$

Hence we can write Eq. 3 as follows:

$$\begin{aligned} f(\boldsymbol{t}; A) &= \prod_{t_i \leq T} \prod_{t_k > T} S(T|t_i; \alpha_{i,k}) \times \prod_{t_i \leq T} \prod_{t_k < t_i} S(t_i|t_k; \alpha_{k,i}) \times \sum_{j:t_j<t_i} H(t_i|t_j; \alpha_{j,i}) \\ &= \prod_{t_i \leq T} \left[\prod_{t_k > T} S(T|t_i; \alpha_{i,k}) \times \prod_{t_k<t_i} S(t_i|t_k; \alpha_{k,i}) \times \sum_{j:t_j<t_i} H(t_i|t_j; \alpha_{j,i}) \right] \end{aligned}$$

Finally we can compute the $\log f(\boldsymbol{t}; A)$ in this way:

$$\log f(\boldsymbol{t}; A) = \sum_{t_i \leq T} \left[\sum_{t_k > T} S(T|t_i; \alpha_{i,k}) + \sum_{t_k<t_i} S(t_i|t_k; \alpha_{k,i}) + \log \left(\sum_{j:t_j<t_i} H(t_i|t_j; \alpha_{j,i}) \right) \right]$$

Since from Table 1, S is a log-concave and H is non-negative concave function and the composition of log function with non-negative and concave function is also concave, we can conclude that $\log f(\boldsymbol{t}; A)$ is a concave function.

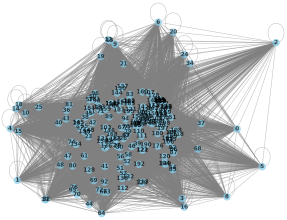
3. Implementation

We have implemented the paper problem using CVXPY library. First, we define the problem as a DCP (Disciplined Convex Programming) in python. For the input data we used Kronecker Graph dataset, which is a common dataset for scoring different approaches in the social computing literature. You can see our results in the next section of this report. You can see more explanation about our implementation in *NetRate.ipynb* jupyter notebook.

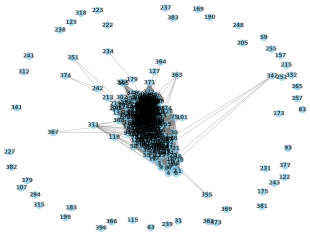
4. Experiments

Result

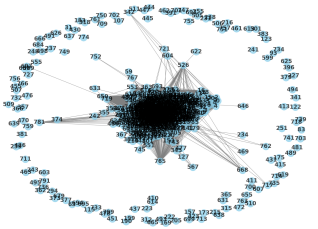
The result for estimating the network using the Kronecker graph data sets with 200, 400 and 800 nodes is:



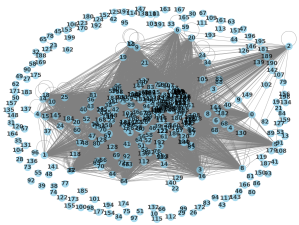
(a) Inferred Kronecker network with 200 nodes



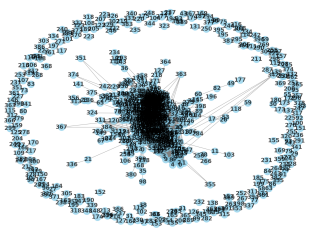
(b) Inferred Kronecker network with 400 nodes



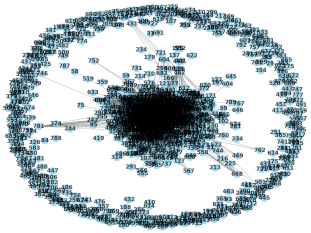
(c) Inferred Kronecker network with 800 nodes



(d) Ground-Truth Kronecker network with 200 nodes



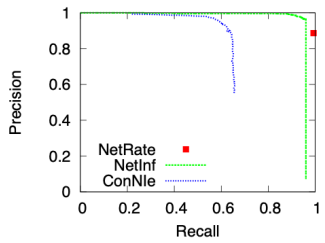
(e) Ground-Truth Kronecker network with 400 nodes



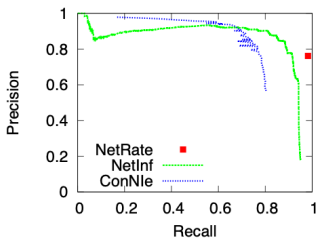
(f) Ground-Truth Kronecker network with 800 nodes

Comparison

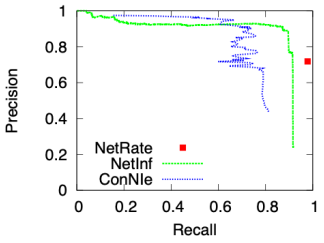
The original paper has compared the proposed algorithm with other two effective methods named NetInf and ConNie. Here we can see this comparison based on precision, recall and accuracy metrics.



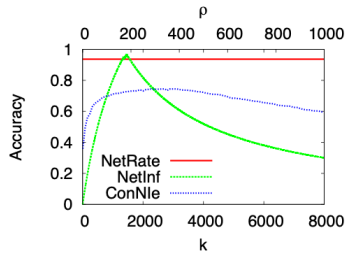
(a) Precision-recall (Hierarchical, Exp)



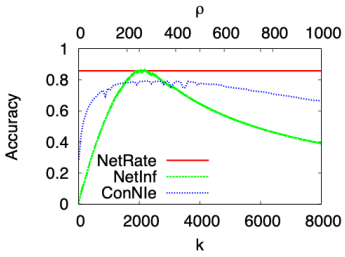
(b) Precision-recall (Forest fire, Pow)



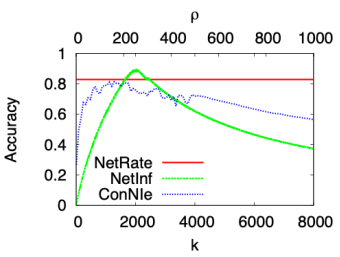
(c) Precision-recall (Random, RAY)



(d) Accuracy (Hierarchical, Exp)



(e) Accuracy (Forest fire, Pow)



(f) Accuracy (Random, RAY)