Stability of Influence Maximization

Xinran He
University of Southern California
941 Bloom Walk
Los Angeles, CA, United States
xinranhe@usc.edu

David Kempe
University of Southern California
941 Bloom Walk
Los Angeles, CA, United States
dkempe@usc.edu

ABSTRACT

In the well-studied Influence Maximization problem, the goal is to identify a set of k nodes in a social network whose joint influence on the network is maximized. A large body of recent work has justified research on Influence Maximization models and algorithms with their potential to create societal or economic value. However, in order to live up to this potential, the algorithms must be robust to large amounts of noise, for they require quantitative estimates of the influence which individuals exert on each other; ground truth for such quantities is inaccessible, and even decent estimates are very difficult to obtain.

We begin to address this concern formally. First, we exhibit simple inputs on which even very small estimation errors may mislead *every* algorithm into highly suboptimal solutions, motivating a need for algorithms that can determine whether a given instance is vulnerable to noise. Analyzing the susceptibility of specific instances to estimation errors leads to a clean algorithmic question which we term the Influence Difference Maximization problem, and for which we present an approximation algorithm based on maximizing a non-monotone submodular function.

Using the proposed techniques, we investigate the susceptibility of synthetic and real-world social network data sets. Roughly, when perturbations are on the order of 10% of the observed parameter values, the objective function is fairly stable, while relative perturbations above 20% may lead to significant instability. Our results thus suggest caution in the use of algorithmic Influence Maximization results.

Categories and Subject Descriptors

[**Human-centered computing**]: Social networks

Keywords

Influence Maximization, Uncertainty, Noise, Submodular Optimization, Robust Optimization

KDD'14, August 24–27, 2014, New York, NY, USA.

http://dx.doi.org/10.1145/2623330.2623746 .

1. INTRODUCTION

The processes and dynamics by which information and behaviors spread through social networks have long interested scientists within many areas. Understanding such processes has the potential to shed light on human social structure, and to impact the strategies used to promote behaviors or products. While the interest in the subject is long-standing, recent increased availability of social network and information diffusion data (through sites such as Facebook, Twitter, and LinkedIn) has raised the prospect of applying social network analysis at a large scale to positive effect. Consequently, the resulting algorithmic questions have received widespread interest in the computer science community.

Among the broad algorithmic domains, Influence Maximization has been repeatedly held up as having the potential to be of societal and financial value. The high-level hope is that based on observed data — such as social network information and past behavior — an algorithm could infer which individuals are likely to influence which others. This information could in turn be used to effect desired behavior, such as refraining from smoking, using superior crops, or purchasing a product. In the latter case, the goal of effecting desired behavior is usually termed *viral marketing*.

Consequently, both the problem of inferring the influence between individuals [10, 11, 18] and that of maximizing the spread of a desired behavior have been studied extensively. For the Influence Maximization problem, a large number of models have been proposed, along with many heuristics with and without approximation guarantees [5, 23, 8, 13, 14, 17]. (See the monograph [7] for a recent overview of work in the area.)

However, one crucial aspect of the problem has — with very few execeptions discussed in Section 1.6 — gone largely unstudied. Contrary to many other algorithmic domains, *noise in social network data is not an exception, but the norm.* Indeed, one could argue that the very notion of a "social link" is not properly defined in the first place, so that any representation of a social network is only an approximation of reality. This issue is much more pronounced for a goal such as Influence Maximization. Here, the required data include, for every pair (u, v) of individuals, a numerical value for the strength of influence from u to v and vice versa. This influence strength will naturally depend on context (e.g., what exact product or behavior is being spread); furthermore, it cannot be observed directly, and must therefore be inferred from observed behavior or individuals' reports; all of these are inherently very noisy.

When the inferred influence strength parameters differ from the actual ground truth, even an optimal algorithm is bound to return suboptimal solutions, for it will optimize the wrong objective function: a solution that appears good with respect to the incorrect parameters may be bad with respect to the actual ones. If relatively small errors in the inferred parameters could lead to highly suboptimal solutions, this would cast serious doubts on the practical viability of algorithmic influence maximization. *Therefore, in the present paper, we begin an in-depth study of the effect of noise on the performance of Influence Maximization algorithms.*

1.1 The Independent Cascade Model

We study this question under two widely adopted models for influence diffusion [13]: the *Independent Cascade (IC) Model* and the *Linear Threshold (LT) Model*. Both of these models fit in the following framework: The algorithm selects a *seed set* A_0 of k nodes, which begin *active* (having adopted the behavior). Starting with A_0, the process proceeds in discrete time steps: in each time step, according to a probabilistic process, additional nodes may become active based on the influence from their neighbors. Active nodes never become inactive, and the process terminates when no new nodes become active in a time step. The goal is to maximize the expected number of active nodes when the process terminates; this expected number is denoted by $\sigma(A_0)$.

To illustrate the questions and approaches, we describe the IC model in this section. (A formal description of the LT model and general definitions of all concepts are given in Section 2.) Under the IC model, the probabilistic process is particularly simple and intuitive. When a node u becomes active in step t, it attempts to activate all currently inactive neighbors in step $t+1$. For each neighbor v, it succeeds with a known probability $p_{u,v}$. If it succeeds, v becomes active; otherwise, v remains inactive. Once u has made all these attempts, it does not get to make further activation attempts at later times. It was shown in [13] (see Lemma 3) that the set of nodes active at the end can be characterized alternatively as follows: for each ordered pair (u, v) independently, insert the directed edge (u, v) with probability $p_{u,v}$. Then, the active nodes are exactly the ones reachable via directed paths from A_0.

1.2 Can Instability Occur?

Suppose that we have inferred all parameters $p_{u,v}$, but are concerned that they may be slightly off: in reality, the influence probabilities are $p'_{u,v} \approx p_{u,v}$. Are there instances in which a seed set A_0 that is very influential with respect to the $p_{u,v}$ may be much less influential with respect to the $p'_{u,v}$? It is natural to suspect that this might not occur: when the objective function σ varies sufficiently smoothly with the input parameters (e.g., for linear objectives), small changes in the parameters only lead to small changes in the objective value; therefore, optimizing with respect to a perturbed input still leads to a near-optimal solution.

However, the objective σ of Influence Maximization does not depend on the parameters in a smooth way. To illustrate the issues at play, consider the following instance of the IC model. The social network consists of two disjoint bidirected cliques K_n, and $p_{u,v} = \hat{p}$ for all u, v in the same clique; in other words, for each directed edge, the same activation probability $\hat{p}$ is observed. The algorithm gets to select exactly $k = 1$ node. Notice that because all nodes look the same, any algorithm essentially chooses an arbitrary node, which may as well be from Clique 1.

Let $\hat{p} = 1/n$ be the sharp threshold for the emergence of a giant component in the Erdős-Rényi Random Graph $G(n, p)$. It is well known [4, 9] that the largest connected component of $G(n, p)$ has size $O(\log n)$ for any $p \leq \hat{p} - \Omega(1/n)$, and size $\Omega(n)$ for any $p \geq \hat{p} + \Omega(1/n)$. Thus, if unbeknownst to the algorithm, all true activation probabilities in Clique 1 are $p \leq \hat{p} - \Omega(1/n)$, while all true activation probabilities in Clique 2 are $p \geq \hat{p} + \Omega(1/n)$, the algorithm only activates $O(\log n)$ nodes in expectation, while it could have reached $\Omega(n)$ nodes by choosing Clique 2. Hence, small adversarial perturbations to the input parameters can lead to highly suboptimal solutions from any algorithm.[1]

1.3 Diagnosing Instability

The example of two cliques shows that there exist *unstable instances*, in which an optimal solution to the observed parameters is highly suboptimal when the observed parameters are slightly perturbed compared to the true parameters. Of course, not every instance of Influence Maximization is unstable: for instance, when the probability $\hat{p}$ in the Two-Clique instance is bounded away from the critical threshold of $G(n, p)$, the objective function varies much more smoothly with $\hat{p}$. This motivates the following algorithmic question, which is the main focus of our paper: *Given an instance of Influence Maximization, can we diagnose efficiently whether it is stable or unstable?*

To make this question precise, we formulate a model of perturbations. We assume that for each edge (u, v), in addition to the observed activation probability $p_{u,v}$, we are given an interval $I_{u,v} \ni p_{u,v}$ of values that the *actual probability* $p'_{u,v}$ could assume. The *true values* $p'_{u,v}$ are chosen from the intervals $I_{u,v}$ by an adversary; they induce an objective function σ' which the algorithm would like to maximize, while the observed values induce a different objective function σ which the algorithm actually has access to.

An instance $(p_{u,v}, I_{u,v})_{u,v}$ is *stable* if $|\sigma(S) - \sigma'(S)|$ is small for all objective functions σ' induced by legal probability settings, and for all seed sets S of size k. Here, "small" is defined relative to the objective function value $\sigma(A_0^*)$ of the optimum set.

When $|\sigma(S) - \sigma'(S)|$ is small compared to $\sigma(A_0^*)$ for all sets S, we will show in Section 3 that a user can have confidence that his optimization result will provide decent performance guarantees even if his input was perturbed. The converse is of course not necessarily true: even in unstable instances, a solution that was optimal for the observed input *may* still be very good for the true input parameters.

[1]The example reveals a close connection between the stability of an IC instance and the question whether a uniform activation probability p lies close to the edge percolation threshold of the underlying graph. Characterizing the percolation threshold of families of graphs has been a notoriously hard problem. Successful characterizations have only been obtained for very few specific classes (such as d-dimensional grids [15] and d-regular expander graphs [2]). Therefore, it is unlikely that a clean characterization of stable and unstable instances can be obtained. The connection to percolation also reveals that the instability was not an artifact of having high node degrees. By the result of Alon et al. [2], the same behavior will be obtained if both components are d-regular expander graphs, since such graphs also have a sharp percolation threshold.

1.4 Influence Difference Maximization

Trying to determine whether there is a function σ' and a set S for which $|\sigma(S) - \sigma'(S)|$ is large motivates the following optimization problem: Maximize $|\sigma(S) - \sigma'(S)|$ over all feasible functions σ' and all sets S. For any given set S, the objective is maximized either by making all probabilities (and thus $\sigma'(S)$) as small as possible, or by making all probabilities (and thus $\sigma'(S)$) as large as possible.[2] We denote the resulting two objective functions by σ^- and σ^+, respectively. The following definition then captures the optimization goal.

DEFINITION 1 (INFLUENCE DIFFERENCE MAXIMIZATION). *Given two instances with probabilities $p_{u,v} \geq p'_{u,v}$ for all u, v, let σ and σ' be their respective influence functions. Find a set S of size k maximizing $\delta(S) := \sigma(S) - \sigma'(S)$.*

The Influence Difference Maximization problem subsumes the Influence Maximization problem, by setting $p'_{u,v} \equiv 0$ (and thus also $\sigma' \equiv 0$). Notice that δ is generally not monotone, as $\delta(\emptyset) = \delta(V) = 0$, while $\delta(S) > 0$ for some sets S. Our main theoretical result is that while δ is not generally monotone, it is *submodular*.[3]

THEOREM 1. *Under the IC (and also LT) model, the influence difference objective function $\delta(S)$ is a non-negative and submodular function of the set S, whenever the functions σ, σ' satisfy the conditions of Definition 1.*

The proof of Theorem 1 is given in Section 4. It lets us leverage known efficient algorithms for non-monotone submodular function maximization subject to a cardinality constraint [6, 22]. In particular, we rely on the recent simple Random Greedy approximation algorithm of Buchbinder et al. [6], which guarantees an approximation factor of at least 0.266, but gets a guarantee close to $1/e$ when $k \ll n$.

If we use the Random Greedy algorithm and determine that the maximum influence difference is significantly smaller than the objective value of Influence Maximization, we can be confident that the solution determined by the Influence Maximization algorithm is a constant-factor approximation. Otherwise, the solution may still be good; however, the algorithm cannot provide a guarantee, as the deviation from parameter misestimates may drown out the objective value.

1.5 Experiments

Next, we investigate how pervasive instabilities are in real data. We evaluate frequently used synthetic models (2D grids, random regular graphs, small-world networks, and preferential attachment graphs) and real-world data sets (computer science theory collaborations and retweets about the Haiti earthquake). We focus on the IC model, and vary the influence strengths over a broad range of commonly studied values. We consider different relative perturbation levels Δ, ranging from 1% to 50%. The adversary can thus choose the actual activation probability to lie in the interval $[p_{u,v}(1-\Delta) \cdot p_{u,v}, p_{u,v} + \Delta \cdot p_{u,v}]$.

Our experiments suggest that perturbations can have significantly different effects depending on the network structure and observed values. As a general rule of thumb, unless the parameters are carefully chosen to lie right at the percolation threshold (in which case even tiny perturbations can have huge effects), perturbations with relative errors below 10% tend to not have huge impacts, while perturbations above 20% could significantly distort the optimum solution.

Since errors above 20% should be considered quite common for estimated social network parameters, our results suggest that practitioners exercise care in evaluating the stability of their problem instances, and treat the output of Influence Maximization algorithms with a healthy dose of skepticism.

1.6 Adversarial vs. Random Perturbations

One may question why we choose to study adversarial instead of random perturbations. This choice is for three reasons:

Theoretical: Worst-case analysis provides stronger guarantees, as it is not based on particular assumptions about the distribution of noise.

Practical: Most random noise models assume independence of noise across edges. However, we believe that in practice, both the techniques used for inferring model parameters as well as the data sources they are based on may well exhibit systematic bias, i.e., the noise will not be independent. For instance, a particular subpopulation may systematically underreport the extent to which they seek others' advice, or may have fewer visible indicators (such as posts) revealing their behavior.

Modeling Interest: Perhaps most importantly, most natural random noise models do not add anything to the IC and LT models. As an illustration, consider the random noise models studied in recent work by Goyal, Bonchi and Lakshmanan [12] and Adiga et al. [1]. Goyal et al. assume that for each edge (u, v), the value of $p_{u,v}$ is perturbed with uniformly random noise from a known interval. Adiga et al. assume that each edge (u, v) that was observed to be present is actually absent with some probability ϵ, while each edge that was not observed is actually present with probability ϵ; in other words, each edge's presence is independently flipped with probability ϵ.

The standard IC model subsumes both models straightforwardly. Suppose that a decision is to be made about whether u activates v. In the model of Goyal et al., we can first draw the actual (perturbed) value of $p'_{u,v}$ from its known distribution; subsequently, u activates v with probability $p'_{u,v}$; in total, u activates v with probability $\mathbb{E}\left[p'_{u,v}\right]$. Thus, we obtain an instance of the IC model in which all edge probabilities $p_{u,v}$ are replaced by $\mathbb{E}\left[p'_{u,v}\right]$. In the special case when the noise has mean 0, this expectation is exactly equal to $p_{u,v}$, which explains why Goyal et al. observed the noise to not affect the outcome at all.

In the model of Adiga et al., we first determine whether the edge is actually present; when it was observed present, this happens with probability $1 - \epsilon$; otherwise

[2] This observation relies crucially on the fact that each $p_{u,v}$ can independently take on any value in $I_{u,v}$. If the adversary were constrained by the total absolute deviation or sum of squares of deviations of parameters, this would no longer be the case. This issue is discussed in Section 6.

[3] Recall that a function f is submodular if it has "diminishing returns:" $f(S+x) - f(S) \geq f(T+x) - f(T)$ for any element x whenever $S \subseteq T$.

with probability ϵ. Subsequently, the activation succeeds with probability p. ([1] assumed uniform probabilities). Thus, the model is an instance of the IC model in which the activation probabilities on all observed edges are $p(1-\epsilon)$, while those on unobserved edges are $p\epsilon$. This reduction explains the theoretical results obtained by Adiga et al.

More fundamentally, practically all "natural" random processes that independently affect edges of the graph can be "absorbed into" the activation probabilities themselves; as a result, random noise does not at all play the result of actual noise.

2. MODELS AND PRELIMINARIES

The social network is modeled by a directed graph $G = (V, E)$ on n nodes. All parameters for non-existing edges are assumed to be 0. We first describe models of influence diffusion, and then models of parameter perturbation.

2.1 Influence Diffusion Models

Most of the models for Influence Maximization have been based on the Independent Cascade Model (see Section 1.1) and Linear Threshold Model studied in [13] and their generalizations. Like the Independent Cascade Model, the Linear Threshold Model also proceeds in discrete rounds. Each edge (u, v) is equipped with a weight $c_{u,v} \in [0, 1]$, satisfying $\sum_{u \to v} c_{u,v} \leq 1$ for all nodes v. (By $u \to v$, we denote that there is a directed edge (u, v).) Each node v initially draws a threshold ψ_v independently and uniformly at random from $[0, 1]$. A set A_0 of nodes is activated at time 0, and we use A_t to denote the set of nodes active at time t. In each discrete round t, each node v checks if $\sum_{u \in A_{t-1}, u \to v} c_{u,v} \geq \psi_v$. If so, v becomes active at time t, and remains active subsequently.

Any instance of the Influence Maximization problem is characterized by its parameters. For the LT model, the parameters are the n^2 edge weights $c_{u,v}$ for all edges (u, v). Similarly, for the IC model, the parameters are the edge activation probabilities $p_{u,v}$ for all edges (u, v). To unify notation, we write $\boldsymbol{\theta} = (\theta_{u,v})_{(u,v) \in E}$ for the vector of all parameter values, where $\theta_{u,v}$ could be either $c_{u,v}$ or $p_{u,v}$.[4]

Both the IC and LT model define random processes that continue until the diffusion process quiesces, i.e., no new activations occur. Let $\tau \leq n$ be the (random) time at which this happens. We denote the stochastic process by $\boldsymbol{P}_{\boldsymbol{\theta}}^{\text{Mod}}(A_0) = (A_t)_{t=0}^{\tau}$, with Mod $\in \{\text{IC}, \text{LT}\}$ denoting the model. The final set of active nodes is A_τ. We can now formally define the Influence Maximization problem:

DEFINITION 2 (INFLUENCE MAXIMIZATION). *The Influence Maximization problem consists of maximizing the objective $\sigma(A_0) := \mathbb{E}[|A_\tau|]$ (i.e., the expected number of active nodes in the end[5]), subject to a cardinality constraint $|A_0| \leq k$.*

[4]Some models besides IC and LT have more or fewer or different parameters. We will briefly see one such example in Section 4.2, but prefer to avoid unnecessarily general notation for now.

[5]Our results carry over unchanged if we assign each node a non-negative value r_v, and the goal is to maximize $\sum_{v \in A_\tau} r_v$. We focus on the case of uniform values for notational convenience only.

The key insight behind most prior work on algorithmic Influence Maximization is that the objective function $\sigma(S)$ is a monotone and submodular function of S. This was proved for the IC and LT models in [13], and subsequently for a generalization called Generalized Threshold Model (proposed in [13]) by Mossel and Roch [17].

2.2 Models for Perturbations

To model adversarial input perturbations, we assume that for each of the edges (u, v), we are given an interval $I_{u,v} = [\ell_{u,v}, r_{u,v}] \subseteq [0, 1]$ with $\theta_{u,v} \in I_{u,v}$. For the LT model, to ensure that the resulting activations functions are always submodular, we require that $\sum_{u \to v} r_{u,v} \leq 1$ for all nodes v. We write $\boldsymbol{\Theta} = \times_{(u,v) \in E} I_{u,v}$ for the set of all allowable parameter settings. The adversary must guarantee that the ground truth parameter values satisfy $\boldsymbol{\theta'} \in \boldsymbol{\Theta}$; subject to this requirement, the adversary can choose the actual parameter values arbitrarily.

Together, the parameter values $\boldsymbol{\theta}$ determine an instance of the Influence Maximization problem. We will usually be explicit about indicating the dependence of the objective function on the parameter setting. We write $\sigma_{\boldsymbol{\theta}}$ for the objective function obtained with parameter values $\boldsymbol{\theta}$, and only omit the parameters when they are clear from the context. For a given setting of parameters, we will denote by $A^*_{\boldsymbol{\theta}} \in \text{argmax}_S \, \sigma_{\boldsymbol{\theta}}(S)$ a solution maximizing the expected influence under parameter values $\boldsymbol{\theta}$.

2.3 Influence Difference Maximization

In order to capture to what extent adversarial changes in the parameters can lead to misestimates of any set's influence, we are interested in the quantity

$$\max_S \max_{\boldsymbol{\theta'} \in \boldsymbol{\Theta}} |\sigma_{\boldsymbol{\theta}}(S) - \sigma_{\boldsymbol{\theta'}}(S)|, \tag{1}$$

where $\boldsymbol{\theta}$ denotes the observed parameter values. For two parameter settings $\boldsymbol{\theta}, \boldsymbol{\theta'}$ with $\boldsymbol{\theta} \geq \boldsymbol{\theta'}$ coordinate-wise, it is not difficult to show using a simple coupling argument (see Lemma 6) that $\sigma_{\boldsymbol{\theta}}(S) \geq \sigma_{\boldsymbol{\theta'}}(S)$ for all S. Therefore, for any fixed set S, the maximum is attained either by making $\boldsymbol{\theta'}$ as large as possible or as small as possible. Hence, solving the following problem is sufficient to maximize (1).

DEFINITION 3. *Given an influence model and two parameter settings $\boldsymbol{\theta}, \boldsymbol{\theta'}$ with $\boldsymbol{\theta} \geq \boldsymbol{\theta'}$ coordinate-wise, define*

$$\delta_{\boldsymbol{\theta}, \boldsymbol{\theta'}}(S) \;=\; \sigma_{\boldsymbol{\theta}}(S) - \sigma_{\boldsymbol{\theta'}}(S). \tag{2}$$

Given the set size k, the Influence Difference Maximization (IDM) problem is defined as follows:

$$\begin{array}{ll} \textit{Maximize} & \delta_{\boldsymbol{\theta}, \boldsymbol{\theta'}}(S) \\ \textit{subject to} & |S| = k. \end{array} \tag{3}$$

3. MAIN THEOREM AND APPLICATIONS

Maximizing $\delta_{\boldsymbol{\theta}, \boldsymbol{\theta'}}(S)$ (approximately) lets us evaluate how susceptible an instance is to noise. Hence, we would like to exhibit useful properties of the function $\delta_{\boldsymbol{\theta}, \boldsymbol{\theta'}}$ making it amenable to optimization. This is accomplished by Theorem 1, which we restate here using the notation from Section 2.

THEOREM 1. *Under the IC and LT models, the function $\delta_{\boldsymbol{\theta}, \boldsymbol{\theta'}}(S)$ is a non-negative submodular function of S whenever $\boldsymbol{\theta} \geq \boldsymbol{\theta'}$ pointwise.*

As we show in Section 3.2, Theorem 1 implies that we can leverage known results on the maximization of submodular functions (see [6, 22] and references therein) in order to approximately maximize $\delta_{\boldsymbol{\theta},\boldsymbol{\theta}'}(S)$, and thus gain an understanding of how large the changes in objective function values are. The proof of Theorem 1 is presented in Section 4.

3.1 Approximation for Perturbed Objectives

We have several times alluded to the idea that if the maximum influence difference is "small," the seed set found by an Influence Maximization algorithm for the *observed* parameters will have useful guarantees even for the *unobserved* actual parameters. We now make the notion of "small" precise and derive a guarantee on the performance vis-à-vis unobserved actual parameters. We consider four different parameter settings and their associated objective functions:

- $\boldsymbol{\theta}$ is the observed/inferred vector of parameters, and $\sigma = \sigma_{\boldsymbol{\theta}}$ is the resulting objective function. Let A^* be a seed set maximizing $\sigma(S)$ subject to $|S| = k$, and A the seed set returned by the greedy algorithm for Influence Maximization. Thus, A satisfies
$$\sigma(A) \;\geq\; (1 - 1/e) \cdot \sigma(A^*). \tag{4}$$
- $\hat{\boldsymbol{\theta}}$ is the vector of ground truth parameters chosen by the adversary, and $\hat{\sigma} = \sigma_{\hat{\boldsymbol{\theta}}}$ the associated objective function. $\hat{A}$ denotes the seed set maximizing $\hat{\sigma}(S)$.
- $\boldsymbol{\theta}^+$ is the vector making each parameter as large as possible, and $\sigma^+ = \sigma_{\boldsymbol{\theta}^+}$ the associated objective function.
- $\boldsymbol{\theta}^-$ is the vector making each parameter as small as possible, and $\sigma^- = \sigma_{\boldsymbol{\theta}^-}$ the associated objective function.

Since the observed objective value $\sigma(A)$ is the only "scale" that an algorithm has available for measuring the impact of perturbations, we relate all deviations to this quantity. Specifically, assume that running the greedy Influence Difference Maximization algorithm has revealed the following[6] for all sets S with $|S| = k$:
$$\delta_{\boldsymbol{\theta}^+,\boldsymbol{\theta}}(S) \;\leq\; \alpha^+ \cdot \sigma(A). \tag{5}$$
In addition to the Influence Difference Maximization estimate, assume that explicitly evaluating the difference for the set A has revealed that
$$\delta_{\boldsymbol{\theta},\boldsymbol{\theta}^-}(A) \;=\; \alpha^- \cdot \sigma(A). \tag{6}$$
Our goal is now to lower-bound $\hat{\sigma}(A)$ in terms of $\hat{\sigma}(\hat{A})$. We begin by lower-bounding $\sigma(A^*)$:
$$\begin{aligned}
\sigma(A^*) &= \Big(\sigma(A^*) - \sigma(\hat{A})\Big) + \Big(\sigma(\hat{A}) - \hat{\sigma}(\hat{A})\Big) + \hat{\sigma}(\hat{A}) \\
&\geq \Big(\sigma(\hat{A}) - \sigma^+(\hat{A})\Big) + \hat{\sigma}(\hat{A}) \\
&\overset{(5)}{\geq} -\alpha^+ \cdot \sigma(A) + \hat{\sigma}(\hat{A}).
\end{aligned}$$

[6]Notice that to reveal such a bound, the algorithm's observed maximum function value $\delta_{\boldsymbol{\theta}^+,\boldsymbol{\theta}}(S)$ must be smaller than $\sigma(A)$ by a constant factor (roughly e), to compensate for the approximation factor of the Influence Difference Maximization algorithm.

For the first inequality, we used that the first term is nonnegative by the optimality of A^* for σ, and that $\sigma^+(S) \geq \hat{\sigma}(S)$ for all sets S. We can now combine Inequality (4) with the previous derivation and solve for $\sigma(A)$:
$$\sigma(A) \;\geq\; \frac{1 - 1/e}{1 + \alpha^+ \cdot (1 - 1/e)} \cdot \hat{\sigma}(\hat{A}).$$
Finally, we use this inequality to bound $\hat{\sigma}(A)$ as follows:
$$\begin{aligned}
\hat{\sigma}(A) &= (\hat{\sigma}(A) - \sigma(A)) + \sigma(A) \qquad (7)\\
&\geq \big(\sigma^-(A) - \sigma(A)\big) + \sigma(A) \\
&\overset{(6)}{=} (1 - \alpha^-) \cdot \sigma(A) \\
&\geq \frac{(1 - \alpha^-) \cdot (1 - 1/e)}{1 + \alpha^+ \cdot (1 - 1/e)} \cdot \hat{\sigma}(\hat{A}), \qquad (8)
\end{aligned}$$
where the first inequality used that $\hat{\sigma}(S) \geq \sigma^-(S)$ for all sets S.

Thus, whenever α^-, α^+ are small enough, we can guarantee that the calculated seed set A is within a constant factor of the best seed set for the unknown ground truth influence parameters.

3.2 Maximizing Submodular Functions

While the maximization of monotone submodular functions is essentially a solved problem (the widely known and used $1 - 1/e$ approximation algorithm due to Nemhauser et al. [19] is best possible unless P=NP), approximation algorithms for the maximization of *non-monotone* submodular functions subject to a cardinality constraint have only more recently received significant attention. (See [6, 22] and references therein.)

In our work, we use the *Random Greedy* algorithm of [6], given as Algorithm 1 below. It is a natural generalization of the simple greedy algorithm widely used for maximizing monotone submodular functions.[7] Instead of picking the *best* single element to add in each iteration, it first finds the set of the k individually best single elements (i.e., the elements which when added to the current set give the largest, second-largest, third-largest, ..., k^{th}-largest gain). Then, it picks one of these k elements uniformly at random and continues.

The approximation guarantee for Random Greedy is captured by Theorem 2 below. The current best approximation guarantee — a factor 0.356 (slightly less than $1/e$) — is achieved by running Random Greedy and another algorithm called "Continuous Double Greedy" and keeping the better of the two solutions. While taking the better of the two algorithms gives a better approximation guarantee, in our experiments, we prefer to just use the Random Greedy algorithm due to its simplicity and efficiency. Notice that when $k \ll n$, the approximation guarantee of Random Greedy is very close to $1/e$; since $k \ll n$ for all of our experiments, we treat the approximation guarantee as $1/e$ in our discussion for simplicity.

THEOREM 2 (BUCHBINDER ET AL. [6]). *Let g be a non-negative submodular (not necessarily monotone) function. Consider the problem of maximizing $g(S)$ subject to the constraint that $|S| = k$; let S_k^* be the optimum set of size k.*

[7]Buchbinder et al. [6] show that for non-monotone submodular functions, the traditional greedy algorithm can perform arbitrarily poorly.

1. *The set S_k returned by the Random Greedy Algorithm guarantees $\mathbb{E}[g(S_k)] \geq \max(0.266, \frac{1}{e} \cdot (1 - \frac{k}{en})) \cdot g(S_k^*)$, where the expectation is taken over the random choices of the algorithm.*

2. *By taking the better of the outputs of the Random Greedy Algorithm and the "Continuous Double Greedy Algorithm," the resulting set $\hat{S}$ guarantees that $\mathbb{E}\left[g(\hat{S})\right] \geq 0.356 \cdot g(S_k^*)$.*

Algorithm 1 Random Greedy Algorithm

```
Initialize: S_0 ← ∅
for i = 1, ..., k do
    Let M_i ⊆ V \ S_{i-1} be the subset of size k maximizing Σ_{u∈M_i} g(S_{i-1} ∪ {u}) − g(S_{i-1}).
    Draw u_i uniformly at random from M_i.
    Let S_i ← S_{i-1} ∪ {u_i}.
end for
Return S_k
```

The running time of the Random Greedy Algorithm is $O(kC|V|)$, where C is the time required to estimate $g(S \cup \{u\}) - g(S)$. In our case, the objective function is #P-hard to evaluate exactly [23, 8], but arbitrarily close approximations can be obtained by Monte Carlo simulation. Since each simulation takes time $O(|V|)$, if we run M iterations of the Monte Carlo simulation in each iteration, the overall running time of the algorithm is $O(kM|V|^2)$. In order to accelerate the algorithm in practice, we use the CELF optimization proposed by Leskovec et al. [16]. It exploits submodularity to avoid computation of $g(S_{i-1} \cup \{u\}) - g(S_{i-1})$ whenever the value of $g(S_{i-2} \cup \{u\}) - g(S_{i-2})$ is sufficiently small that u cannot be the optimal element to add in iteration i.

4. PROOF OF THEOREM 1

We exploit and extend the *Triggering Set* technique from [13]. The idea of the Triggering Set technique is to characterize the outcome A_τ of the dynamic and stochastic activation process $\boldsymbol{P}_{\boldsymbol{\theta}}^{\text{Mod}}(A_0)$ by a more static process called a Triggering Model: Generate a random graph G according to a particular distribution, and let $R_G(A_0)$ be the set of nodes reachable from A_0 in G. ([13] calls the edges in G *live* and the edges missing from G *blocked*.) The Triggering Set technique then consists in showing that for each of the models $\{\text{IC}, \text{LT}\}$, one can identify a Triggering Model (i.e., a distribution over random graphs G) such that the random set $R_G(A_0)$ has precisely the same distribution (under the random graph G) as the random set A_τ under the random process $\boldsymbol{P}_{\boldsymbol{\theta}}^{\text{Mod}}(A_0)$. More specifically, [13] proves the following two lemmas:

LEMMA 3 (CLAIM 2.3 FROM [13]). *Given an instance of the IC model with edge probabilities $\boldsymbol{\theta} = (p_{u,v})$, let $\mathcal{G}_{\boldsymbol{\theta}}^{IC}$ be the random graph model in which each directed edge (u, v) is present (or live) independently with probability $p_{u,v}$, and blocked otherwise. Then, for all sets A_0, S,*

$$\text{Prob}_{G \sim \mathcal{G}_{\boldsymbol{\theta}}^{IC}}[R_G(A_0) = S] \;=\; \text{Prob}_{\boldsymbol{P}_{\boldsymbol{\theta}}^{IC}(A_0)}[A_\tau = S].$$

LEMMA 4 (CLAIM 2.6 FROM [13]). *Given an instance of the LT model with edge weights $\boldsymbol{\theta} = (c_{u,v})$, let $\mathcal{G}_{\boldsymbol{\theta}}^{LT}$ be the random graph model in which each node v picks the incoming edge (u, v) with probability $c_{u,v}$; the presence of the edges is mutually exclusive, so that each node v has at most one incoming live edge, and with probability $1 - \sum_u c_{u,v}$ has no incoming live edge. Then, for all sets A_0, S,*

$$\text{Prob}_{G \sim \mathcal{G}_{\boldsymbol{\theta}}^{LT}}[R_G(A_0) = S] \;=\; \text{Prob}_{\boldsymbol{P}_{\boldsymbol{\theta}}^{LT}(A_0)}[A_\tau = S].$$

These lemmas imply that instead of studying a dynamic process, we can randomly generate graphs, and then study reachability on those graphs. When we consider two parameter settings $\boldsymbol{\theta} \geq \boldsymbol{\theta'}$ (pointwise), it seems intuitive that we should be able to generate two corresponding graphs G, G' (on the same vertex set V) according to the respective distributions, such that $E(G) \supseteq E(G')$ *always holds*. More formally, below, we will show the following coupling lemma:

LEMMA 5 (COUPLING OF RANDOM GRAPHS). *For any instances $\boldsymbol{\theta}, \boldsymbol{\theta'}$ of the IC Model (or LT Model) with $\boldsymbol{\theta} \geq \boldsymbol{\theta'}$, there is a distribution D over pairs (G, G') of graphs on the same vertex set V, with the following properties:*

1. *$E(G) \supseteq E(G')$ always holds.*

2. *$\sum_{G'} \text{Prob}_D[(G, G')] = \text{Prob}_{G \sim \mathcal{G}_{\boldsymbol{\theta}}^{Mod}}[G]$. That is, the first graph in isolation follows the random graph distribution $\mathcal{G}_{\boldsymbol{\theta}}^{Mod}$ (where Mod could be IC or LT).*

3. *$\sum_{G} \text{Prob}_D[(G, G')] = \text{Prob}_{G' \sim \mathcal{G}_{\boldsymbol{\theta'}}^{Mod}}[G']$. That is, the second graph in isolation follows the random graph distribution $\mathcal{G}_{\boldsymbol{\theta'}}^{Mod}$ (where again Mod could be IC or LT).*

Finally, we analyze "reachability difference" between two arbitrary graphs with $E(G) \supseteq E(G')$.

LEMMA 6 (COMPARISON OF REACHABILITY). *Let G, G' be two fixed graphs with $E(G) \supseteq E(G')$. Then, $R_G(S) \supseteq R_{G'}(S)$.*

Furthermore, let $\rho(S) := |R_G(S) \setminus R_{G'}(S)|$ be the number of nodes reachable from S in G, but not in G'. $\rho(S)$ is a submodular function of S.

To complete the proof, we can decompose the influence difference objective into a distribution over individual graphs as follows:

$$\begin{aligned}
\delta_{\boldsymbol{\theta}, \boldsymbol{\theta'}}(A_0) &= \mathbb{E}_{G \sim \mathcal{G}_{\boldsymbol{\theta}}^{\text{Mod}}}[|R_G(A_0)|] - \mathbb{E}_{G' \sim \mathcal{G}_{\boldsymbol{\theta'}}^{\text{Mod}}}[|R_{G'}(A_0)|] \\
&= \mathbb{E}_{(G,G') \sim D}[|R_G(A_0)| - |R_{G'}(A_0)|] \\
&= \mathbb{E}_{(G,G') \sim D}[|R_G(A_0) \setminus R_{G'}(A_0)|] \\
&= \sum_{G, G'} \text{Prob}_{(G,G') \sim D}[(G, G')] \cdot |R_G(A_0) \setminus R_{G'}(A_0)| \\
&= \sum_{G, G'} \text{Prob}_{(G,G') \sim D}[(G, G')] \cdot \rho(A_0).
\end{aligned}$$

The second equality used that D defines a valid coupling over pairs of graphs; the third equality used that $E(G) \supseteq E(G')$, and the fourth equality used the definition of the expectation.

Notice that the final form is a non-negative linear combination of submodular and non-negative functions, which is known to be submodular. This completes the proof. ∎

4.1 Proof of Lemmas

Proof of Lemma 6. Fix two graphs G, G' on the same vertex set V, with $E(G) \supseteq E(G')$. $R_G(S) \supseteq R_{G'}(S)$ is obvious, so we focus on the second part of the lemma. To show that $\rho(S)$ is submodular, we will show a slightly stronger statement. Let $\Delta R(S) := R_G(S) \setminus R_{G'}(S)$ be the actual *set* of nodes reachable from S in G, but not in G'. We will show that for all sets $S \subseteq T$ and nodes v, we have $\Delta R(T+v) \setminus \Delta R(T) \subseteq \Delta R(S+v) \setminus \Delta R(S)$. This directly implies submodularity of $\rho(S) = |\Delta R(S)|$.

By definition, a node u is in $\Delta R(S)$ if and only if there exists a path from S to u in G, but not in G'. Thus, a node u is in $\Delta R(T+v) \setminus \Delta R(T)$ if and only if there is (1) a path from $T+v$ to u in G, (2) no path from $T+v$ to u in G', and (3) no path from T to u in G.

By the first and third condition, there must be a path from v to u in G (since the addition of the source node v is the only difference between those conditions). Hence, there is a path from $S+v$ to u in G. Second, because there is no path from $T+v$ to u in G', and $S \subseteq T$, there cannot be a path from $S+v$ to u in G'. Third, because there is no path from T to u in G, and $S \subseteq T$, there cannot be a path from S to u in G. Hence, by definition, u is in $\Delta R(S+v) \setminus \Delta R(S)$. In other words, $u \in \Delta R(T+v) \setminus \Delta R(T)$ implies that $u \in \Delta R(S+v) \setminus \Delta R(S)$, completing the proof. ∎

Proof of Lemma 5. We begin by proving the lemma for the IC model, for which the coupling is simpler. We need to decide, for each edge (u,v) independently, whether to include (u,v) in G and G'. To do so, we generate, independently for each (u,v), a number $x_{u,v}$ uniformly from $[0,1]$. The edge (u,v) is included in G iff $x_{u,v} \leq p_{u,v}$; similarly, (u,v) is included in G' iff $x_{u,v} \leq p'_{u,v}$. Thus, the inclusion probabilities are correct. Furthermore, because $p_{u,v} \geq p'_{u,v}$ by assumption, whenever (u,v) is included in G', it is also included in G. Thus, the proposed generative process is a valid coupling for the IC model.

Next, we exhibit a valid coupling for the LT model. We need to decide, for each node v independently, which single incoming edge (u,v) (if any) to include in G and G'. We use the following generative process. For each node v, we consider the interval $[0,1]$. For every edge (u,v) into v, we place an interval J_u of length $c_{u,v}$ into the $[0,1]$ interval, such that the intervals $J_u, J_{u'}$ for $u \neq u'$ are disjoint. This is possible because $\sum_u c_{u,v} \leq 1$. Then, for each node u, we pick a subinterval $J'_u \subseteq J_u$ of length $c'_{u,v}$.

Finally, a random number x_v is drawn uniformly at random from $[0,1]$. If x_v lies in J_u, then (u,v) is inserted into G, and if it lies in J'_u, then (u,v) is inserted into G'. If x_v does not lie in any of the J_u, then v has no incoming edge in G; similarly for G'. Notice that the incoming edge in both graphs is chosen with the correct distribution, proving that we have a valid coupling. Furthermore, because $J'_u \subseteq J_u$, whenever the edge (u,v) is included in G', it is also included in G, so that $E(G) \supseteq E(G')$, as desired. ∎

4.2 A more general view

The Triggering Set technique we used in the proof of Theorem 1 can be applied to prove a more general version of the theorem. It shows that $\delta_{\boldsymbol{\theta},\boldsymbol{\theta'}}(A_0)$ is submodular whenever the following two hold:

1. The diffusion model is equivalent to a Triggering Model.
2. Whenever $\boldsymbol{\theta} \geq \boldsymbol{\theta'}$ holds pointwise, the random graphs (G, G') with parameters $\boldsymbol{\theta}, \boldsymbol{\theta'}$ can be coupled such that $E(G) \supseteq E(G')$ always holds.

For example, [13] defined the "Only Listen Once" model: in it, a node v is activated with some probability p_v by the first neighbor trying to convince it; if the attempt is unsuccessful, all subsequent activation attempts deterministically fail. [13] showed that this model is equivalent to a Triggering Model in which a node has live incoming edges from all of its neighbors with probability p_v, and from none of its neighbors with probability $1 - p_v$. When $p_v \geq p'_v$ for all v, these decisions can be coupled like the presence of single edges in the IC model. Thus, Influence Difference Maximization is also submodular for the "Only Listen Once" model.

More generally, the conditions under which a coupling is possible are characterized by Strassen's Monotone Coupling Theorem [21]:

THEOREM 7 (STRASSEN [21]). *Let $(\mathcal{X}, \preceq)$ be a finite partially ordered set, and let μ, ν be probability distributions on $\mathcal{X}$. If μ stochastically dominates ν[8], then there exist (coupled) random variables $M \sim \mu, N \sim \nu$ such that $M \succeq N$ always holds.*

Strassen's Theorem can be applied with $\mathcal{X}$ being the set of all subsets of edges, ordered by the $\subseteq$ order. We obtain the following corollary:

COROLLARY 8. *Consider any Influence Maximization model equivalent to a Triggering model, and instances with parameters $\boldsymbol{\theta}, \boldsymbol{\theta'}$. The Influence Difference Maximization objective function $\delta_{\boldsymbol{\theta},\boldsymbol{\theta'}}(S)$ is submodular whenever $\mathcal{G}^{Mod}_{\boldsymbol{\theta}}$ stochastically dominates $\mathcal{G}^{Mod}_{\boldsymbol{\theta'}}$.*

5. EXPERIMENTS

While we saw in Section 1.2 that examples highly susceptible (with errors of magnitude $\Omega(n)$) to small perturbations exist, the goal of this section is to evaluate experimentally how widespread this behavior is for realistic social networks.

5.1 Experimental Setting

We carry out experiments under the IC model, for six classes of graphs — four synthetic and two real-world. In each case, the model/data give us a simple graph or multigraph. Multigraphs are converted to simple graphs by collapsing parallel edges to a single edge with weight c_e equal to the number of parallel edges; for simple graphs, all weights are $c_e = 1$. The observed probabilities for edges are $p_e = c_e \cdot p$; across experiments, we vary the base probability p to take on the values $\{0.01, 0.02, 0.05, 0.1\}$. The resulting parameter vector is denoted by $\boldsymbol{\theta}$.

The uncertainty interval for e is $I_e = [(1-\Delta)p_e, (1+\Delta)p_e]$; here, Δ is an uncertainty parameter for the estimation, which takes on the values $\{1\%, 5\%, 10\%, 20\%, 50\%\}$ in our experiments. The parameter vectors $\boldsymbol{\theta}^+$ and $\boldsymbol{\theta}^-$ describe the settings in which all parameters are as large (as small, respectively) as possible.

[8]We say that μ stochastically dominates ν if for every set $A \subseteq \mathcal{X}$ which is upward-closed (i.e., $x \in A$ and $y \succeq x$ implies $y \in A$), $\mu(A) \geq \nu(A)$.

5.2 Network Data

We run experiments on four synthetic networks and two real social networks. Synthetic networks provide a controlled environment in which to compare observed behavior to expectations, while real social networks may give us indications about the prevalence of vulnerability to perturbations in real networks that have been studied in the past.

Synthetic Networks. We generate synthetic networks according to four widely used network models. In all cases, we generate undirected networks with 400 nodes. The network models are: (1) the 2-dimensional grid, (2) random regular graphs, (3) the Watts-Strogatz Small-World (SW) Model [24] on a ring with each node connecting to the 5 closest nodes on each side initially, and a rewiring probability of 0.1. (4) The Barabási-Albert Preferential Attachment (PA) Model [3] with 5 outgoing edges per node. For all synthetic networks, we select $k = 20$ seed nodes.

Real Networks. We consider two real networks to evaluate the susceptibility of practical networks: one (*STOCFOCS*) is a co-authorship network of theoretical CS papers; the other (*Haiti*) is a Retweet network.

The co-authorship network, *STOCFOCS*, is a multigraph extracted from published papers in the conferences STOC and FOCS from 1964–2001. Each node in the network is a researcher with at least one publication in one of the conferences. For each multi-author paper, we add a complete undirected graph among the authors. As mentioned above, parallel edges are then compressed into a single edge with corresponding weight. The resulting graph has 1768 nodes and 10024 edges. Due to its larger size, we select 50 seed nodes.

The *Haiti* network is extracted from tweets of 274 users on the topic *Haiti Earthquake* in Twitter. For each tweet of user u that was retweeted by v, we add a directed edge (u, v). We obtain a directed multigraph; after contracting parallel edges, the directed graph has 383 weighted edges. For this network, due to its smaller size, we select 20 seeds.

In all experiments, we work with uniform edge weights p, since — apart from edge multiplicities — we have no evidence on the strength of connections. It is a promising direction for future in-depth experiments to use influence strengths inferred from real-world cascade datasets by network inference methods such as [10, 11, 18].

5.3 Results

In all our experiments, the results for the Grid and Small-World network are sufficiently similar that we omit the results for grids here. As a first sanity check, we empirically computed $\max_{S:|S|=1} \delta_{\boldsymbol{\theta}+,\boldsymbol{\theta}-}(S)$ for the *complete graph* on 200 nodes with $I_e = [1/200 \cdot (1-\Delta), 1/200 \cdot (1+\Delta)]$ and $k = 1$. According to the analysis in Section 1.2, we would expect extremely high instability. The results, shown in Table 1, confirm this expectation.

Δ	$\sigma_{\boldsymbol{\theta}+}$	$\sigma_{\boldsymbol{\theta}-}$
50%	66.529	1.955
20%	23.961	4.253
10%	15.071	6.204

Table 1: Instability for the clique K_{200}.

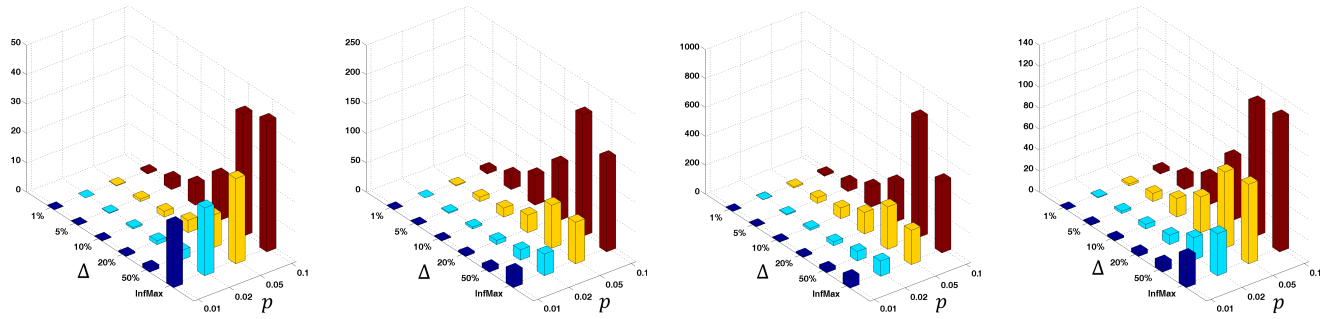
Next, Figure 1 shows the (approximately) computed values $\max_{S:|S|=k} \delta_{\boldsymbol{\theta}+,\boldsymbol{\theta}-}(S)$, and — for calibration purposes — $\max_{A_0:|A_0|=k} \sigma_{\boldsymbol{\theta}}(A_0)$ for all networks and parameter settings. Notice that the figure shows the *computed* values; in reality, $\max_S \delta_{\boldsymbol{\theta}+,\boldsymbol{\theta}-}(S)$ could be up to about a factor e larger than the displayed values due to the loss of an approximation factor.

While individual networks vary somewhat in their susceptibility, the overall trend is that larger estimates of baseline probabilities p make the instance more susceptible to noise, as do (obviously) larger uncertainty parameters Δ. In particular, for $\Delta \geq 20\%$, the noise (after scaling) dominates the Influence Maximization objective function value, meaning that optimization results should be used with care. On the other hand, when $\Delta \leq 10\%$, even the scaled-up misestimation is noticeably less than the Influence Maximization objective, so that reasonable approximation guarantees are obtained even for perturbed data.

Next, we compute what approximation guarantees could be given for the instances using Formula (8). Specifically, using $\boldsymbol{\theta}$ as the observed edge probabilities, we run the standard greedy algorithm for Influence Maximization to determine a set A_0. Based on this set and the lower bounds $\boldsymbol{\theta}^-$, the value α^- is directly computed. To compute α^+, we run the Random Greedy Influence Difference Maximization algorithm on $\delta_{\boldsymbol{\theta}+,\boldsymbol{\theta}}(S)$ and scale up the maximum value it finds by a factor e. We use the best solution of 10 independent runs of the Influence Difference Maximization algorithm. The objective function value is approximated using $M = 2000$ iterations of the Monte Carlo simulation. As discussed in Section 3.2, since $k \ll n$ in all our experiments, the approximation guarantee is very close to $1/e$, so that this scaling ensures that α^+ does not underestimate the effect of perturbations. Using all of these values, we can now estimate the approximation guarantee using Formula (8). The results are shown in Figure 2.

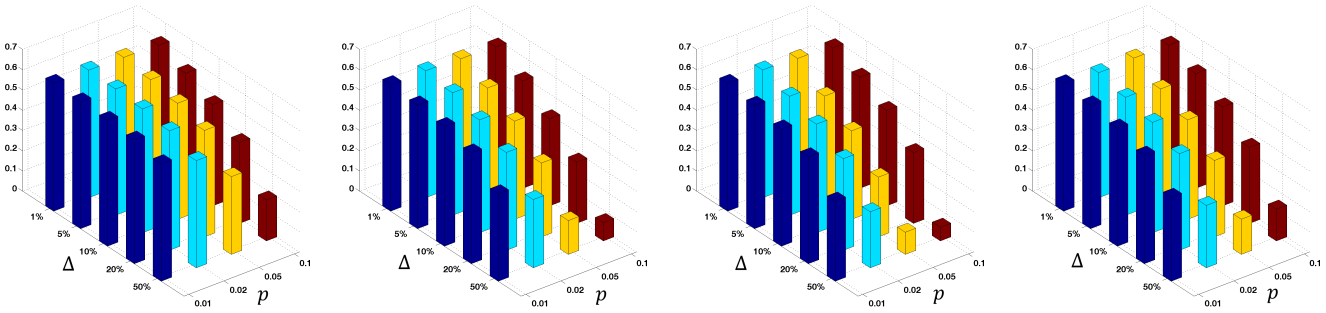
In interpreting Figure 2, notice that even with perfect data, one cannot exceed an approximation guarantee of $1 - 1/e \approx 0.63$. As indicated already by Figure 1, when the baseline probability is large and the error exceeds 20%, the approximation guarantee begins to deteriorate significantly. According to Formula (8), Influence Maximization can tolerate $\alpha^+ > 1$, which corresponds to significant perturbations that make parameter values *larger*. On the other hand, for deviations that make parameter values *smaller* (which would indicate that the solution A_0 may only *appear* good based on an overestimate of some parameters), a value of α^- close to 1 would have devastating effects. ($\alpha^- > 1$ is impossible by definition.) Fortunately, α^- is computed only for one specific set A_0, and thus not subject to the $1/e$ approximation factor for Influence Difference Maximization.

Next, we evaluate the dependence of the noise tolerance on the degrees of the graph, by experimenting with random d-regular graphs whose degrees vary from 5 to 25. It is known that such graphs are expanders with high probability, and hence have percolation thresholds of $1/d$ [2]. Accordingly, we set the base probability to $(1+\alpha)/d$ with $\alpha \in \{-40\%, -20\%, 0, 20\%, 40\%\}$. We use the same setting for uncertainty intervals as in the previous experiments. Figure 3 shows the ratio between Influence Difference Maximization and Influence Maximization, i.e., $\frac{\max_S \delta_{\boldsymbol{\theta}+,\boldsymbol{\theta}-}(S)}{\max_S \sigma_{\boldsymbol{\theta}}(S)}$, with $\alpha = 0$. (The results for $\alpha \neq 0$ are similar and omitted due to space constraints.) It indicates that for random regular graphs, the degree does not appear to significantly affect stability, and that again, noise around 20% begins to pose



(a) Small World (b) PA (c) STOCFOCS (d) Haiti

Figure 1: Comparison between Influence Difference Maximization and Influence Maximization results for four different networks.



(a) Small World (b) PA (c) STOCFOCS (d) Haiti

Figure 2: Approximation Guarantees according to the bound in (8), for four different networks.

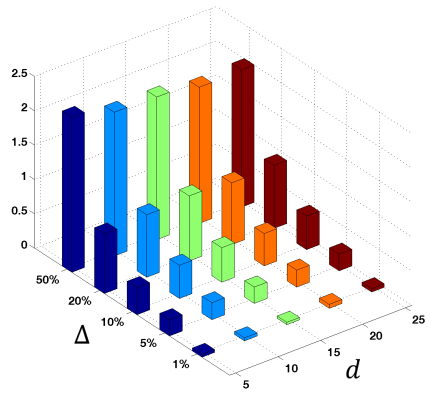


Figure 3: Ratio between Influence Difference Maximization and Influence Maximization under random regular graphs with different degree.

a significant challenge. Moreover, we observe that the approximation guarantee reaches its minimum when the edge activation probability is exactly at the percolation threshold $1/d$. This result is in line with percolation theory and also the analysis of Adiga et al. [1].

As a general takeaway message, decent approximation guarantees are obtained when the noise is below 10%, and in some cases even for noise around 20%. However, for larger amounts of noise — which may well occur in practice — a lot of caution is advised in using the results of algorithmic Influence Maximization.

6. DISCUSSION

We began a study of the stability of Influence Maximization when the input data are adversarially noisy. We showed that estimating the susceptibility of an instance to perturbations can be cast as an Influence Difference Maximization problem with a non-monotone submodular objective function. This let us leverage known algorithms for maximizing such functions to evaluate the susceptibility of real and synthetic data sets. In the examples we studied, typically, relative perturbations of less than 10% still lead to satisfactory solutions for Influence Maximization, while relative perturbations above 20% lead to significant risk of wrong outputs. Given the noise inherent in all estimates of social network data, this suggests applying extreme caution before relying heavily on results of algorithmic Influence Maximization.

Our work raises a number of interesting questions for follow-up work. First, while we prove submodularity of Influence Difference Maximization for the IC and LT models, it is an interesting question whether Influence Difference Maximization is submodular under more general models, such as the General Threshold Model [13, 17], and how general of an adversarial perturbation model can be tolerated. A path towards an extension to Triggering Models is suggested by Corollary 8. However, while a full characterization of models equivalent to Triggering Models was achieved by Salek et al. [20], it is not clear how to leverage such a characterization to determine what pairs of parameter vectors lead to stochastic domination.

Second, our adversarial model specified a valid range of parameters for each parameter of the model, and allowed the adversary to perturb the parameters arbitrarily within that range. Instead, we may posit an upper bound on the

total (or *total squared*) perturbation the adversary can apply; such a bound creates dependencies between the perturbations at different nodes, making it much harder to even determine which legal parameter setting would maximize $|\delta_{\theta,\hat{\theta}}(S)|$ for a given set S.

While we begin an investigation of how pervasive susceptibility to perturbations is in Influence Maximization data sets, our investigation is necessarily limited. Most importantly, ground truth data are by definition impossible to obtain, and even good and reliable inferred data sets of actual influence probabilities are currently not available. The values we assigned for our experimental evaluation cover a wide range of parameter values studied in past work, but the community does not appear to have answered the question whether these ranges actually correspond to reality.

At an even more fundamental level, the *models* themselves have received surprisingly little thorough experimental validation, despite having served as models of choice for hundreds of papers over the last decade. In addition to verifying the susceptibility of models to parameter perturbations, it is thus a pressing task to verify how susceptible the optimization problems are to incorrect models. The verification or falsification of sociological models for collective behavior likely falls outside the expertise of the computer science community, but nonetheless needs to be undertaken before any significant impact of work on Influence Maximization can be truthfully claimed.

Acknowledgments

We would like to thank Shaddin Dughmi for useful pointers and feedback, Shishir Bharathi and Mahyar Salek for useful discussions, and anonymous reviewers for useful feedback. Xinran He was supported in part by the grant ONR MURI W911NF-11-1-0332 and DARPA SMISC W911NF-12-1-0034.

7. REFERENCES

[1] A. Adiga, C. J. Kuhlman, H. S. Mortveit, and A. K. S. Vullikanti. Sensitivity of diffusion dynamics to network uncertainty. In *Proc. 28th AAAI Conf. on Artificial Intelligence*, 2013.

[2] N. Alon, I. Benjamini, and A. Stacey. Percolation on finite graphs and isoperimetric inequalities. *Annals of Probability*, 32:1727–1745, 2004.

[3] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[4] B. Bollobás. *Random Graphs*. Cambridge University Press, second edition, 2001.

[5] C. Borgs, M. Brautbar, J. T. Chayes, and B. Lucier. Maximizing social influence in nearly optimal time. In *Proc. 25th ACM-SIAM Symp. on Discrete Algorithms*, pages 946–957, 2014.

[6] N. Buchbinder, M. Feldman, J. S. Naor, and R. Schwartz. Submodular maximization with cardinality constraints. In *Proc. 25th ACM-SIAM Symp. on Discrete Algorithms*, pages 1433–1452, 2014.

[7] W. Chen, L. V. Lakshmanan, and C. Castillo. *Information and Influence Propagation in Social Networks*. Synthesis Lectures on Data Management. Morgan & Claypool, 2013.

[8] W. Chen, Y. Yuan, and L. Zhang. Scalable influence maximization in social networks under the linear threshold model. In *Proc. 10th Intl. Conf. on Data Mining*, pages 88–97, 2010.

[9] P. Erdős and A. Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci.*, 5:17–61, 1960.

[10] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf. Uncovering the temporal dynamics of diffusion networks. In *Proc. 28th Intl. Conf. on Machine Learning*, pages 561–568, 2011.

[11] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. Learning influence probabilities in social networks. In *Proc. 3rd ACM Intl. Conf. on Web Search and Data Mining*, pages 241–250, 2010.

[12] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. A data-based approach to social influence maximization. *Proc. VLDB Endow.*, 5(1):73–84, Sept. 2011.

[13] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence in a social network. In *Proc. 9th Intl. Conf. on Knowledge Discovery and Data Mining*, pages 137–146, 2003.

[14] D. Kempe, J. Kleinberg, and E. Tardos. Influential nodes in a diffusion model for social networks. In *Proc. 32nd Intl. Colloq. on Automata, Languages and Programming*, pages 1127–1138, 2005.

[15] H. Kesten. Asymptotics in high dimension for percolation. In G. Grimmett and D. Welsh, editors, *Disorder in Physical System*, pages 219–240. Oxford University Press, 1990.

[16] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. S. Glance. Cost-effective outbreak detection in networks. In *Proc. 13th Intl. Conf. on Knowledge Discovery and Data Mining*, pages 420–429, 2007.

[17] E. Mossel and S. Roch. Submodularity of influence in social networks: From local to global. *SIAM J. Comput.*, 39(6):2176–2188, 2010.

[18] S. A. Myers and J. Leskovec. On the convexity of latent social network inference. In *Proc. 22nd Advances in Neural Information Processing Systems*, pages 1741–1749, 2010.

[19] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of the approximations for maximizing submodular set functions. *Mathematical Programming*, 14:265–294, 1978.

[20] M. Salek, S. Shayandeh, and D. Kempe. You share, I share: Network effects and economic incentives in P2P file-sharing systems. In *Proc. 6th Workshop on Internet and Network Economics (WINE)*, 2010.

[21] V. Strassen. The existence of probability measures with given marginals. *The Annals of Mathematical Statistics*, 36(2):423–439, 1965.

[22] J. Vondrák. Symmetry and approximability of submodular maximization problems. In *Proc. 50th IEEE Symp. on Foundations of Computer Science*, pages 651–670, 2009.

[23] C. Wang, W. Chen, and Y. Wang. Scalable influence maximization for independent cascade model in large-scale social networks. *Data Mining and Knowledge Discovery Journal*, 25(3):545–576, 2012.

[24] D. J. Watts and S. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.