Supplementary Materials for Inferring Causal Direction in Relational Data

1 Proofs

1.1 Proof of Proposition 2

Proposition 2. *Assume that the true generative process is* $\mathbf{y} = \beta \cdot D^{-1}A\mathbf{x} + \epsilon$ *for some constant* β*, where* ϵ *is a vector with the noise terms. Moreover, assume that assumptions A1-A5 hold and* X *and* Y *are scaled to mean 0. Then the following holds:*

$$\rho^2(\mathbf{x}', \mathbf{y}) > \rho^2(\mathbf{y}', \mathbf{x}) \Leftrightarrow$$
$$\frac{Var(AD^{-1}A\mathbf{x}) + Var(A\epsilon)}{Var(D^{-1}A\mathbf{x}) + Var(\epsilon)} > \frac{Var(A\mathbf{x})}{Var(\mathbf{x})}.$$

Proof.

$$\begin{aligned}\rho(\mathbf{x}', \mathbf{y}) &= \rho(A\mathbf{x}, D^{-1}A\mathbf{x} + \epsilon) \quad (1)\\ &= \frac{Cov(A\mathbf{x}, D^{-1}A\mathbf{x}) + Cov(A\mathbf{x}, \epsilon)}{Var(A\mathbf{x})\big(Var(D^{-1}A\mathbf{x}) + Var(\epsilon)\big)}\\ &= \frac{Cov(A\mathbf{x}, D^{-1}A\mathbf{x})}{Var(A\mathbf{x})\big(Var(D^{-1}A\mathbf{x}) + Var(\epsilon)\big)} \quad (2)\end{aligned}$$

$$\begin{aligned}\rho(\mathbf{y}', \mathbf{x}) &= \rho(AD^{-1}A\mathbf{x} + D^{-1}A\epsilon, \mathbf{x}) \quad (3)\\ &= \frac{Cov(AD^{-1}A\mathbf{x}, \mathbf{x}) + Cov(\mathbf{x}, D^{-1}A\epsilon)}{Var(\mathbf{x})\big(Var(AD^{-1}A\mathbf{x}) + Var(D^{-1}A\epsilon)\big)}\\ &= \frac{Cov(AD^{-1}A\mathbf{x}, \mathbf{x})}{Var(\mathbf{x})\big(Var(AD^{-1}A\mathbf{x}) + Var(D^{-1}A\epsilon)\big)} \quad (4)\end{aligned}$$

The covariance, given that the mean of X and Y is 0, is equal to the inner product of the variables.

$$\begin{aligned}Cov(A\mathbf{x}, D^{-1}A\mathbf{x}) &= \langle A\mathbf{x}, D^{-1}A\mathbf{x}\rangle \quad (5)\\ &= \mathbf{x}^\top A^\top D^{-1}A\mathbf{x} \quad (6)\\ &= \mathbf{x}^\top AD^{-1}A\mathbf{x} \quad (7)\\ Cov(AD^{-1}A\mathbf{x}, \mathbf{x}) &= \langle AD^{-1}A\mathbf{x}, \mathbf{x}\rangle \quad (8)\\ &= \mathbf{x}^\top AD^{-1}A\mathbf{x} \quad (9)\end{aligned}$$

Therefore, for the square of the correlations we can write:

$$\begin{aligned}&\rho(\mathbf{x}', \mathbf{y}) > \rho(\mathbf{y}', \mathbf{x}) \Leftrightarrow\\ &\frac{1}{Var(A\mathbf{x})\big(Var(D^{-1}A\mathbf{x}) + Var(\epsilon)\big)} >\\ &\frac{1}{Var(\mathbf{x})\big(Var(AD^{-1}A\mathbf{x}) + Var(A\epsilon)\big)} \Leftrightarrow\\ &\frac{Var(AD^{-1}A\mathbf{x}) + Var(A\epsilon}{Var(D^{-1}A\mathbf{x}) + Var(\epsilon)} > \frac{Var(A\mathbf{x})}{Var(\mathbf{x})}\end{aligned}$$

□

1.2 Proof of Proposition 3

Proof. Assume that the true generative structure is:

$$\begin{aligned}\mathbf{y} &\sim D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}}\\ \mathbf{x} &\sim D^{-1}A\mathbf{z} + \epsilon_{\mathbf{x}}\end{aligned}$$

The covariance between $A\mathbf{x}$ and $A\mathbf{y}$ is then given by

$$\begin{aligned}&Cov(Ax, Ay)\\ &= Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{y}}, AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{x}})\\ &= Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{y}}, AD^{-1}A\mathbf{z}) +\\ &\quad Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{y}}, A\epsilon_{\mathbf{x}})\\ &= Cov(AD^{-1}A\mathbf{z}, AD^{-1}A\mathbf{z}) + Cov(AD^{-1}A\mathbf{z}, A\epsilon_{\mathbf{x}})\\ &= Cov(AD^{-1}A\mathbf{z}, AD^{-1}A\mathbf{z})\end{aligned}$$

The covariance between $A\mathbf{x}$ and $\mathbf{y}$, is given by:

$$\begin{aligned}&Cov(Ax, y)\\ &= Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{x}}, D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}})\\ &= Cov(AD^{-1}A\mathbf{z}, D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}}) +\\ &\quad Cov(A\epsilon_{\mathbf{x}}, D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}})\\ &= Cov(AD^{-1}A\mathbf{z}, D^{-1}A\mathbf{z}) + Cov(D^{-1}A\mathbf{z}, \epsilon_{\mathbf{y}})\\ &= Cov(AD^{-1}A\mathbf{z}, D^{-1}A\mathbf{z})\\ &\leq Cov(AD^{-1}A\mathbf{z}, AD^{-1}A\mathbf{z})\end{aligned}$$

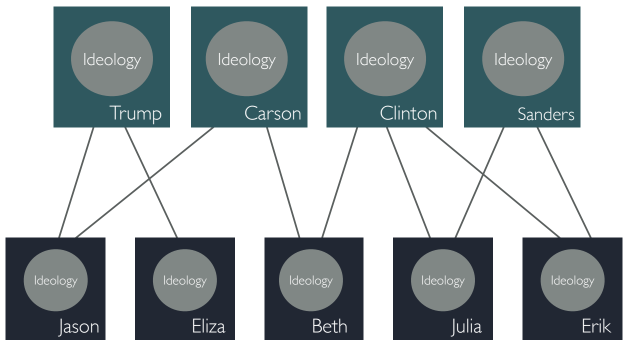


Figure 1: Schematic representation of dataset consisting of presidential candidates, voters. The upper row consists of four candidates and the bottom row of five voters. An undirected edge indicates that a voter would consider voting for the presidential candidate.

$AD^{-1}A\mathbf{z}$ and $D^{-1}A\mathbf{z}$ are bounded by the size of the intersection between the set of a node's immediate neighbors and the set of its two-hop neighbors, since we have assumed $\mathbf{z}$ are marginally independent by construction. Each pair of one hop and two hop neighborhoods will diverge for at least the degree of the node for each node, since the two hop walk beginning from node i will return to that node an equal number of its degree, which implies the final inequality. □

2 Confounding Experiments

In addition to the experiments presented in the main text for determining the direction of dependence, we also empirically evaluated the efficacy of confounding detection. We replicated the experimental settings described in sections 6.1 and 6.2, except in this case both $\mathbf{x}$ and $\mathbf{y}$ are drawn using a direct dependence on a third variable $\mathbf{z}$. We then determined confounding by testing whether the covariance between $A\mathbf{x}$ and $A\mathbf{y}$ was greater than both $Cov(A\mathbf{x}, \mathbf{y})$ and $Cov(A\mathbf{y}, \mathbf{x})$. The results for regular graphs can be seen in Figure 2. The confounding test is very robust across all of these dimensions. There is only a slight decrease in accuracy in even the most adversarial settings of large degree and high-noise generating scenarios. Figure 3 shows performance as the noise level is increased, across three non-regular graph generation algorithms. For two of the three graph generation procedures (Watts-Strogatz and Barabasi-Albert), there is near perfect performance. The Erdos-Renyi graph performance is considerably poorer. We conjecture that this is due to the high connectivity (each node is connected to approximately 20% of its neighbors), which greatly reduces the effective sample size. We plan on investigating methods to address causal inference on high-connectivity graphs as future work.

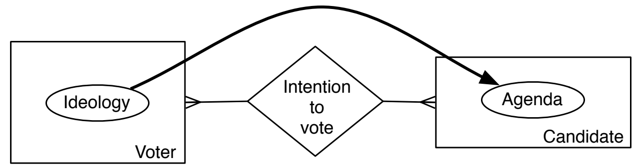


Figure 4: Example relational model that consists of the underlying relational schema (ER diagram) and a relational dependency (depicted with an arrow).

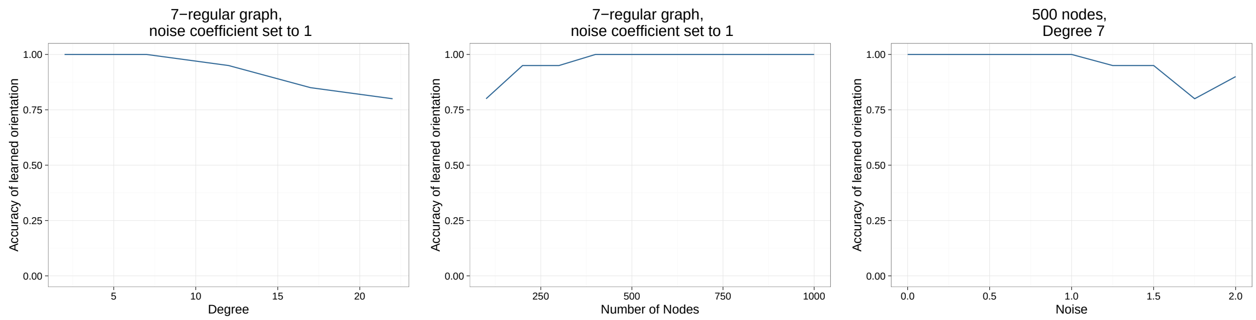
3 Extension to Multi-Relational Domains

In the main paper, we focused on single-entity single-relationship networks (using the example of a social network where people are friends with other people). However, our results can generalize to the (more expressive) fully relational case. Relational domains consist of multiple types of entities that interact with each other through multiple types of relationships. For example, consider a domain that consists of two types of entities, presidential candidates and voters, and a single type of relationship, which candidate will a voter vote for.

In this work, we adopt the framework of probabilistic relational models (PRMs) [Friedman *et al.*, 1999; Heckerman *et al.*, 2007] to represent relational domains and reason over them.. In what follows, we introduce the basic concepts and notation for PRMs.

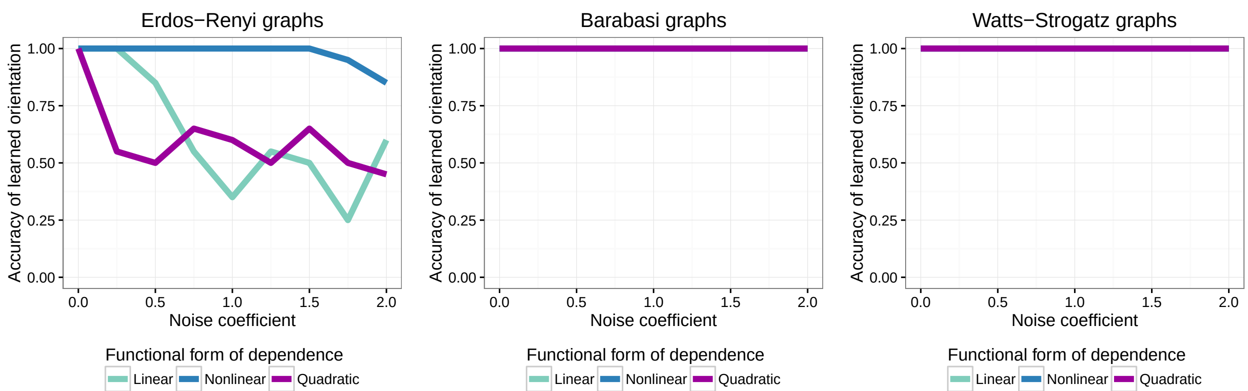
A *relational schema* specifies the set of entity classes $(E_1, \ldots, E_n)$ and the set of relationship classes $(R_1, \ldots, R_m)$ of a domain. Relationship classes are ordered tuples of entity classes. Every entity and relationship class is associated with a set of attributes and a set of *reference slots*. We use the notation $C.A$ to denote that A is an attribute of class C. A reference slot for a relationship class corresponds to an entity that participates in the relationship. Conversely, a reference slot for an entity, is a relationship it participates in. The notation $C.\rho$ is used to denote the reference slot ρ of class C. Reference slots can be combined to form a *slot chain*, $\tau = \rho_1 \cdots \rho_k$. A relational schema can be graphically represented with an Entity-Relationship (ER) diagram.

Figure 4 shows an example ER diagram for the domain of voters and candidates. The relational schema for that domain consists of two entity classes, Voter and $\mathit{Candidate}$, and one relationship class, $\mathit{IntentionToVote}$. The entity class Voter has one attribute, $\mathit{Ideology}$, and the entity class $\mathit{Candidate}$ has one attribute, Agenda. The entity class Voter has a reference slot $\mathit{Voter}.\mathit{IntentionToVote}$ and the entity class $\mathit{Candidate}$ has a reference slot $\mathit{Candidate}.\mathit{IntentionToVote}$. The relationship class $\mathit{IntentionToVote}$ has two reference slots $\mathit{IntentionToVote}.\mathit{Voter}$ and $\mathit{IntentionToVote}.\mathit{Candidate}$. Crow's feet notation



(a) (b) (c)

Figure 2: Accuracy detecting confounding for regular graphs for varying degree (3a), size of network (3b), and noise coefficient (3c).



(a) (b) (c)

Figure 3: Orientation accuracy for different types of networks graphs for varying noise.

is used to denote that the $IntentionToVote$ relationship is many-to-many, i.e., a voter might intend to vote for multiple candidates and a candidate can be voted by multiple voters.

A *relational skeleton* is a partial instantiation of a relational schema that specifies the set of entity and relationship instances that exist in the domain. Figure 1 depicts an example relational skeleton for the domain described in Figure 4. It consists of five $Voter$ instances and four $Candidate$ instances.

A *probabilistic relational model* consists of a dependence structure, $\mathcal{D}$, and a set of parameters associated with it, $\theta_{\mathcal{D}}$. The dependence structure specifies for every attribute $C.X$ of the relational schema a set of parents, $Pa(C.X)$. The parents of an attribute $C.A$ are other attributes, either in the same class C (for example $C.A'$), or on a different class, reachable through some slot chain τ (for example $C.\tau.A''$). In our example, a dependence structure could be $Pa(Candidate.Agenda) = Candidate.IntentionToVote.Voter.Ideology$ and $Pa(Voter.Ideology) = \emptyset$, shown in Figure 4. This means that the agenda of a candidate is affected by the ideology of his/her potential voters, while nothing affects a voter's ideology. Finally, a *ground graph* can be constructed by applying the dependencies of a probabilistic relational model to a relational skeleton.

We refer to variables of the form $C.X$ as *propositional variables* and to variables of the form $C.\tau.X$ as *relational variables*. It is worth noting that relational variables, when instantiated, might produce a set of values. For example, $Candidate.Agenda$ is a propositional variable and $Candidate.IntentionToVote.Voter.Ideology$ is a relational variable. The instantiation of this relational variable for a given candidate consists of the set of ideology scores for all voters that would vote for that candidate. In this work, we are concerned with detecting dependence between a propositional variable $C.X$ and a relational variable $C.\tau.X$.

The results presented in the main text can be extended for the fully relational case. For the case of single-entity single-relationship networks, we required regular graphs, i.e., graphs where every node has the same degree. Similar regularity assumptions can be made for the case of fully relational graphs.

References

Nir Friedman, Lise Getoor, Daphne Koller, and Avi Pfeffer. Learning probabilistic relational models. In *International Joint Conference on Artificial Intelligence*, 1999.

David Heckerman, Chris Meek, and Daphne Koller. Probabilistic entity-relationship models, prms, and plate models. *Introduction to statistical relational learning*, pages 201–238, 2007.