Probabilistic Causal Analysis of Social Influence

Francesco Bonchi
ISI Foundation, Italy
Eurecat, Spain
francesco.bonchi@isi.it

Francesco Gullo
UniCredit, R&D Dept., Italy
gullof@acm.org

Bud Mishra
New York University, NY, USA
mishra@nyu.edu

Daniele Ramazzotti
Stanford University, CA, USA
daniele.ramazzotti@stanford.edu

ABSTRACT

Mastering the dynamics of social influence requires separating, in a database of information propagation traces, the genuine causal processes from temporal correlation, i.e., homophily and other spurious causes. However, most studies to characterize social influence, and, in general, most data-science analyses focus on correlations, statistical independence, or conditional independence. Only recently, there has been a resurgence of interest in "causal data science," e.g., grounded on causality theories. In this paper we adopt a principled causal approach to the analysis of social influence from information-propagation data, rooted in the theory of probabilistic causation.

Our approach consists of two phases. In the first one, in order to avoid the pitfalls of misinterpreting causation when the data spans a mixture of several subtypes ("Simpson's paradox"), we partition the set of propagation traces into groups, in such a way that each group is as less contradictory as possible in terms of the hierarchical structure of information propagation. To achieve this goal, we borrow the notion of "*agony*" [26] and define the Agony-bounded Partitioning problem, which we prove being hard, and for which we develop two efficient algorithms with approximation guarantees. In the second phase, for each group from the first phase, we apply a constrained MLE approach to ultimately learn a minimal causal topology. Experiments on synthetic data show that our method is able to retrieve the genuine causal arcs w.r.t. a ground-truth generative model. Experiments on real data show that, by focusing only on the extracted causal structures instead of the whole social graph, the effectiveness of predicting influence spread is significantly improved.

ACM Reference Format:
Francesco Bonchi, Francesco Gullo, Bud Mishra, and Daniele Ramazzotti. 2018. Probabilistic Causal Analysis of Social Influence. In *The 27th ACM International Conference on Information and Knowledge Management (CIKM '18), October 22–26, 2018, Torino, Italy*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3269206.3271756

CIKM '18, October 22–26, 2018, Torino, Italy

ACM ISBN 978-1-4503-6014-2/18/10…$15.00
https://doi.org/10.1145/3269206.3271756

1 INTRODUCTION

Sophisticated empirical analyses of a variety of social phenomena have now become possible as a result of two related developments: the explosion of on-line social networks and the unprecedented availability of network data. In this context a phenomenon that has attracted a lion's share of interest is the study of *social influence*, i.e., the causal processes motivating the actions of a user to induce similar actions from her peers.Mastering the dynamics of social influence (i.e., observing, understanding, and measuring it) could pave the way for many important applications, among which the most prominent is *viral marketing*, i.e., the exploitation of social influence by letting adoption of new products to hitch-hike on a viral meme sweeping through a social influence network [9, 19, 31]. Moreover, patterns of influence can be taken as a proxy for trust and exploited in trust-propagation analysis [22, 25, 51, 56] in large networks and in P2P systems. Other applications include personalized recommendations [48, 49], feed ranking in social networks [30, 44], and the analysis of information propagation in social media [8, 12, 42, 55].

Considerable attention has been devoted to the problem of estimating the strength of influence in a social network [24, 34, 43], mainly by observing how often information successfully propagates between any two users. However, by social influence it is understood to mean a *genuine causal process*, i.e., a force that an individual exerts on her peers to an extent to introduce a change in opinions or behaviors. Thus, to properly deal with influence-induced viral sweeps, it does not suffice to record temporally and spatially similar events, as they could just be due to correlation, or other *spurious* causes [5, 40], such as common causes (e.g., assortative mixing [39] or "homophily"), unobserved influence, or transitivity of influence [46]. Other confounding factors (Simpson's paradox [54], temporal clustering [1], screening-off [29], etc.) make the problem of separating genuine causes from spurious ones even harder. The problem is intimately connected to the fundamental question in many causal theories of misinterpreting causation when partitioning genuine vs. spurious causes, and it points to the need for a rigorous foundation, for example, the ones developed by such prominent philosophers as Cartwright, Skyrms, or Suppes [29].

In this paper we tackle the problem of deriving, from a database of propagation traces (i.e., traces left by entities flowing in a social network), a set of directed acyclic graphs (DAGs), each of which representing a genuine causal process underlying the network: multiple causal processes might involve *different communities*, or represent *the backbone of information propagation for different topics*.

Our approach builds upon *Suppes' theory of probabilistic causation* [50], a theory grounded on a logical foundation, having a

long history in philosophy, and whose empirical effectiveness has been largely demonstrated in numerous contexts [10, 11, 21]. Although exhibiting various known limitations, such a theory can be expressed in probabilistic computational tree logic with efficient model checkers that allow for devising efficient learning algorithms [32]. All of this makes it particularly appealing given the computational burden of the problem we tackle in this work, and thus preferable to other more sophisticated yet computationally heavier theories, such as the one advocated by Judea Pearl and indirectly related to the philosophical foundations laid by Lewis and Reichenbach [40]. The central notion behind Suppes' theory is *prima facie causes*: to be recognized as a cause, an event must occur before the effect (*temporal priority*) and must lead to an increase of the probability of observing the effect (*probability raising*). In the context of social influence the latter means that the most influential users heavily influence the behavior of their social ties.

Challenges and contributions. Suppes' and similar notions of causality suffer from a well-known weakness of misinterpreting causation when the data spans a mixture of subtypes, i.e., "Simpson's paradox" [5]. This issue is predominant in information propagation, as users have different interests, and different topics produce different propagation traces, affecting diverse populations. Hence, analyzing the social dynamics as a whole will most likely end up in misinterpreting causation. In this work we tackle this problem by partitioning the input propagation traces and assigning a different causal interpretation for each set of the identified partition.

More in detail, our goal is to partition the propagation traces into groups, in such a way that each group is as minimally contradictory as possible in terms of the hierarchical structure of information propagation. For this purpose, we borrow the notion of "*agony*", introduced by Gupte *et al.* [26], as a measure of how clearly defined a hierarchical structure is in a directed graph. We introduce the Agony-bounded Partitioning problem, where the input propagation traces are partitioned into groups exhibiting small agony. We prove that Agony-bounded Partitioning is **NP**-hard, and devise efficient algorithms with provable approximation guarantees.

For each group identified in the partitioning step, the part of social network spanned by the union of all propagation traces in every such group may be interpreted as a *prima-facie* graph, i.e., a graph representing all causal claims (of the causal process underlying that group) that are consistent with the Suppes' notion of prima facie causes. As prima facie causes may be either genuine or spurious [29], such prima-facie graphs need to be further processed so as to filter out spurious claims. We accomplish this second step of our approach by selecting the minimal set of arcs that are the most explanatory of the input propagation traces within the corresponding group. We cast this problem as a constrained maximum likelihood estimation (MLE) problem. The result is a minimal causal structure (a dag) for each group, representing all the genuine causal claims of the social-influence causal process underlying that group.

We present an extensive experimental evaluation on synthetic data with a known ground-truth generative model, and show that our method achieves high accuracy in the task of identifying genuine causal arcs in the ground truth. On real data we show that our approach can improve subsequent tasks that make use of social influence, in particular in the task of predicting influence spread.

To summarize, the main contributions of this paper are:

- We adopt a principled causal approach to the analysis of social influence from information-propagation data, following Suppes' probabilistic causal theory.
- We introduce the Agony-bounded Partitioning problem, where the input set of propagations (dags) is partitioned into groups exhibiting a clear hierarchical structure. We prove the **NP**-hardness of the problem and devise efficient algorithms with approximation guarantees. For each resulting group of propagation traces we apply a constrained MLE approach to ultimately learn a minimal causal topology.
- Experiments on synthetic data show that our method is able to retrieve the genuine causal arcs with respect to a known ground-truth generative model. Experiments on real data show that, by focusing only on the causal structures extracted instead of the whole social network, we can improve the effectiveness of predicting influence spread.

The rest of the paper is organized as follows. Section 2 overviews the related literature. Section 3 discusses input data and background notions. Section 4 defines our problem, by formulating and theoretically characterizing the Agony-bounded Partitioning problem, and discussing the minimal-causal-topology learning problem. Section 5 describes the approximation algorithms for Agony-bounded Partitioning. Section 6 presents the experimental evaluation, while Section 7 concludes the paper.

2 RELATED WORK

A large body of literature has focused on empirically analyzing the effects of social influence and other factors of correlation, such as homophily. Crandall *et al.* [17] present a study over Wikipedia editors' social network and LiveJournal blogspace, showing that there exists a feedback effect between users' similarity and social influence, and that combining features based on social ties and similarity is more predictive of future behavior than either social influence or similarity features alone. Cha *et al.* [13] analyze information dynamics on the Flickr social network and provide empirical evidence that the social links are the dominant method of information propagation.Leskovec *et al.* [35, 36] show patterns of influence by studying person-to-person recommendation for purchasing books and videos, while finding conditions under which such recommendations are successful. Hill *et al.* [28] analyze the adoption of a new telecommunications service and show that it is possible to predict with a significant confidence whether customers will sign up for a new calling plan once one of their phone contacts does so.

Additional effort has been devoted to distinguishing genuine social influence from homophily and other external factors. Anagnostopoulos *et al.* [6] devise techniques (e.g., *shuffle test* and *edge-reversal* test) to separate influence from correlation, showing that in Flickr, while there is substantial social correlation in tagging behavior, such a correlation cannot be ascribed to influence. Aral *et al.* [7] develop a matched sample-estimation framework, which accounts for both homophily and influence. Fond and Neville [20] present a randomization technique for temporal-network data where the

attributes and links change over time. Sharma and Cosley [47] propose a statistical procedure to discriminate between social influence and personal preferences in online activity feeds.

Our work distinguishes itself from this literature as it adopts a principled causal approach to the analysis of social influence from information-propagation data, rooted in probabilistic causal theory. The idea of adopting Suppes' theory to infer causal structures and represent them into graphical models is not new, but it has been used in completely different contexts, such as cancer-progression analysis [37, 41], discrimination detection in databases [10], and financial stress-testing scenarios [21].

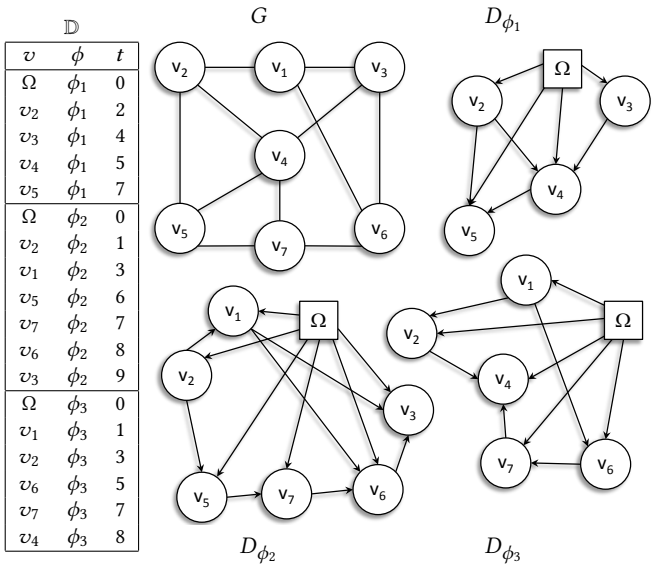
3 PRELIMINARIES

Input data. The data we take as input in this work consists of: (*i*) a directed graph $G = (V, A)$ representing a network of interconnected objects and hereinafter informally referred to as the "social graph", (*ii*) a set $\mathcal{E}$ of *entities*, and (*iii*) a set $\mathbb{O}$ of *observations* involving the objects of the network and the entities in $\mathcal{E}$. Each observation in $\mathbb{O}$ is a triple $\langle v, \phi, t\rangle$, where $v \in V$, $\phi \in \mathcal{E}$, and $t \in \mathbb{N}^+$, denoting that the entity ϕ is observed at node v at time t. For instance, G may represent users of a social network interconnected via a follower-followee relation, entities in $\mathcal{E}$ may correspond to pieces of multimedia content (e.g., photos, videos), and an observation $\langle v, \phi, t\rangle \in \mathbb{O}$ may refer to the event that the multimedia item ϕ has been enjoyed by user v at time t. We assume an entity cannot be observed multiple times at the same node; should this happen, we consider only the first one (in order of time) of such observations.

The set $\mathbb{O}$ of observations can alternatively be viewed as a database $\mathbb{D}$ of *propagation traces* (or simply *propagations*), i.e., traces left by entities "flowing" over G. Formally, a propagation trace of an entity ϕ corresponds to the subset $\{\langle v, \phi', t\rangle \in \mathbb{O} \mid \phi' = \phi\}$ of all observations in $\mathbb{O}$ involving that entity. Coupled with the graph G, the database of propagations corresponds to a set $\mathbb{D} = \{D_\phi \mid \phi \in \mathcal{E}\}$ of *directed acyclic graphs* (DAGs), where, for each $\phi \in \mathcal{E}$, $D_\phi = (V_\phi, A_\phi)$, $V_\phi = \{v \in V \mid \langle v, \phi, t\rangle \in \mathbb{O}\}$, $A_\phi = \{(u, v) \in A \mid \langle u, \phi, t_u\rangle \in \mathbb{O}, \langle v, \phi, t_v\rangle \in \mathbb{O}, t_u < t_v\}$. Note that each $D_\phi \in \mathbb{D}$ contains no cycles, due to the assumption of time irreversibility. In the remainder of the paper we will refer to $\mathbb{D}$ *as a database of propagations* or *as a set of* DAGs interchangeably. Also, we assume that each propagation is started at time 0 by a dummy node $\Omega \notin V$, representing a source of information external to the network that is implicitly connected to all nodes in V. An example of our input is provided in Figure 1. Given a set $\mathcal{D} \subseteq \mathbb{D}$ of propagations, $G(\mathcal{D})$ denotes the union graph of all DAGs in $\mathcal{D}$, where the union of two graphs $G_1 = (V_1, A_1)$ and $G_2 = (V_2, A_2)$ is $G_1 \cup G_2 = (V_1 \cup V_2 \setminus \{\Omega\}, \{(u, v) \in A_1 \cup A_2 \mid u \neq \Omega, v \neq \Omega\})$.[1] Note that, although $\mathcal{D}$ is a set of DAGs, $G(\mathcal{D})$ is *not necessarily* a DAG.

Hierarchical structure. As better explained in the next section, in our approach we borrow the notion of "*agony*" introduced by Gupte et al. [26] to reconstruct a proper hierarchical structure of a directed graph $G = (V, A)$. Such a notion is defined as follows. Consider a ranking function $r : V \to \mathbb{N}$ for the nodes in V, such that the inequality $r(u) < r(v)$ expresses the fact that u is "higher" in the hierarchy than v, i.e., the smaller $r(u)$ is, the more u is an

[1]For the sake of presentation of the technical details, we assume union graphs not containing the dummy node Ω.



v	ϕ	t
Ω	ϕ_1	0
v_2	ϕ_1	2
v_3	ϕ_1	4
v_4	ϕ_1	5
v_5	ϕ_1	7
Ω	ϕ_2	0
v_2	ϕ_2	1
v_1	ϕ_2	3
v_5	ϕ_2	6
v_7	ϕ_2	7
v_6	ϕ_2	8
v_3	ϕ_2	9
Ω	ϕ_3	0
v_1	ϕ_3	1
v_2	ϕ_3	3
v_6	ϕ_3	5
v_7	ϕ_3	7
v_4	ϕ_3	8

Figure 1: ***An example of the input of our problem: a social graph G, and a database of propagation traces $\mathbb{D}$ defined over a set of entities $\mathcal{E} = \{\phi_1, \phi_2, \phi_3\}$. Here G is represented undirected: each edge corresponds to the two directed arcs. Each propagation is started at time 0 by a dummy node $\Omega \notin V$. Given the graph G, the propagation database $\mathbb{D}$ is equivalent to the set $\{D_{\phi_1}, D_{\phi_2}, D_{\phi_3}\}$ of DAGs.***

"early-adopter". If $r(u) < r(v)$, then the arc $u \to v$ is expected and does not result in any "social agony". If, instead, $r(u) \geq r(v)$ the arc $u \to v$ leads to agony, because it means that u has a follower v (in the social-graph terminology) that is higher-ranked than u itself. Therefore, given a graph G and a ranking r, the agony of each arc (u, v) is defined as $\max\{r(u) - r(v) + 1, 0\}$, and the agony $a(G, r)$ of the whole graph for a given ranking r is the sum over all arcs: $a(G, r) = \sum_{(u,v)\in A} \max\{r(u) - r(v) + 1, 0\}$. In most contexts (as in ours) the ranking r is not explicitly provided. The objective thus becomes finding a ranking that minimizes the agony of the graph. This way, one can compute the agony of any graph G as $a(G) = \min_r a(G, r)$. As a DAG implicitly induces a partial order over its nodes, it has zero agony: the nodes of a DAG form a perfect hierarchy. For instance, in the DAGs $D_{\phi_1}, D_{\phi_2}, D_{\phi_3}$ in Figure 1, it is sufficient to take the temporal ordering as a ranking to get no agony, i.e., $r(u) = t_u$, where $\langle u, \phi_i, t_u\rangle \in D_{\phi_i}$. On the other hand, merging several DAGs leads to a graph that is not necessarily a DAG, and can thus have non-zero agony. In fact, a cycle of length k (and not overlapping with other cycles) incurs agony equal to k [26]. A ranking r yielding minimum agony for the union graph $D_{\phi_1} \cup D_{\phi_2}$ of the DAGs D_{ϕ_1} and D_{ϕ_2} in Figure 1 is $(v_2 : 0)(v_1 : 1)(v_4 : 2)(v_5 : 3)(v_7 : 4)(v_6 : 5)(v_3 : 6)$. This ranking yields no agony on all the arcs, except for arc $v_3 \to v_4$, which incurs agony equal to the length of the cycle involving v_3 and v_4, i.e., 6-2+1 = 5.

Gupte et al. [26] provide a polynomial-time algorithm for finding a ranking of minimum agony, running in $O(|V| \times |A|^2)$ time. Tatti [52] devises a more efficient method, which takes $O(|A|^2)$ time in the worst case, but recognized as much faster in practice. In another work [53] Tatti also provides methods to compute agony on weighted graphs and when cardinality constraints are specified.

4 PROBLEM STATEMENT

The main goal of this work consists in deriving *a set of causal DAGs that are well-representative of the social-influence dynamics underlying an input database of propagation traces*. The idea is that every derived DAG represents a *causal process* underlying the input data, where an arc from u to v models the situation where *actions of user u are likely to cause actions of user v*. The various causal processes may correspond to the *different topics* on which users exert *social influence* on each other. For instance, a causal DAG concerning politics may contain an arc $u \rightarrow v$ meaning that user u is influential on user v for politics-related matters, while a DAG for the music context may miss that influence relation or have it inverted.

To achieve our goal, we resort to the theory of *probabilistic causation* [29], which was introduced by Suppes in [50], via the notion of *prima facie causes*.

Definition 1 (Prima facie causes [50]). *For any two events c (cause) and e (effect), occurring respectively at times t_c and t_e, under the mild assumption that the probabilities $\mathcal{P}(c)$ and $\mathcal{P}(e)$ of the two events satisfy the condition $0 < \mathcal{P}(c), \mathcal{P}(e) < 1$, the event c is called a* prima facie cause *of the event e if it occurs* before *e and* raises the probability *of e, i.e., $t_c < t_e \ \wedge \ \mathcal{P}(e \mid c) > \mathcal{P}(e \mid \overline{c})$.*

While being a well-acknowledged definition of causality, Suppes' prima facie causes suffer from three main limitations: (*i*) If the input data spans multiple causal processes, causal claims may be hidden or misinterpreted if one looks at the data as a whole: this anomaly is the well-known Simpson's paradox [54] . (*ii*) Suppes' definition lacks any characterization in terms of *spatial proximity*, which is a critical concept in social influence, as users who never interact with each other should intuitively not be involved in a causal relation. (*iii*) As discussed in the Introduction, prima facie causes may be either *genuine* or *spurious*. The desideratum is that only the former ones are detected and presented as output.

Motivated by the above discussion, we propose to derive the desired social-influence causal DAGs with a two-step methodology, where the successive execution of the two steps ultimately output social-influence causal DAGs that are consistent with the principles of Suppes' theory, while at the same time overcoming its limitations. Specifically, we accomplish the two steps by formulating and solving two problems:

- Agony-bounded Partitioning (Section 4.1), a novel combinatorial-optimization problem mainly designed to get rid of the Simpson's paradox, where the input propagation set is partitioned into homogeneous groups. Formulating, theoretically characterizing, and solving this problem constitute the main technical contribution of this work.
- Minimal Causal Topology (Section 4.2), a learning problem where a minimal causal DAG is derived from the union graph of each group of propagations identified in the first step. The main goal here is to remove all the spurious relationships.

4.1 Partitioning the propagation set

The input propagation set is typically so large that it may easily span multiple causal processes, each of which corresponds to a different social-influence topic. Due to the aforementioned Simpson's paradox, attempting to infer causality from all input propagations at once is therefore bound to fail. For this purpose, we aim at preventively partitioning the input set of propagations into *homogeneous* groups, each of which is likely to identify a single causal process. Homogeneity means that the propagations in a group should exhibit as few "contradictions" as possible, where a contradiction arises when a user u is a (direct or indirect) influencer for user v in some propagations, while the other way around holds in some other propagations. In other words, a homogeneous group of propagations should be such that the union graph of all those propagations has a clear *hierarchical structure*, i.e., it is as similar as possible to a DAG. *This requirement is fully reflected in the notion of agony* discussed in Section 3. For this reason, here we resort to that notion and define the Agony-bounded Partitioning problem: given a threshold $\eta \in \mathbb{N}$, partition the input propagations into the minimum number of sets such that the union graph of every of such sets has agony no more than η. Additionally, we require for each set to be limited in size and to exhibit a union graph that is (weakly) connected, as too large propagation sets or disconnected union graphs are unlikely to represent single causal processes.

Problem 1 (Agony-bounded Partitioning). *Given a set $\mathbb{D}$ of DAGs and two positive integers $K, \eta \in \mathbb{N}$, find a partition $\mathbf{D}^* \in \mathcal{P}(\mathbb{D})$ (where $\mathcal{P}(\cdot)$ denotes the set of all partitions of a given set) such that*

$$\mathbf{D}^* \ = \ \text{argmin}_{\mathbf{D} \in \mathcal{P}(\mathbb{D})} |\mathbf{D}| \qquad \textit{subject to}$$
$$\forall \mathcal{D} \in \mathbf{D}: \ a(G(\mathcal{D})) \leq \eta, \ \ |\mathcal{D}| \leq K, \ \ G(\mathcal{D}) \textit{ is weakly-connected.} \quad (1)$$

For every $\mathcal{D} \in \mathbf{D}^*$, we informally term the union graph $G(\mathcal{D})$ *prima-facie graph*. In fact, apart from addressing the Simpson's paradox, every union graph $G(\mathcal{D})$ may be interpreted as a graph containing all prima-facie causes underlying the propagation group $\mathcal{D}$. To this purpose, note that every $G(\mathcal{D})$ reflects both the temporal-priority and the probability-raising conditions in Suppes' theory. Temporal priority is guaranteed by the input itself, as an arc $u \rightarrow v$ in some input DAG (and, thus, in every output union graph) exists only if an input observation exists in $\mathbb{O}$, where the same entity is observed first in u and then in v. Probability raising arises as we ask for a partition with the minimum number of small-agony groups, which means that the identified groups will be as large as possible (as long as the problem constraints are satisfied). This way, every output group will likely contain as much evidence as possible for every causal claim $u \rightarrow v$, i.e., a large evidence that the effect is observed thanks to the cause. At the same time, the output union graphs overcome the limitation of missing spatial proximity in Suppes' theory: all arcs within the output graphs come from the input social graph G, which implicitly encodes a notion of spatial proximity corresponding to the social relationships among users.

Switching to a technical level, a simple observation on Problem 1 is that it is well-defined, as it always admits at least the solution where every DAG forms a singleton group. A more interesting characterization is the **NP**-hardness of the problem, which we formally state next. In Section 5 we will instead present the approximation algorithms we designed to solve the problem.

Theorem 1. *Problem 1 is* **NP***-hard.*

Proof. We consider the well-known **NP**-complete (decision version of the) Set Cover problem [15]: given a universe $U = \{e_1, \ldots, e_n\}$ of elements, a set $\mathcal{S} = \{S_1, \ldots, S_m\} \subseteq 2^U$ of subsets

of U, and a positive integer p, is there a subset $\mathcal{S}^* \subseteq \mathcal{S}$ of size $\leq p$ covering the whole universe U? We reduce Set Cover to the decision version of our Agony-bounded Partitioning, which is as follows: given a set $\mathbb{D}$ of dags, and three positive integers K, η, q, is there a partition of $\mathbb{D}$ of size $\leq q$ satisfying all constraints in Equation (1)? Given an instance $I = \langle U, \mathcal{S}, p \rangle$ of Set Cover, we construct an instance $I' = \langle \mathbb{D}, K, \eta, q \rangle$ of Agony-bounded Partitioning so that I is a yes-instance for Set Cover if and only if I' is a yes-instance for Agony-bounded Partitioning. To this end, we set $\eta = m(n+2) - 1$, $K = |\mathbb{D}|$, $q = p$, and we let $\mathbb{D}$ be composed of a dag D_i for every element $e_i \in U$. Each dag $D_i \in \mathbb{D}$ has node set $V(D_i) = \{u_i, v_1, \ldots, v_m\} \cup V_1^i \cup \cdots \cup V_m^i$, i.e., it comprises a node u_i, nodes $v_1, \ldots, v_m$ such that v_j corresponds to set $S_j \in \mathcal{S}$, and further node sets $V_1^i, \ldots, V_m^i$ defined as

$$V_j^i = \begin{cases} V_j \setminus \{v_{ji}\}, & \text{if } e_i \notin S_j, \\ V_j \setminus \{v_{j0}, v_{ji}\}, & \text{otherwise,} \end{cases}$$

where $V_j = \{v_{j0}, v_{j1}, \ldots, v_{jn}\}$. The arc set of D_i is composed of an arc from u_i to every node v_j, and, for each v_j, (i) an arc (v_j, v_{j0}) (only if $v_{j0} \in V_j^i$), (ii) an arc (v_{jn}, v_j) (only if $v_{jn} \in V_j^i$), and (iii) an arc (v_{jk}, v_{jk+1}) for every $k \in [0..n-1]$ such that $v_{jk}, v_{jk+1} \in V_j^i$. It is easy to see that each D_i can be constructed in polynomial time in the size of I, and is actually a dag. In this regard, note that, for each v_j, V_j^i misses at least one node needed to form the cycle $v_j \to v_{j0} \to \cdots \to v_{jn} \to v_j$ (i.e., at least the node v_{ji}).

Let $\mathbf{D}^+ \subseteq 2^{\mathbb{D}}$ be the set of all dag sets that are admissible for Agony-bounded Partitioning on instance I'. We now show that $\mathbf{D}^+ \equiv \mathcal{S}^+$, where $\mathcal{S}^+ = \bigcup_{S \in \mathcal{S}} 2^S$ corresponds to the set $\mathcal{S}$ augmented by all possible subsets of every set within it. First of all, we observe that every dag set $\mathcal{D} \subseteq \mathbb{D}$ meets both the constraint on the connectedness of the union graph $G(\mathcal{D})$ (each dag is connected and shares at least one node with every other dag in $\mathbb{D}$), and the constraint on the maximum size (as $K = |\mathbb{D}|$). This way, the set $\mathbf{D}^+$ of admissible dags is determined by the agony constraint only. For every $j \in [1..m]$, the union graph $G(\mathcal{D})$ of a dag set $\mathcal{D} \subseteq \mathbb{D}$ contains a cycle $v_j \to v_{j0} \to \cdots \to v_{jn} \to v_j$ only if set S_j does not contain all elements (corresponding to the dags) in $\mathcal{D}$. Denoting by $m_{\mathcal{D}}$ the number of such cycles being present in $G(\mathcal{D})$, the agony of $G(\mathcal{D})$ is equal to $m_{\mathcal{D}}(n+2)$, as all cycles have length $n+2$ and they are all node-disjoint. Therefore, the agony of $G(\mathcal{D})$ exceeds the threshold η only if $m_{\mathcal{D}} = m$, i.e., only if there exists no set in $\mathcal{S}^+$ containing $\mathcal{D}$, thus meaning that $\mathbf{D}^+ \equiv \mathcal{S}^+$.

We are now ready to show the desired "$\Leftrightarrow$" implication between instances I and I'. As for the "$\Rightarrow$" part, we notice that, from the covering $\mathcal{S}^*$ representing the solution of Set Cover on I, it can always be derived a covering $\mathcal{S}'$ of size $|\mathcal{S}'| = |\mathcal{S}^*|$ that is a partition of U (e.g., by processing each $S \in \mathcal{S}^*$ following some ordering, and replacing any set S containing a subset $S' \subseteq S$ of elements covered by already-processed sets with the set $S \setminus S'$). $\mathcal{S}'$ is a solution of Agony-bounded Partitioning on I' as its size is $|\mathcal{S}'| = |\mathcal{S}^*| \leq p = q$, it is a partition of U (and, therefore, of $\mathbb{D}$), and it is admissible as $\mathcal{S}' \subseteq \mathcal{S}^+$. The "$\Leftarrow$" part holds due to the following reasoning. Given the solution $\mathbf{D}^*$ of Agony-bounded Partitioning on I', one can derive a covering $\mathbf{D}'$ where every set $\mathcal{D} \in \mathbf{D}^*$ such that $\mathcal{D} \in \mathcal{S}^+ \setminus \mathcal{S}$ is replaced with the original set $S_{\mathcal{D}} \in \mathcal{S}$ where $\mathcal{D}$ has been derived from. $\mathbf{D}'$ is a solution of Set Cover on I as its size is $|\mathbf{D}'| = |\mathbf{D}^*| \leq q = p$, it covers U, and it is a subset of $\mathcal{S}$. □

4.2 Learning the minimal causal topology

As mentioned above, prima facie causes may be genuine or spurious [29]. Thus, from the prima-facie graphs identified in the previous step we still need to remove the spurious relationships. To this end, we aim at identifying a *minimal causal topology* for each prima-facie graph, i.e., selecting the minimal set of arcs that best explain the input propagations spanned by that graph.

Given a partition $\mathbf{D}^*$ of the input database $\mathbb{D}$ of propagations (computed in the previous step), we first reconstruct a dag $G_D(\mathcal{D})$ from the prima-facie graph $G(\mathcal{D})$ of every group $\mathcal{D} \in \mathbf{D}^*$. To this end, we exploit the by-product of agony minimization on $G(\mathcal{D})$, i.e., a ranking r of the nodes in $G(\mathcal{D})$ (see Section 3). Specifically, we build $G_D(\mathcal{D})$ by taking all and only those arcs of $G(\mathcal{D})$ that are in accordance with r, i.e., all arcs (u, v) such that $r(u) < r(v)$. Then, for every reconstructed dag $G_D(\mathcal{D})$, we learn its minimal causal topology via (constrained) *maximum likelihood estimation* (MLE), where the arcs of $G_D(\mathcal{D})$ maximizing a likelihood score such as *Bayesian Information Criterion* (BIC) [45] or *Akaike Information Criterion* (AIC) [4] are identified (we experimented with both criteria, see Section 6). More precisely, given a database $\mathbb{D}$ of propagations and a set $\hat{A} \subseteq A$ of arcs, we define

$$f(\hat{A}, \mathbb{D}) = LL(\mathbb{D}|\hat{A}) - \mathcal{R}(\hat{A}),$$

where $LL(\cdot)$ is the log-likelihood, while $\mathcal{R}(\cdot)$ is a regularization term. The dag induced by $\hat{A}$ in turn induces a probability distribution over its nodes $\{u_1, \ldots, u_n\}$:

$$\mathcal{P}(u_1, \ldots, u_n) = \prod_{u_i=1}^{n} \mathcal{P}(u_i \mid \pi_i), \qquad \mathcal{P}(u_i \mid \pi_i) = \boldsymbol{\theta}_{u_i|\pi_i},$$

where $\pi_i = \{u_j \mid u_j \to u_i \in \hat{A}\}$ are u_i's parents in the dag, and $\boldsymbol{\theta}_{u_i|\pi(u_i)}$ is a probability density function. Then, the log-likelihood of the network is defined as:

$$LL(\mathbb{D}|\hat{A}) = \log \mathcal{P}(\mathbb{D} \mid \hat{A}, \boldsymbol{\theta}).$$

The regularization term $\mathcal{R}(\hat{A})$ introduces a penalty term for the number of parameters in the model and the size of the data. Specifically, S being the number of samples, $\mathcal{R}(\hat{A})$ is defined as $|\hat{A}|$ for AIC and $\frac{|\hat{A}|}{2} \log S$ for BIC.

The problem we tackle here is formally stated as follows.

Problem 2 (Minimal Causal Topology). *Given a database $\mathbb{D}$ of propagations and a dag $G_D(\mathcal{D}) = (V_D, A_D)$, find $A_D^*(\mathcal{D}) = \arg\max_{\hat{A}_D \subseteq A_D} f(\hat{A}_D, \mathbb{D})$.*

Even if constrained (the output arc set A_D^* must be a subset of A_D), Problem 2 can easily be shown to be still **NP**-hard [14]. Therefore, we solve the problem by a classic greedy hill-climbing heuristic, whose effectiveness has been well-recognized [33].

The ultimate output of our second step (and of our overall approach) is a set of dags $\{G_D^*(\mathcal{D})\}_{\mathcal{D} \in \mathbf{D}^*}$, where each $G_D^*(\mathcal{D})$ is the dag identified by the arc set $A^*(\mathcal{D})$ as defined in Problem 2. Every $G_D^*(\mathcal{D})$ is a causal dag representative of a specific social-influence causal process underlying the input propagation database $\mathbb{D}$.

5 ALGORITHMS

In this section we focus on the algorithms for the Agony-bounded Partitioning problem (Problem 1). Due

Algorithm 1 Two-step-Agony-Partitioning

Input: A set $\mathbb{D}$ of DAGs; two positive integers K, η
Output: A partition $\mathbf{D}^*$ of $\mathbb{D}$

1: $\mathbf{D}^+ \leftarrow$ Mine-Valid-DAG-sets($\mathbb{D}$, K, η)
2: $\mathbf{D}^* \leftarrow$ Greedy-Set-Cover($\mathbf{D}^+$)

to its **NP**-hardness, we clearly cannot aim at optimality, and focus instead on the design of effective and efficient approximation algorithms. Specifically, we first show how a simple two-step strategy solves the problem with provable guarantees. This method however suffers from the limitation that the first step is exponential in the size of the input DAG set, which considerably limits its applicability in practice. We hence design a more refined sampling-based algorithm, which still comes with provable guarantees, while also overcoming the exponential blowup.

A simple two-step algorithm. An immediate solution to Problem 1 consists in first computing all subsets of the input DAG set $\mathbb{D}$ that satisfy the constraints on agony, size, and connectedness listed in Equation (1) (Step I), and then taking a minimum-sized subset of these valid DAG sets that is a partition of $\mathbb{D}$ (Step II).

Step I can be solved by resorting to frequent-itemset mining [2]. Specifically, in our setting the DAGs in $\mathbb{D}$ correspond to items and the support of a DAG set (itemset) $\mathcal{D}$ is given by the agony of the union graph $G(\mathcal{D})$. It is easy to see that the constraint on agony is monotonically non-decreasing as the size of a DAG set increases, i.e., $a(G(\mathcal{D}')) \leq a(G(\mathcal{D}''))$ for any two DAG sets $\mathcal{D}' \subseteq \mathcal{D}''$. This way, any downward-closure-based algorithm for frequent itemset-mining (e.g., Apriori [3]) can easily be adapted to mine all DAG sets satisfying the agony constraint. The two additional constraints on (1) connectedness and (2) size can easily be fulfilled by (1) filtering out all mined DAG sets that are not connected, and (2) stopping the mining procedure once the maximum size K has been reached.

As for Step II, we observe that solving Set Cover [15] on the set $\mathbf{D}^+ \subseteq 2^{\mathbb{D}}$ mined in Step II gives the optimal solution to Problem 1 too. Therefore, we solve Step II by the well-known Set Cover greedy algorithm which iteratively brings to the solution the set having the maximum number of still uncovered elements [15], and has approximation factor logarithmic in the maximum size of an input set. Note that, as $\mathbf{D}^+$ contains all subsets of every set $\mathcal{D} \in \mathbf{D}^+$, at each step of the greedy Set Cover algorithm there are multiple sets maximizing the number of uncovered elements, all of them being equivalent in terms of soundness and approximation ratio. Among them, we therefore choose the one having no already-covered elements. This way, the output covering is guaranteed to be a partition of $\mathbb{D}$, as required by Agony-bounded Partitioning.

The outline of this simple two-step method is reported as Algorithm 1. The algorithm achieves a $\log K$ approximation ratio.

Theorem 2. *Algorithm 1 is a* $(\log K)$*-approximation for Problem 1.*

Proof. Step I of Algorithm 1 computes all DAG sets $\mathbf{D}^+$ meeting the constraints of Problem 1. For every set $\mathcal{D} \in \mathbf{D}^+$, $\mathbf{D}^+$ contains all subsets $\mathcal{D}' \subseteq \mathcal{D}$ too. This ensures that, for every feasible solution $\hat{\mathcal{S}}$ of Set Cover on $\mathbf{D}^+$, there exists a Set Cover solution $\hat{\mathcal{S}}'$ being a partition of $\mathbb{D}$ and having the same objective-function value as $\hat{\mathcal{S}}$. Thus, for any β-approximation solution of Set Cover on $\mathbf{D}^+$, there exists a β-approximation solution that is a partition of $\mathbb{D}$. The traditional greedy approximation algorithm for Set Cover has an approximation factor proportional to the logarithm of the maximum size of an input set. Hence, running such a greedy method on input $\mathbf{D}^+$ gives approximation guarantees of $\log K$, as all sets in $\mathbf{D}^+$ have size no more than K, due to Problem 1's constraints. □

Algorithm 2 Sampling-Agony-Partitioning

Input: A set $\mathbb{D}$ of DAGs; two positive integers K, η; a real number $\alpha \in (0, 1]$
Output: A partition $\mathbf{D}^*$ of $\mathbb{D}$

1: $\mathbf{D}^* \leftarrow \emptyset$, $\mathbb{D}_u \leftarrow \mathbb{D}$
2: **while** $|\mathbb{D}_u| > 0$ **do**
3: $\quad\mathcal{D}_s \leftarrow \emptyset$
4: $\quad$**while** $|\mathcal{D}_s| < \lceil \alpha \times \min\{K, |\mathbb{D}_u|\} \rceil$ **do**
5: $\quad\quad\mathcal{D}_s \leftarrow$ Sample-Maximal-DAG-set($\mathbb{D}_u$, K, η) {Algorithm 3}
6: $\quad\mathbf{D}^* \leftarrow \mathbf{D}^* \cup \{\mathcal{D}_s\}$, $\mathbb{D}_u \leftarrow \mathbb{D}_u \setminus \mathcal{D}_s$

Algorithm 3 Sample-Maximal-DAG-set

Input: A set $\mathbb{D}_u$ of DAGs; two positive integers K, η
Output: $\mathcal{D}_s \subseteq \mathbb{D}_u$

1: $\mathcal{D}_s \leftarrow \emptyset$, $G(\mathcal{D}_s) \leftarrow$ empty graph, $\mathbb{D}'_u \leftarrow \mathbb{D}_u$
2: **while** $|\mathcal{D}_s| < \min\{K, |\mathbb{D}_u|\} \wedge a(G(\mathcal{D}_s)) \leq \eta$ **do**
3: $\quad\mathbb{D}_s \leftarrow \emptyset$
4: $\quad$**for all** $D \in \mathbb{D}'_u$ **do**
5: $\quad\quad G(\mathcal{D}'_s) \leftarrow G(\mathcal{D}_s) \cup D$
6: $\quad\quad$**if** $G(\mathcal{D}'_s)$ is weakly connected **then**
7: $\quad\quad\quad a(G(\mathcal{D}'_s)) \leftarrow$ Compute-Agony($G(\mathcal{D}'_s)$) {cf. [52]}
8: $\quad\quad\quad$**if** $a(G(\mathcal{D}'_s)) \leq \eta$ **then** $\mathbb{D}_s \leftarrow \mathbb{D}_s \cup \{D\}$
9: $\quad\quad\quad$**else** $\mathbb{D}'_u \leftarrow \mathbb{D}'_u \setminus \{D\}$
10: $\quad D^* \leftarrow$ sample a DAG from $\mathbb{D}_s$
11: $\quad\mathcal{D}_s \leftarrow \mathcal{D}_s \cup \{D^*\}$, $G(\mathcal{D}_s) \leftarrow G(\mathcal{D}_s) \cup D^*$, $\mathbb{D}'_u \leftarrow \mathbb{D}'_u \setminus \{D^*\}$

A sampling-based algorithm. The algorithm described above is easy-to-implement and comes with provable approximation guarantees. Nevertheless, it has a major drawback that the first step is intrinsically exponential in the size of the input DAG set. Even though pruning techniques can be borrowed from the frequent-itemset-mining domain, there is no guarantee in practice that the algorithm always terminates in reasonable time. For instance, when the size of the DAG sets satisfying the input constraints tends to be large, the portion of the lattice to be visited may explode regardless of the pruning power of the specific method. This is a well-recognized issue of frequent-itemset mining [2].

Faced with this hurdle, we devise an advanced algorithm that deals with the pattern-explosion issue, while still achieving approximation guarantees. The proposed algorithm, termed Sampling-Agony-Partitioning and outlined as Algorithm 2, follows a greedy scheme and has a parameter $\alpha \in (0, 1]$ to trade off between accuracy and efficiency. The algorithm iteratively looks for a *maximal* admissible DAG set $\mathcal{D}_s$ covering a number of still uncovered DAGs no less than $\alpha \times \min\{K, |\mathbb{D}_u|\}$, where $\mathbb{D}_u$ is the set of all still uncovered DAGs (Lines 4–5). $\mathcal{D}_s$ is computed by repeatedly sampling the lattice of all admissible (and not yet covered) DAG sets, until a DAG set satisfying the requirement has been found. Sampling can be performed by, e.g., uniform [27] or random [38] maximal frequent-itemset sampling. In this work we use the latter. The outline of the sampling subroutine is in Algorithm 3. That procedure takes the set $\mathbb{D}_u$ of uncovered DAGs and selects DAGs until $\mathbb{D}_u$ has become empty, or the max size K has been reached, or the agony constraint on the union graph of the current DAG set has been violated (Lines 2–11).

To select a DAG, the subset $\mathbb{D}_s \subseteq \mathbb{D}'_u$ of admissible DAGs is first built by retaining all DAGs that meet constraints on connectedness and agony if added to the current union graph (Lines 3–9). Note that, if a DAG violates the agony constraint, it cannot become admissible anymore, thus it is permanently discarded (Line 9). The same does not hold for the connectedness constraint.

Sampling-Agony-Partitioning can be proved to be a $\frac{\log K}{\alpha}$-approximation algorithm for Agony-bounded Partitioning, as formally stated in Theorem 3. Thus, parameter α represents a knob to trade off accuracy vs. efficiency: a larger α gives a better approximation factor (thus, better accuracy), but, at the same time, leads to bigger running time as more sampling iterations are needed to find a DAG set meeting a more strict constraint.

Theorem 3. *Algorithm 2 is a $\frac{\log K}{\alpha}$-approximation for Problem 1.*

Proof. For each step t, let $u_{max}(t)$ denote the maximum number of uncovered elements that can be covered by a set not yet included in the current solution. It is known that a greedy algorithm for Set Cover finding at each step a set that covers a fraction of uncovered elements no less than $\alpha \times u_{max}(t)$ achieves $\frac{\log h_{max}}{\alpha}$ approximation guarantees, where h_{max} is the maximum size of an input set (see Lemma 2 in [16]). In our context $h_{max} \leq K$, $u_{max}(t) = \min\{K, |\mathbb{D}_u(t)|\}$, and the DAG set $\mathcal{D}_s(t)$ that is added to the solution by Algorithm 2 at Step t covers a number of still uncovered DAGs that is $\geq \alpha \times \min\{K, |\mathbb{D}_u(t)|\} = \alpha \times u_{max}(t)$. Thus, the ultimate approximation factor of Algorithm 2 is $\frac{\log K}{\alpha}$. □

Time complexity. Let H be the (maximum) number of sampling iterations needed to find a valid DAG set $\mathcal{D}_s$. The sampling subroutine (Algorithm 3) takes $O(K\ |\mathbb{D}|\ T_a)$ time, where T_a is the (maximum) time spent for a single agony computation. The subroutine is executed $O(H\ |\mathbf{D}^*|)$ times in Algorithm 2. Thus, the overall time complexity of the proposed Sampling-Agony-Partitioning algorithm is $O(H\ |\mathbf{D}^*|\ K\ |\mathbb{D}|\ T_a)$. The efficient agony-computation method in [52] takes time quadratic in the edges of the input graph. Hence, T_a is bounded by $O(|A|^2)$, where A is the arc set of the input social graph G. However, this is a very pessimistic bound, first because, as remarked by the authors themselves, the method in [52] is much faster in practice, and, more importantly, because agony computation in Algorithm 3 is run on much smaller subgraphs of G.

Implementation. A number of expedients may be employed to speed up the Sampling-Agony-Partitioning algorithm in practice, including: (*i*) Preventively discard input DAGs violating the connectedness constraint with all other DAGs. (*ii*) Algorithm 3, Lines 5 and 7: check whether D has arcs spanning nodes already in $G(\mathcal{D}_s)$; if not, skip agony computation of $G(\mathcal{D}'_s)$ (and set it equal to $a(G(\mathcal{D}_s))$). (*iii*) Approximate agony instead of computing it exactly (by, e.g., allowing only a fixed amount of time to the anytime algorithm in [52]); correctness of the algorithm is not affected as the approximated agony is an upper bound on the exact value. (*iv*) Adopt a *beam-search* strategy: sample $\widetilde{\mathbb{D}}_u$ from $\mathbb{D}_u$ (with $|\widetilde{\mathbb{D}}_u| < |\mathbb{D}_u|$, e.g., $|\widetilde{\mathbb{D}}_u| = O(\log |\mathbb{D}_u|)$) and use $\widetilde{\mathbb{D}}_u$ in Algorithm 3 instead of $\mathbb{D}_u$. (*v*) Repeat the sampling procedure at Lines 4–5 of Algorithm 2 for a maximum number of iterations; after that, sample a DAG D from $\mathbb{D}_u$, and add $\{D\}$ to the solution. (*vi*) Run Algorithm 2 until $|\mathbb{D}_u| > \epsilon$; after that, add the DAGs still in $\mathbb{D}_u$ as singletons to the solution.

6 EXPERIMENTS

In this section we show the performance of the proposed method on both synthetic data, where the true generative model is given, as well as on real data, where no ground-truth is available.

Reproducibility. Code and datasets available at bit.ly/2BEV5k9.

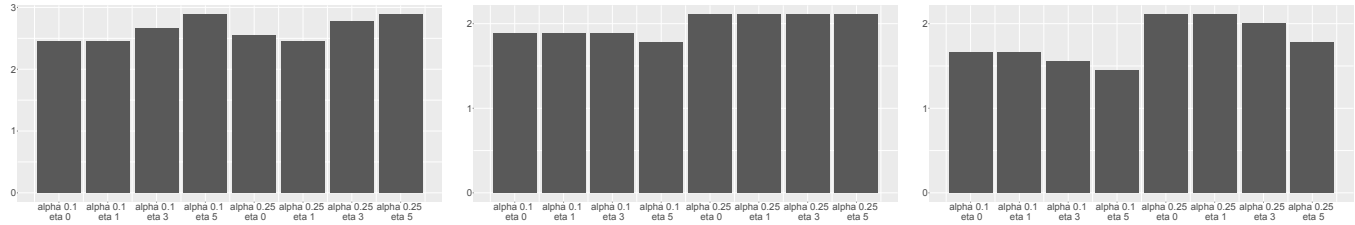
6.1 Synthetic data

Generation. We employ the following methodology:

(1) Randomly generate a directed graph $G = (V, A)$ (our social graph) with n nodes and density δ (where by density we mean number of edges divided by $\binom{n}{2}$).
(2) Randomly partition the node set V of the generated graph G into a set of k groups $\{S_1, \ldots, S_k\}$ such that: (*i*) $\forall i \in [1, k]$, $G(S_i)$ is weakly-connected, (*ii*) $\forall i \in [1, k] : |S_i| \in [\text{card}_{min}, \text{card}_{max}] \in (0, n)$, and (*iii*) the number of overlapping nodes between any pair of groups is bounded by $\text{card}_{overlap} \leq \text{card}_{max}$.
(3) Generate a causal DAG G_{cause_i} of density $\delta_{G_{cause}}$ for each of the k groups. Namely, we obtain $\mathbb{G}_{cause} = \{G_{cause_1}, \ldots, G_{cause_k}\}$, by considering the union graph of each partition independently and then randomly removing arcs from it in order to discard cycles. Then, we assign each remaining edge a random value bounded by $[p_{min}, p_{max}]$ (where $p_{min}, p_{max} \in (0, 1)$), representing the conditional probability of the child node given the connected parent (we assume the probability of any child given the absence of all its parents to be 0).
(4) Generate a set $\mathbb{O}_k$ of observations for each of the k groups in terms of triples $\langle v, \phi, t\rangle$ (with $\bigcup_k \mathbb{O}_k$ corresponding to the whole input set $\mathbb{O}$ of observations), such that each user performs at least one action, but, at the same time, she does not perform an action on at least one item of each cluster. To do this, we sample a set of traces with probability distributions induced by the causal DAGs; in this way, we obtain a set of users/entities that are observed in the trace. We then add the time t to each observation randomly, but still consistently with the total ordering defined by the DAG.

We generate synthetic data for 100 independent runs. In each run we generate social graphs of $n = 100$ nodes, of the form of either Erdös-Rény (with density δ uniformly sampled from the range $[0.05, 0.1]$), or power-law (with density δ being either 0.05 or 0.1). We partition each randomly generated social network into $k = 10$ clusters of $\text{card}_{min} = 8$, $\text{card}_{max} = 12$ and $\text{card}_{overlap} = 10$ nodes (1%). For each cluster we generate causal DAGs of density $\delta_{G_{cause}}$ uniformly sampled from the range $[0.35, 0.5]$, and p_{min}, p_{max} uniformly sampled from $(0, 1)$.

We consider both a noise-free model and a noisy model. As for the former, we assume a perfect regularity in the actions of directly connected users: all the actions of a child user follow those of one of its parents, thus letting us constrain the induced distribution of the generating DAG as follows. With u being a user in the network and $P(u)$ being all the users with arcs pointing to u (i.e., u's parent set), the probability of observing any action from u is 0 if none of u's parents had performed an action before (i.e., $P(u)$ are independently influencing u). As far as the noisy model, we consider probabilities



(a) Erdös-Rény social graph (b) Power-law $\delta = 0.05$ social graph (c) Power-law $\delta = 0.1$ social graph

Figure 2: Synthetic data: execution time of the proposed PSC method (milliseconds), by varying the α and η parameters and the social graph ($|\mathbb{O}| = 1000$, noise 5%, BIC regularizator).

Table 1: Synthetic data: performance of the proposed PSC method vs. the baseline, by varying the α and η parameters, on the Erdös-Rény social graph ($|\mathbb{O}| = 1000$, noise 5%, BIC regularizator).

		$\alpha = 0.1$				$\alpha = 0.25$			
		$\eta=0$	$\eta=1$	$\eta=3$	$\eta=5$	$\eta=0$	$\eta=1$	$\eta=3$	$\eta=5$
accuracy	PSC	0.932	0.932	0.933	0.934	0.933	0.933	0.933	0.934
	B	0.765	0.767	0.786	0.793	0.773	0.774	0.791	0.799
NMI		0.67	0.669	0.674	0.677	0.671	0.671	0.675	0.679

Table 2: Synthetic data: performance of the proposed PSC method vs. the baseline, by varying the α and η parameters, on the power-law $\delta = 0.05$ social graph ($|\mathbb{O}| = 1000$, noise 5%, BIC regularizator).

		$\alpha = 0.1$				$\alpha = 0.25$			
		$\eta=0$	$\eta=1$	$\eta=3$	$\eta=5$	$\eta=0$	$\eta=1$	$\eta=3$	$\eta=5$
accuracy	PSC	0.979	0.979	0.979	0.98	0.979	0.978	0.979	0.98
	B	0.938	0.938	0.942	0.944	0.938	0.938	0.941	0.945
NMI		0.563	0.563	0.563	0.563	0.563	0.563	0.563	0.563

Table 3: Synthetic data: performance of the proposed PSC method vs. the baseline, by varying the α and η parameters, on the power-law $\delta = 0.1$ social graph ($|\mathbb{O}| = 1000$, noise 5%, BIC regularizator).

		$\alpha = 0.1$				$\alpha = 0.25$			
		$\eta=0$	$\eta=1$	$\eta=3$	$\eta=5$	$\eta=0$	$\eta=1$	$\eta=3$	$\eta=5$
accuracy	PSC	0.966	0.965	0.967	0.968	0.965	0.965	0.967	0.968
	B	0.882	0.882	0.888	0.892	0.883	0.882	0.887	0.893
NMI		0.63	0.63	0.631	0.631	0.63	0.63	0.63	0.63

$e_+, e_- \in (0, 1)$ of false-positive and false-negative observations in each item, respectively, i.e., the probability of adding or removing from $\mathbb{O}$ a certain triple $\langle v, \phi, t \rangle$, regardless of how it was sampled from the underlying causal process. Note that these two sources of noise aim at modeling imperfect regularities in the causal phenomena in terms of either false positive/negative observations or noise in the orderings among the events. We ultimately generate our noisy datasets with a probability of a noisy entry of either 5% or 10% per entry, with any noisy entry having uniform probability of being either a false positive or a false negative.

Given the above settings, we sample observation sets at different sizes, i.e., $|\mathbb{O}| = 500$, $|\mathbb{O}| = 1000$, and $|\mathbb{O}| = 5000$. The observations are sampled across the $k = 10$ groups in such a way that the size of each group is guaranteed to be within $\frac{|\mathbb{O}|}{k} \times (1 \pm [0, 0.5])$, e.g., for size $|\mathbb{O}| = 500$, every cluster will have between 25 and 75 traces.

Assessment criteria. We use the following metrics:

- *Causal topology.* We assess how well the causal structure ultimately inferred by our method reflects the generative model. For this assessment we resort to the traditional Hamming distance. Specifically, we first build the union graph of the causal DAGs of each cluster to obtain the graph of all the true causal claims. We do this for both the generative DAGs (i.e., the ground-truth ones) and the inferred ones (i.e., the ones yielded by the proposed method or the baseline). Then, we compute the Hamming distance from the ground-truth and the inferred structures, i.e., we count the minimum number of substitutions required to remove any inconsistency from the output topologies, when compared to the ground-truth ones. We ultimately report the performance in terms of *accuracy* $= \frac{(TP+TN)}{(TP+TN+FP+FN)}$, with TP and FP being the arcs recognized as true and false positives, respectively, and TN and FN being the arcs recognized as true and false negatives, respectively.
- *Partitioning.* We also estimate how well our partitioning step can effectively group users involved in the same causal process. To this end, we measure the similarity between the clusters identified by our method and the ground-truth clusters by means of the well-established Normalized Mutual Information (NMI) measure [18].

Results. We compare the performance of the proposed two-step Probabilistic Social Causation method (for short, PSC) against a baseline B that only performs the first one of the two steps of the PSC algorithm (i.e., only the grouping step as described in Algorithm 2), and reconstruct a DAG $G_D(\mathcal{D})$ from the prima-facie graph $G(\mathcal{D})$ of every group $\mathcal{D} \in \mathbf{D}^*$ outputted by that step, without learning the minimal causal topology. All results are averaged over the 100 data-generation runs performed, and, unless otherwise specified, they refer to $K = 100$ (for the proposed PSC method).

First, we evaluate the execution time of the proposed PSC method by varying the algorithm parameters (i.e., α and η), and the type of social graph underlying the generated data. The results of this experiment are reported in Figure 2. As depicted in the figure, the running time of the proposed method is in the order of a few milliseconds. Also, the different values of α and η, as well as the form of the social graph, do not seem to have a significant impact on the execution time.

Shifting the attention to effectiveness, Tables 1–3 report the performance of the proposed PSC method vs. the B baseline, by varying the algorithm parameters (α and η), on the various social graphs. In all cases the *accuracy* of the proposed PSC is evidently higher than the one of the baseline. The performance is rather independent of the algorithm parameters or the social graph. In terms of NMI (which is the same for both PSC and B, as it concerns the first step of the proposed method that is common to PSC and B), the performance is in the range $[0.56, 0.68]$, which is a fair result considering the difficulty of the subtask of recognizing the exact ground-truth cluster structure.

Table 4 reports on the performance by varying the number of sampled observations. As expected, the trends follow the common intuition: the performance of both PSC and B decreases as the number of observations increases. However, a major result to remark here is that the advantage of the proposed PSC over B gets higher

Table 4: Synthetic data: performance of the proposed PSC method vs. the baseline, by varying the size $|\mathbb{O}|$ of input observations ($\alpha = 0.1$, $\eta = 1$, noise 5%, BIC regularizator).

		Erdös-Rény			Power-law $\delta = 0.05$			Power-law $\delta = 0.1$		
		$\|\mathbb{O}\| = 500$	$\|\mathbb{O}\| = 1000$	$\|\mathbb{O}\| = 5000$	$\|\mathbb{O}\| = 500$	$\|\mathbb{O}\| = 1000$	$\|\mathbb{O}\| = 5000$	$\|\mathbb{O}\| = 500$	$\|\mathbb{O}\| = 1000$	$\|\mathbb{O}\| = 5000$
accuracy	PSC	0.939	0.932	0.909	0.983	0.979	0.963	0.974	0.965	0.938
	B	0.815	0.767	0.585	0.95	0.938	0.913	0.904	0.882	0.835
NMI		0.669	0.662	0.662	0.563	0.567	0.571	0.630	0.636	0.642

Table 5: Synthetic data: performance of the proposed PSC method vs. the baseline, by varying the noise level ($\alpha = 0.1$, $\eta = 1$, $|\mathbb{O}| = 1000$, BIC regularizator).

	$\alpha = 0.1$								$\alpha = 0.25$							
	$\eta = 0$		$\eta = 1$		$\eta = 3$		$\eta = 5$		$\eta = 0$		$\eta = 1$		$\eta = 3$		$\eta = 5$	
	BIC	AIC	BIC	AIC	BIC	AIC	BIC	AIC	BIC	AIC	BIC	AIC	BIC	AIC	BIC	AIC
accuracy	0.979	0.971	0.979	0.971	0.979	0.972	0.98	0.973	0.979	0.971	0.978	0.971	0.979	0.972	0.98	0.973

Table 6: Synthetic data: performance of the proposed PSC method with varying the regularizator, i.e., BIC vs. AIC ($|\mathbb{O}| = 1000$, noise 5%, power-law $\delta = 0.05$ social graph).

		Erdös-Rény			Power-law $\delta = 0.05$			Power-law $\delta = 0.1$		
		no noise	noise 5%	noise 10%	no noise	noise 5%	noise 10%	no noise	noise 5%	noise 10%
accuracy	PSC	0.936	0.932	0.93	0.98	0.979	0.978	0.967	0.965	0.964
	B	0.839	0.767	0.713	0.941	0.938	0.936	0.887	0.882	0.878
NMI		0.686	0.669	0.661	0.563	0.563	0.563	0.63	0.63	0.63

with the increasing observations. This attests to the effectiveness of our method even for large observation-set sizes.

The last experiments we focus on are on the impact of the regularizator and the noise level on the performance of PSC. The results of these experiments are shown in Table 5 and 6, respectively. As far as the former, BIC is recognized as slightly more accurate than AIC. As for the noise level, we observe only a slight decrease of the performance of the proposed PSC as the noise level increases, which attests to the high robustness of the proposed method.

6.2 Real data

We also experiment with three real-world datasets, whose main characteristics are summarized in Table 7.

Table 7: Characteristics of real data: number of observations ($|\mathbb{O}|$); number of propagations/DAGs ($|\mathbb{D}|$); nodes ($|V|$) and arcs ($|A|$) of the social graph G; min, max, and avg number of nodes in a DAG of $\mathbb{D}$ (n_{min}, n_{max}, n_{avg}); min, max, and avg number of arcs in a DAG of $\mathbb{D}$ (m_{min}, m_{max}, m_{avg}).

	$\|\mathbb{O}\|$	$\|\mathbb{D}\|$	$\|V\|$	$\|A\|$	n_{min}	n_{max}	n_{avg}	m_{min}	m_{max}	m_{avg}
Last.fm	1 208 640	51 495	1 372	14 708	6	472	24	5	2 704	39
Twitter	580 141	8 888	28 185	1 636 451	12	13 547	66	11	240 153	347
Flixster	6 529 012	11 659	29 357	425 228	14	16 129	561	13	85 165	1 561

Last.fm (`www.last.fm`). Lastfm is a music website, where users listen to their favorite tracks and communicate with each other. The dataset was created starting from the *HetRec 2011 Workshop* dataset available at www.grouplens.org/node/462, and enriching it by crawling. The graph G corresponds to the friendship network of the service. The entities in $\mathcal{E}$ are the songs listened to by the users. An observation $\langle u, \phi, t\rangle \in \mathbb{O}$ means that the first time that the user u listens to the song ϕ occurs at time t.

Twitter (`twitter.com`). We obtained the dataset by crawling the public timeline of the popular online microblogging service. The nodes of the graph G are the Twitter users, while each arc (u, v) expresses the fact that v is a follower of u. The entities in $\mathcal{E}$ correspond to URLs, while an observation $\langle u, \phi, t\rangle \in \mathbb{O}$ means that the user u (re-)tweets (for the first time) the URL ϕ at time t.

Flixster (`www.flixster.com`). Flixster is a social movie site where people can meet each other based on tastes in movies. The graph G corresponds to the social network underlying the site. The entities in $\mathcal{E}$ are movies, and an observation $\langle u, \phi, t\rangle$ is included in $\mathbb{O}$ when the user u rates for the first time the movie ϕ with the rating happening at time t.

As real data comes with no ground-truth, here we resort to the well-established *spread-prediction* task to assess the effectiveness of our method. This task aims at predicting the expected number of nodes that eventually get activated due to an information-propagation process initiated in some nodes [23]. Specifically, in our experiments we consider the well-established propagation model defined by Goyal *et al.* [23], which takes as input a graph (representing relationships among some users) and a database of propagations (representing actions taken by those users), and learns a model that is able to predict the influence spread. In our context we (randomly) split our input propagations into training set and test set (70% vs. 30%), and use the training set to learn the Goyal *et al.*'s model. As a graph, we consider both the ultimate causal structure computed by our PSC method, and the whole input social graph. The latter constitutes a baseline to evaluate the effectiveness of PSC: the rationale is that providing the Goyal *et al.*'s model with a graph corresponding to the causal structure recognized by our method, instead of the whole social network, is expected to improve the performance of the spread-prediction task, as it will not burden the model with noisy relationships. Once the Goyal *et al.*'s model has been learned, we use it to predict spread, and measure the accuracy of the experiment in terms of *mean squared error* (MSE) between the predicted spreads and the real ones exhibited by the test set.

The results of this experiment are shown in Figure 3, where for our PSC method we set $\alpha = 0.20$, $\eta = 5$, and $K = 500$, while for both PSC and the baseline we perform spread prediction by running 10,000 Monte Carlo simulations (as suggested in [23]). The figure shows that, on all datasets, the error is consistently smaller when our method is used compared to when the original social network is given as input to the spread-prediction algorithm. This finding hints at a promising capability achievable by the proposed method in extracting the real causal relations within the social network.

7 CONCLUSIONS

In this paper we tackle the problem of deriving causal DAGs that are well-representative of the social-influence dynamics underlying an input database of propagation traces. We devise a principled two-step methodology that is based on Suppes' probabilistic-causation theory. The first step of the methodology aims at partitioning the input set of propagations, mainly to get rid of the Simpson's paradox,

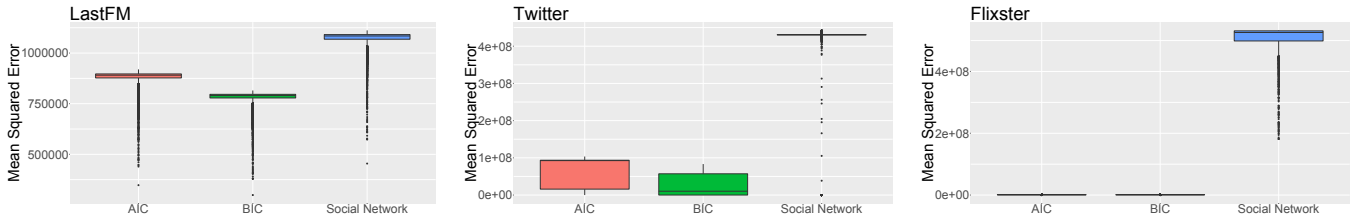


Figure 3: Real-data: spread-prediction performance of the proposed PSC method (equipped with BIC or AIC regularizator) vs. a baseline that considers the whole social graph ($\alpha = 0.2$, $\eta = 5$).

while the second step derives the ultimate minimal causal topology via constrained MLE. Experiments on synthetic data attest to the high accuracy of the proposed method in detecting ground-truth causal structures, while experiments on real data show that our method performs well in a task of spread prediction.

REFERENCES

[1] P. K. Agarwal, L. J. Guibas, H. Edelsbrunner, J. Erickson, M. Isard, S. Har-Peled, J. Hershberger, C. Jensen, L. Kavraki, P. Koehl, et al. Algorithmic issues in modeling motion. *ACM Computing Surveys (CSUR)*, 34(4):550–572, 2002.

[2] C. C. Aggarwal and J. Han. *Frequent Pattern Mining*. Springer, 2014.

[3] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB*, pages 487–499, 1994.

[4] H. Akaike. A new look at the statistical model identification. *IEEE Trans. Aut. Contr.*, 19(6):716–723, 1974.

[5] J. Aldrich. Correlations genuine and spurious in pearson and yule. *Statistical Science*, pages 364–376, 1995.

[6] A. Anagnostopoulos, R. Kumar, and M. Mahdian. Influence and correlation in social networks. In *KDD*, pages 7–15, 2008.

[7] S. Aral, L. Muchnik, and A. Sundararajan. Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks. *PNAS*, 106(51):21544–21549, 2009.

[8] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts. Everyone's an influencer: quantifying influence on twitter. In *WSDM*, pages 65–74, 2011.

[9] F. Bonchi. Influence propagation in social networks: A data mining perspective. *IEEE Intelligent Informatics Bulletin, Vol.12 No.1: 8-16, December 2011.*

[10] F. Bonchi, S. Hajian, B. Mishra, and D. Ramazzotti. Exposing the probabilistic causal structure of discrimination. *IJDSA*, 3(1):1–21, 2017.

[11] G. Caravagna, A. Graudenzi, D. Ramazzotti, R. Sanz-Pamplona, L. De Sano, G. Mauri, V. Moreno, M. Antoniotti, and B. Mishra. Algorithmic methods to infer the evolutionary trajectories in cancer progression. *PNAS*, 113(28):E4025–E4034, 2016.

[12] C. Castillo, M. Mendoza, and B. Poblete. Information credibility on Twitter. In *WWW*, pages 675–684, 2011.

[13] M. Cha, A. Mislove, and P. K. Gummadi. A measurement-driven analysis of information propagation in the Flickr social network. In *WWW*, pages 721–730, 2009.

[14] D. M. Chickering, D. Heckerman, and C. Meek. Large-sample learning of bayesian networks is np-hard. *JMLR*, 5:1287–1330, 2004.

[15] S. Cook. The Complexity of Theorem-proving Procedures. In *STOC*, pages 151–158, 1971.

[16] G. Cormode, H. Karloff, and A. Wirth. Set cover algorithms for very large datasets. In *CIKM*, pages 479–488, 2010.

[17] D. J. Crandall, D. Cosley, D. P. Huttenlocher, J. M. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *KDD*, pages 160–168, 2008.

[18] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas. Comparing community structure identification. *JSTAT*, 2005(09):P09008, 2005.

[19] P. Domingos and M. Richardson. Mining the network value of customers. In *KDD*, pages 57–66, 2001.

[20] T. L. Fond and J. Neville. Randomization tests for distinguishing social influence and homophily effects. In *WWW*, pages 601–610, 2010.

[21] G. Gao, B. Mishra, and D. Ramazzotti. Efficient simulation of financial stress testing scenarios with suppes-bayes causal networks. *Procedia Computer Science*, 108:272–284, 2017.

[22] J. Golbeck and J. Hendler. Inferring binary trust relationships in web-based social networks. *ACM Trans. Internet Technol.*, 6(4):497–529, 2006.

[23] A. Goyal, F. Bonchi, and L. V. Lakshmanan. A data-based approach to social influence maximization. *PVLDB*, 5(1):73–84, 2011.

[24] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. Learning influence probabilities in social networks. In *WSDM*, pages 241–250, 2010.

[25] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *WWW*, 2004.

[26] M. Gupte, P. Shankar, J. Li, S. Muthukrishnan, and L. Iftode. Finding hierarchy in directed online social networks. In *WWW*, pages 557–566, 2011.

[27] M. A. Hasan and M. J. Zaki. Musk: Uniform sampling of k maximal patterns. In *SDM*, pages 650–661, 2009.

[28] S. Hill, F. Provost, and C. Volinsky. Network-based marketing: Identifying likely adopters via consumer networks. *Statistical Science*, 21(2):256–276, 2006.

[29] C. Hitchcock. Probabilistic causation. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Winter 2012 edition, 2012.

[30] D. Ienco, F. Bonchi, and C. Castillo. The meme ranking problem: Maximizing microblogging virality. In *SIASP IEEE ICDM Work.*, 2010.

[31] D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *KDD*, pages 137–146, 2003.

[32] S. Kleinberg. *An algorithmic enquiry concerning causality*. PhD thesis, New York University, 2010.

[33] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[34] K. Kutzkov, A. Bifet, F. Bonchi, and A. Gionis. STRIP: stream learning of influence probabilities. In *KDD*, pages 275–283, 2013.

[35] J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. *TWEB*, 1(1), 2007.

[36] J. Leskovec, A. Singh, and J. M. Kleinberg. Patterns of influence in a recommendation network. In *PAKDD*, pages 380–389, 2006.

[37] L. O. Loohuis, G. Caravagna, A. Graudenzi, D. Ramazzotti, G. Mauri, M. Antoniotti, and B. Mishra. Inferring tree causal models of cancer progression with probability raising. *PloS ONE*, 9(10):e108358, 2014.

[38] S. Moens and B. Goethals. Randomly Sampling Maximal Itemsets. In *KDD IDEA Work.*, pages 79–86, 2013.

[39] M. E. Newman. Mixing patterns in networks. *Physical Review E*, 67(2):026126, 2003.

[40] J. Pearl. *Causality*. Cambridge university press, 2009.

[41] D. Ramazzotti, G. Caravagna, L. Olde Loohuis, A. Graudenzi, I. Korsunsky, G. Mauri, M. Antoniotti, and B. Mishra. CAPRI: efficient inference of cancer progression models from cross-sectional data. *Bioinformatics*, 31(18):3016–3026, 2015.

[42] D. M. Romero, B. Meeder, and J. M. Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on Twitter. In *WWW*, pages 695–704, 2011.

[43] K. Saito, R. Nakano, and M. Kimura. Prediction of information diffusion probabilities for independent cascade model. In *KES*, pages 67–75, 2008.

[44] J. J. Samper, P. A. Castillo, L. Araujo, and J. J. M. Guervós. Nectarss, an rss feed ranking system that implicitly learns user preferences. *CoRR*, abs/cs/0610019, 2006.

[45] G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

[46] C. R. Shalizi and A. C. Thomas. Homophily and contagion are generically confounded in observational social network studies. *Sociological methods & research*, 40(2):211–239, 2011.

[47] A. Sharma and D. Cosley. Distinguishing between personal preferences and social influence in online activity feeds. In *CSCW*, pages 1091–1103, 2016.

[48] X. Song, Y. Chi, K. Hino, and B. L. Tseng. Information flow modeling based on diffusion rate for prediction and ranking. In *WWW*, pages 191–200, 2007.

[49] X. Song, B. L. Tseng, C.-Y. Lin, and M.-T. Sun. Personalized recommendation driven by information flow. In *SIGIR*, pages 509–516, 2006.

[50] P. Suppes. *A Probabilistic Theory of Causality*. North-Holland Publishing Company, 1970.

[51] M. Taherian, M. Amini, and R. Jalili. Trust inference in web-based social networks using resistive networks. In *ICIW*, pages 233–238, 2008.

[52] N. Tatti. Faster way to agony: Discovering hierarchies in directed graphs. In *ECML PKDD*, pages 163–178, 2014.

[53] N. Tatti. Tiers for peers: a practical algorithm for discovering hierarchy in weighted networks. *DAMI*, 31(3):702–738, 2017.

[54] C. H. Wagner. Simpson's paradox in real life. *The American Statistician*, 36(1):46–48, 1982.

[55] J. Weng, E.-P. Lim, J. Jiang, and Q. He. TwitterRank: finding topic-sensitive influential twitterers. In *WSDM*, pages 261–270, 2010.

[56] C.-N. Ziegler and G. Lausen. Propagation models for trust and distrust in social networks. *Information Systems Frontiers*, 7(4-5):337–358, 2005.