

Evaluation of Induced Expert Knowledge in Causal Structure Learning by NOTEARS

Jawad Chowdhury, Rezaur Rashid and Gabriel Terejanu

Dept. of Computer Science, University of North Carolina at Charlotte, Charlotte, NC, U.S.A.

Keywords: Causality, Structured Prediction and Learning, Supervised Deep Learning, Optimization for Neural Networks.

Abstract: Causal modeling provides us with powerful counterfactual reasoning and interventional mechanism to generate predictions and reason under various what-if scenarios. However, causal discovery using observation data remains a nontrivial task due to unobserved confounding factors, finite sampling, and changes in the data distribution. These can lead to spurious cause-effect relationships. To mitigate these challenges in practice, researchers augment causal learning with known causal relations. The goal of the paper is to study the impact of expert knowledge on causal relations in the form of additional constraints used in the formulation of the nonparametric NOTEARS. We provide a comprehensive set of comparative analyses of biasing the model using different types of knowledge. We found that (i) knowledge that correct the mistakes of the NOTEARS model can lead to statistically significant improvements, (ii) constraints on active edges have a larger positive impact on causal discovery than inactive edges, and surprisingly, (iii) the induced knowledge does not correct on average more incorrect active and/or inactive edges than expected. We also demonstrate the behavior of the model and the effectiveness of domain knowledge on a real-world dataset.

1 INTRODUCTION

Machine learning models have been breaking records in terms of achieving higher predictive accuracy. Nevertheless, out-of-distribution (OOD) generalization remains a challenge. One solution is adopting causal structures (Lake et al., 2017) to constrain the models and remove spurious correlations. The underlying causal knowledge of the problem of interest can significantly help with domain adaptability and OOD generalization (Magliacane et al., 2017). Furthermore, causal models go beyond the capability of correlation-based models to produce predictions. They provide us with the powerful counterfactual reasoning and interventional mechanism to reason under various what-if scenarios (Pearl, 2009).

Two of the most prominent approaches in observational causal discovery are constraint-based and score-based methods (Spirtes et al., 2000; Pearl and Verma, 1995; Colombo et al., 2012; Chickering, 2002; Ramsey et al., 2017). Although these methods are quite robust if the underlying assumptions are true, they are computationally expensive and their computational complexity increases with the number of system variables due to the combinatorial nature of the DAG constraint. NOTEARS (Zheng et al., 2018) tackles this problem with an algebraic char-

acterization of acyclicity which reduces the combinatorial problem to a continuous constrained optimization. Different approaches (Yu et al., 2019; Lachapelle et al., 2019; Ng et al., 2019; Zheng et al., 2020) have been proposed as the nonlinear or nonparametric extensions of this linear continuous optimization, which provides flexibility in modeling different causal mechanisms.

Learning the causal structure purely based on observational data is not a trivial task due to various limitations such as finite sampling, unobserved confounding factors, selection bias, and measurement errors (Cooper, 1995; Elkan, 2001; Zadrozny, 2004). These can result in spurious cause-effect relationships. To mitigate these challenges in practice, researchers augment causal learning with prior causal relations as featured in software packages such as CausalNex¹, causal-learn², bnlearn (Scutari, 2009), gCastle (Zhang et al., 2021), and DoWhy (Sharma and Kiciman, 2020). Heindorf et al. (Heindorf et al., 2020) in their work attempts to construct the first large scale open domain causality graph that can be included in the existing knowledge bases. The work further analyze and demonstrates the benefits of large

¹<https://github.com/quantumblacklabs/causalnex>

²<https://github.com/cmu-phil/causal-learn>

scale causality graph in causal reasoning. Given a partial ancestral graph (PAG), representing the qualitative knowledge of the causal structure, Jaber et al. (Jaber et al., 2018) in their study compute the interventional distribution from observational data. Combining expert knowledge with structural learning further constrains the search space minimizing the number of spurious mechanisms (Wei et al., 2020) and researchers often leverage these background knowledge by exploiting them as additional constraints for knowledge-enhanced event causality identification (Liu et al., 2021). O’Donnell et al. (O’Donnell et al., 2006) use expert knowledge as prior probabilities in learning Bayesian Network (BN) and Gencoglu and Gruber (Gencoglu and Gruber, 2020) use the linear NOTEARS model to incorporate knowledge to detect how different characteristics of the COVID-19 pandemic are causally related to each other. Different experts’ causal judgments can be aggregated into collective ones (Bradley et al., 2014) and Alrajeh et al. (Alrajeh et al., 2020) in their work, studied how these judgments can be combined to determine effective interventions. An interesting exploration by Andrews et al. (Andrews et al., 2020) defines tiered background knowledge and shows that with this type of background knowledge the FCI algorithm (Spirtes et al., 2000) is sound and complete.

However, understanding how to effectively incorporate and evaluating the impact of induced knowledge is yet to be explored and we believe knowledge regarding this can mitigate some of the challenges of observational causal discovery. Human expertise can play a vital role to assess the learned model in causal structure learning (Bhattacharjya et al., 2021; Li et al., 2021). In practice, human assessment and validation process often take place in an iterative or sequential manner (Holzinger, 2016; Xin et al., 2018; Yang et al., 2019). In structure learning, this is more realistic for a sufficiently large causal network where one can learn, validate, and induce newly formed knowledge-set in the learning process following sequential feedback loops. The goal of this paper is not to create a new causal discovery algorithm but rather to study this iterative interaction between prior causal knowledge from domain experts that takes the form of model constraints and a state-of-the-art causal structure learning algorithm. Wei et al. (Wei et al., 2020) have been the first to augment NOTEARS with additional optimization constraints to satisfy the Karush-Kuhn-Tucker (KKT) optimality conditions and Fang et al. (Fang et al., 2020) in their work leverages the low rank assumption in the context of causal DAG learning by augmented NOTEARS that shows significant improvements. However, none of them have

studied the impact of induced knowledge on causal structure learning by augmenting NOTEARS with the optimization constraints. For completeness, in Section 3, we do provide our formulation of nonparametric NOTEARS (Zheng et al., 2020) with functionality to incorporate causal knowledge in the form of known direct causal and non-causal relations. Nevertheless, in this work, we aim to study the impact of expert causal knowledge on causal structure learning.

The main contributions are summarized as follows. (1) We demonstrate an iterative modeling framework to learn causal relations, impose causal knowledge to constrain the causal graphs, and further evaluate the model’s behavior and performance. (2) We empirically evaluate and demonstrate that: (a) knowledge that corrects model’s mistake can lead to statistically significant improvements, (b) constraints on active edges have a larger positive impact on causal discovery than inactive edges, and (c) the induced knowledge does not correct on average more incorrect active and/or inactive edges than expected. Finally, we illustrate the impact of additional knowledge in causal discovery on a real-world dataset.

This paper is structured as follows: Section 2 introduces the background on causal graphical models (CGMs), score-based structure recovery methods, and a study using the score-based approach formulated as a continuous optimization and its recent nonparametric extension. In Section 3, we present our extension of the nonparametric continuous optimization to incorporate causal knowledge in structure learning and detail the proposed knowledge induction process. Section 4 shows the empirical evaluations and comparative analyses of the impact of expert knowledge on the model’s performance. Finally, in Section 5, we summarize our findings and provide a brief discussion on future work.

2 BACKGROUND

In this section, we review the basic concepts related to causal structure learning and briefly cover a recent score-based continuous causal discovery approach using structural equation models (SEMs).

2.1 Causal Graphical Model (CGM)

A directed acyclic graph (DAG) is a directed graph without any directed cyclic paths (Spirtes et al., 2000). A causal graphical model $CGM(P_X, \mathcal{G})$ can be defined as a pair of a graph \mathcal{G} and an observational distribution P_X over a set of random variables $X = (X_1, \dots, X_d)$. The distribution P_X is Markovian

with respect to \mathcal{G} where $\mathcal{G} = (V, E)$ is a DAG that encodes the causal structures among the random variables $X_i \in X$ (Peters et al., 2017). The node $i \in V$ corresponds to the random variable $X_i \in X$ and edges $(i, j) \in E$ correspond to the causal relations encoded by \mathcal{G} . In a causal graphical model, the joint distribution P_x can be factorized as $p(x) = \prod_{i=1}^d p(x_i | x_{pa_i}^{\mathcal{G}})$ where $X_{pa_i}^{\mathcal{G}}$ refers to the set of parents (direct causes) for the variable X_i in DAG \mathcal{G} and for each $X_j \in X_{pa_i}^{\mathcal{G}}$ there is an edge $(X_j \rightarrow X_i) \in E$ (Peters et al., 2017).

2.2 Score-Based Structure Recovery

In a structure recovery method, given n i.i.d. observations in the data matrix $\mathbf{X} = [\mathbf{x}_1 | \dots | \mathbf{x}_d] \in \mathbb{R}^{n \times d}$, our goal is to learn the underlying causal relations encoded by the DAG \mathcal{G} . Most of the approaches follow either a constraint-based or a score-based strategy for observational causal discovery. A score-based approach typically concentrates on identifying the DAG model \mathcal{G} that fits the observed set of data \mathbf{X} according to some scoring criterion $S(\mathcal{G}, X)$ over the discrete space of DAGs \mathbb{D} where $\mathcal{G} \in \mathbb{D}$ (Chickering, 2002). The optimization problem for structure recovery in this case can be defined as follows:

$$\begin{aligned} & \min_{\mathcal{G}} S(\mathcal{G}, X) \\ & \text{subject to} \quad \mathcal{G} \in \mathbb{D} \end{aligned} \quad (1)$$

The challenge with Eq. 1 is that the acyclicity constraint in the optimization is combinatorial in nature and scales exponentially with the number of nodes d in the graph. This makes the optimization problem NP-hard (Chickering, 1996; Chickering et al., 2004).

2.3 NOTEARS: Continuous Optimization for Structure Learning

NOTEARS (Zheng et al., 2018) is a score-based structure learning approach which reformulates the combinatorial optimization problem to a continuous one through an algebraic characterization of the acyclicity constraint in Eq. 1 via trace exponential. This method encodes the graph \mathcal{G} defined over the d nodes to a weighted adjacency matrix $W = [w_1 | \dots | w_d] \in \mathbb{R}^{d \times d}$ where $w_{ij} \neq 0$ if there is an active edge $X_i \rightarrow X_j$ and $w_{ij} = 0$ if there is not. The weighted adjacency matrix W entails a linear SEM by $X_i = f_i(X) + N_i = w_i^T X + N_i$; where N_i is the associated noise. The authors define a smooth score function on the weighted matrix as $h(W) = \text{tr}(e^{W \circ W}) - d$ where \circ is the Hadamard product and e^M is the matrix exponential of M . This embedding of the graph

\mathcal{G} and the characterization of acyclicity turns the optimization in Eq. 1 into its equivalent:

$$\begin{aligned} & \min_{W \in \mathbb{R}^{d \times d}} L(W) \\ & \text{subject to} \quad h(W) = 0 \end{aligned} \quad (2)$$

where $L(W)$ is the least square loss over W and $h(W)$ score defines the DAG-ness of the graph.

2.4 Nonparametric Extension of NOTEARS

A nonparametric extension of the continuous optimization suggested by a subsequent study (Zheng et al., 2020) uses partial derivatives for asserting the dependency of f_j on the random variables. The authors define $f_j \in H^1(\mathbb{R}^d) \subset L^2(\mathbb{R}^d)$ over the Sobolev space of square integrable functions whose derivatives are also square integrable. The authors show that f_j can be independent of random variable X_i if and only if $\|\partial_i f_j\|_{L^2} = 0$ where ∂_i denotes partial derivative with respect to the i -th variable. This redefines the weighted adjacency matrix with $W(f) = W(f_1, \dots, f_d) \in \mathbb{R}^{d \times d}$ where each W_{ij} encodes the partial dependency of f_j on variable X_i . As a result, we can equivalently write Eq. 2 as follows:

$$\begin{aligned} & \min_{f: f_j \in H^1(\mathbb{R}^d), \forall j \in [d]} L(f) \\ & \text{subject to} \quad h(W(f)) = 0 \end{aligned} \quad (3)$$

for all $X_j \in X$. Two of the general instances proposed by (Zheng et al., 2020) are: NOTEARS-MLP and NOTEARS-Sob. A multilayer perceptron having h number of hidden layers and $\sigma: \mathbb{R} \rightarrow \mathbb{R}$ activation function can be defined as $M(\mathbf{X}; L) = \sigma(L^{(h)} \sigma(\dots \sigma(L^{(1)} X))$ where $L^{(l)}$ denotes the parameters associated with l -th hidden layer. The authors in (Zheng et al., 2020) show that if $\|i$ -th column of $L_j^{(1)}\|_2 = 0$ then $M_j(\mathbf{X}; L)$ will be independent of variable X_i which replaces the association of partial derivatives in Eq. 3 and redefines the adjacency matrix as $W(\theta)$ with $W(\theta)_{ij} = \|i$ -th column of $L_j^{(1)}\|_2$ where $\theta = (\theta_1, \dots, \theta_d)$; θ_k denoting the set of parameters for the $M_k(\mathbf{X}; L)$ (k -th MLP). With the usage of neural networks and the augmented Lagrangian method (Bertsekas, 1997) NOTEARS-MLP solves the constrained problem in Eq. 3 as follows:

$$\begin{aligned} & \min_{\theta} F(\theta) + \lambda \|\theta\|_1 \\ F(\theta) = & L(\theta) + \frac{\rho}{2} |h(W(\theta))|^2 + \alpha h(W(\theta)) \end{aligned} \quad (4)$$

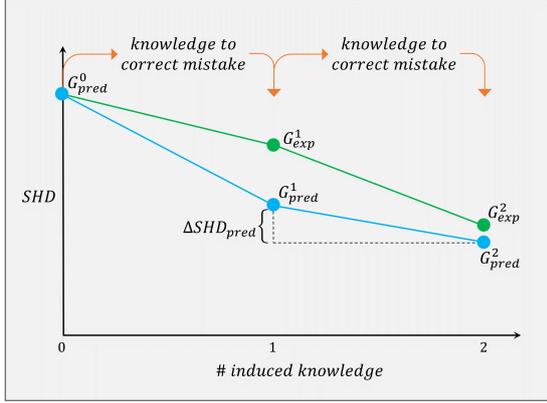


Figure 1: Knowledge induction process. We induce knowledge by carrying over the existing knowledge set along with a new random correction informed by model mistakes.

3 KNOWLEDGE INDUCTION

In our formulation, we use the multilayer perceptrons of NOTEARS-MLP proposed by (Zheng et al., 2020) as our estimators. We extend this framework to incorporate causal knowledge by characterizing the extra information as additional constraints in the optimization in Eq. 3.

Knowledge Type. We distinguish between these two types of knowledge: (i) *known inactive* is knowledge from the true inactive edges (absence of direct causal relation), and (ii) *known active* is knowledge from the true active edges (presence of direct causal relation).

Knowledge Induction Process. We adopt an interactive induction process, where the expert knowledge is informed by the outcome of the causal discovery model. Namely, the knowledge is induced to correct the mistakes of the model in the causal structure, in the hope that the new structure is closer to the true causal graph. This process is applied sequentially by correcting the mistakes of the model at each step.

In the following subsections we present the formulation of the NOTEARS optimization with constraints and detail the sequential induction process.

3.1 Expert Knowledge as Constraints

An induced knowledge associated with a true active edge, $X_i \rightarrow X_j$ (*known active*) enforces the corresponding cell in the adjacency matrix to be non-zero, $[W(\theta)]_{ij} \neq 0$. We consider this knowledge as

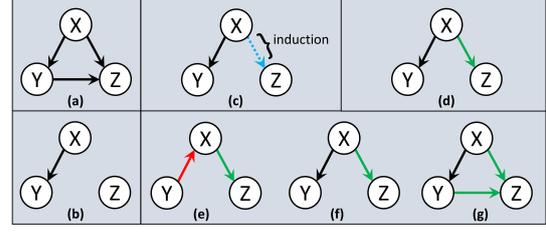


Figure 2: Expected graph formulation: (a) true graph, G_{true} , (b) predicted graph by model at step k , G_{pred}^k , (c) induced knowledge at step $(k+1)$, (d) expected graph at step $(k+1)$, G_{exp}^{k+1} . Three different examples of many possible predicted graphs at step $(k+1)$, G_{pred}^{k+1} where the model performs (e) less than expectation, (f) par with expectation, and (g) more than expectation.

inequality constraint in our extension of the optimization such that the following statement holds:

$$h_{ineq}^p(W(\theta)) > 0 \quad (5)$$

where p enumerates over all the inequality constraints due to induction from the set of *known active* and h_{ineq} is the penalty score associated with the violation of these inequality constraints. On the other hand, knowledge associated with true inactive edge, $X_i \not\rightarrow X_j$ (*known inactive*) enforces the related cell in $W(\theta)$ to be equal to zero, $[W(\theta)]_{ij} = 0$ if the induction implies there should not be an edge from X_i to X_j . We consider this knowledge as equality constraint in our optimization such as:

$$h_{eq}^q(W(\theta)) = 0 \quad (6)$$

where q enumerates over all the equality constraints, induced from the set of *known inactive* and h_{eq} is the penalty score associated with the violation of these equality constraints. With these additional constraints in Eqs. 5, 6 we extend Eq. 3 to incorporate causal knowledge in the optimization as follows:

$$\begin{aligned} & \min_{f: f_j \in H^1(\mathbb{R}^d), \forall j \in [d]} L(f) \\ & \text{subject to} \quad h(W(\theta)) = 0, \\ & \quad \quad \quad h_{eq}^q(W(\theta)) = 0, \\ & \quad \quad \quad h_{ineq}^p(W(\theta)) > 0 \end{aligned} \quad (7)$$

NOTEARS uses a thresholding step on the estimated edge weights to reduce false discoveries by pruning all the edges with weights falling below a certain threshold. Because of this, in practice, even the equality constraints in Eq. 6 become inequalities to allow for small weights. Finally, slack variables are introduced in the implementation to transform the inequality constraints into equality constraints (see detailed formulation in Appendix A).

By using the similar strategy suggested by Zheng et al. (Zheng et al., 2020) with augmented Lagrangian method the reframed constrained optimization of Eq. 4 takes the following form:

$$\begin{aligned}
 & \min_{\theta} F(\theta) + \lambda \|\theta\|_1 \\
 F(\theta) = & L(\theta) + \frac{\rho}{2} |h(W(\theta))|^2 + \alpha h(W(\theta)) \\
 & + \sum_p \left(\frac{\rho_{ineq}}{2} |h_{ineq}^p(W(\theta))|^2 + \alpha_p h_{ineq}^p(W(\theta)) \right) \quad (8) \\
 & + \sum_q \left(\frac{\rho_{eq}}{2} |h_{eq}^q(W(\theta))|^2 + \alpha_q h_{eq}^q(W(\theta)) \right)
 \end{aligned}$$

3.2 Sequential Knowledge Induction

In case of knowledge induction, the optimization is run in a sequential manner where the constraints are informed by the causal mistakes made by the model in the previous step. We start with our baseline model without imposing any additional knowledge from the true DAG and get the predicted causal graph denoted by \mathcal{G}_{pred}^0 in Figure 1. Then at each iterative step ($k+1$), based on the mistakes in the causal graph \mathcal{G}_{pred}^k predicted by the NOTEARS-MLP, we select one additional random piece of knowledge to correct one of the mistakes, and add it to the set of constraints identified in the previous k steps, and rerun NOTEARS. We note that a batch of corrections can also be selected, however for this work we have focused on estimating the contribution of each piece of knowledge in the form of known active/inactive edge. Our observations are illustrated in Section 4.1, Section 4.2, Section 4.3, and Section 4.4.

Expected Causal Graph. We consider the expected causal graph, \mathcal{G}_{exp}^{k+1} at step ($k+1$) by considering the case where all the knowledge has successfully been induced without impacting any other edges. Figure 2d illustrates an example of how we formulate our expected graph for a particular step in the iterative process. We note that the correction might yield a directed graph (Expected Causal Graph) that is not necessary a DAG. The objective is to compare the performance between the causal graph predicted by NOTEARS and the expected causal graph. Our intuition is that the induced knowledge will probably correct additional incorrect edges, see Figure 2g, yielding a performance better than expected.

Table 1: Performance metrics considered with their corresponding desirability.

Metric	Desirability
Δ FDR	Lower is better
Δ TPR	Higher is better
Δ FPR	Lower is better
Δ SHD	Lower is better

Table 2: Results for inducing redundant knowledge.

Metric	Mean \pm Stderr.	Remarks
Δ FDR	-0.00030 \pm 0.00017	No harm
Δ TPR	-0.00035 \pm 0.00027	No harm
Δ FPR	-0.00097 \pm 0.00059	No harm
Δ SHD	-0.00154 \pm 0.00167	No harm

4 EXPERIMENTS

To empirically evaluate the impact of additional causal knowledge on causal learning and to keep our experimental setup similar to the study in Ref. (Zheng et al., 2020), we have used an MLP with 10 hidden units and sigmoid activation functions. In all our experimental setup, we assume the prior knowledge is correct (agrees with the true DAG). Despite the known sensitivity of the NOTEARS algorithm to data scaling, as demonstrated in previous study (Reisach et al., 2021), we have conducted experiments using both unscaled and scaled data to ensure the robustness of our findings and we are pleased to report that our conclusions remain unchanged regardless of the scaling of the data, indicating the stability and reliability of our results. While we present the results using the unscaled data for consistency with the original implementation of NOTEARS (Zheng et al., 2020), it is important to note that our conclusions hold true even when the data is scaled.

Simulation. We investigate the performance of our formulation and the impact of induced knowledge by comparing the DAG estimates with the ground truths. For our simulations with synthetic data, we have considered 16 different combinations following the simulation criteria: two random graph models, Erdos-Renyi (ER) and Scale-Free (SF), number of nodes, $d = \{10, 20\}$, sample size, $n = \{200, 1000\}$, edge density, $s_0 = \{1d, 4d\}$. For each of these combinations, we have generated 10 different random graphs or true DAGs (as 10 trials for a particular combination) and corresponding data by following nonlinear data generating process with index models (similar to the study in Ref. (Zheng et al., 2020) for which the underlying true DAGs are identifiable. The results

Table 3: Results for inducing knowledge that corrects model’s mistake.

Metric	Knowledge	Mean \pm Stderr.	Improvement
Δ FDR	inactive	-0.018 ± 0.002	Significant
Δ FDR	active	-0.008 ± 0.001	Significant
Δ TPR	inactive	-0.007 ± 0.003	Not significant
Δ TPR	active	0.024 ± 0.003	Significant
Δ FPR	inactive	-0.023 ± 0.004	Significant
Δ FPR	active	-0.008 ± 0.003	Significant
Δ SHD	inactive	-0.032 ± 0.012	Significant
Δ SHD	active	-0.071 ± 0.011	Significant

are summarized over all these 160 random true DAGs and datasets. In our simulations, we have considered the regularization parameter, $\lambda = 0.01$. We evaluate the performance of causal learning based on the mean and the standard error of different metrics. For statistical significance analysis, we have used t-test with $\alpha = 0.05$ as the significance level.

Metrics. For the comparative analysis, we consider the following performance metrics: False Discovery Rate (FDR), True Positive Rate (TPR), False Positive Rate (FPR), and Structural Hamming Distance (SHD). However, since we are evaluating the performance over all these 160 random graphs of varying sizes, we consider Structural Hamming Distance per node (SHD/d) as our SHD measure that scales with the number of nodes (FDR, TPR, and FPR scale by definition). To evaluate the impact of induced knowledge, we calculate the differences in the metrics at different steps (where we have different sizes of induced knowledge set) and referred them as Δ FDR, Δ TPR, Δ FPR, and Δ SHD, see also Table 1. For example, based on our model’s prediction we calculate the impact of inducing one additional piece of knowledge on the metric SHD (Δ SHD_{pred}) as follows:

$$\Delta\text{SHD}_{\text{pred}} = \text{SHD}(\mathcal{G}_{\text{pred}}^{k+1}) - \text{SHD}(\mathcal{G}_{\text{pred}}^k) \quad (9)$$

Sanity Check - Redundant Knowledge Does no Harm. As part of our sanity check, we investigate the impact of induced knowledge that matches the causal relationships successfully discovered by the NOTEARS-MLP. Therefore, in this section, we consider the set of edges that our baseline model correctly classifies as our knowledge source. Here, we do not distinguish between the edge types of our induced knowledge (*known inactive* & *active*) since our goal is to investigate whether having redundant knowledge as additional constraints affects model’s performance or not. The results are illustrated in Table 2. Our empirical evaluation shows that adding redundant knowledge does not deteriorate the performance of NOTEARS-MLP. Our performed statistical test re-

flects that the results after inducing the knowledge from the correctly classified edge set are not statistically different than the results from the model without these knowledge inductions. However, we have noticed that the performance gets worse with highly regularized models. This is consistent with observations by Ng et al. (Ng et al., 2020) where sparse DAGs result in missing some of the true active edges.

4.1 Knowledge that Corrects Model’s Mistake

We first investigate the role of randomly chosen knowledge that corrects model’s mistake based on the cause-effect relations of the true graph. Therefore, in this case, we consider the set of misclassified edges from the estimated causal graph as the knowledge source for biasing the model. The results are illustrated in Table 3. Our empirical result shows statistically significant improvements whenever the induced knowledge corrects misclassified edges in the estimated causal graph except for the case of Δ TPR with *known inactive* edges. However, this behavior is not totally unexpected since knowledge from *known inactive* edges helps to get rid of false discoveries or false positives, which hardly have impact on true positives.

4.2 Known Inactive vs Known Active

In this subsection, we are interested in understanding the impact of different types of induced knowledge on causal discovery to correct the mistakes in the estimated causal graph. As a result, the experimental setup is similar to Section 4.1 where we consider the misclassified edge set as the knowledge source. We consider both *known inactive* and *known active* types of knowledge to induce separately and analyze the differences of their impact on the performance. The results are illustrated in Table 4. Based on our statistical test, we have found that inducing *known inactive* is more effective when we compare the performance

Table 4: Comparison between the impact of inducing knowledge regarding inactive vs active edges.

Metric	Inactive	Active	Better
ΔFDR	-0.019 ± 0.002	-0.008 ± 0.001	inactive
ΔTPR	-0.007 ± 0.003	0.024 ± 0.003	active
ΔFPR	-0.023 ± 0.004	-0.009 ± 0.004	inactive
ΔSHD	-0.033 ± 0.013	-0.072 ± 0.011	active

Table 5: Comparison between the empirical performance vs expectation.

Metric	Knowledge	Empirical	Expected	Remarks
ΔFDR	inactive	-0.019 ± 0.002	-0.016 ± 0.002	No difference
ΔFDR	active	-0.008 ± 0.001	-0.006 ± 0.001	No difference
ΔTPR	inactive	-0.007 ± 0.003	-0.002 ± 0.003	No difference
ΔTPR	active	0.024 ± 0.003	0.022 ± 0.002	No difference
ΔFPR	inactive	-0.023 ± 0.004	-0.021 ± 0.004	No difference
ΔFPR	active	-0.009 ± 0.003	-0.007 ± 0.003	No difference
ΔSHD	inactive	-0.033 ± 0.013	-0.047 ± 0.010	No difference
ΔSHD	active	-0.072 ± 0.011	-0.056 ± 0.010	No difference

based on FDR and FPR as misclassification of inactive edges has more impact on these metrics. On the other hand, the results show that inducing *known active* is more effective on TPR as misclassification of active edges has more impact on this metric. Interestingly, we have found that *known active* provides a significant improvement over *known inactive* in terms of SHD. This can be attributed to the fact that the induced knowledge based on the true inactive edge (*known inactive*) between two random variables, i.e. from X_i to X_j allows for two extra degrees of freedom since it is still possible to have no edge at all or an active edge from X_j to X_i . However, the induced knowledge based on the true active edge doesn't allow any degrees of freedom. This type of knowledge is more restraining for causal graph discovery and therefore carries more information.

4.3 Empirical Performance vs Expectation

In this subsection, we are interested in understanding whether inducing knowledge to correct model's mistakes exceeds expected improvement. The experimental setup is similar to Section 4.1 and Section 4.2 where we consider the misclassified edge set as the knowledge source. We have conducted the experiments using both *known inactive* and *known active* types of knowledge separately. The expected causal graph, \mathcal{G}_{exp} is formulated in a similar manner described in Fig. 2. Table 5 shows the summary of the performance comparison in these cases with the expected results. Our statistical test shows that the induced correct knowledge does not correct on average more incorrect active and/or inactive

edges than expected. Therefore, using the information from induced knowledge does not have additional impact than expected in the global optimization scheme. However, this is likely due to the fact that the structure of the expected causal graph, \mathcal{G}_{exp} is not well-posed. It's worth noting that \mathcal{G}_{exp} isn't necessarily a DAG since there isn't any constraining mechanism to enforce acyclicity as compared to \mathcal{G}_{pred} (NOTEARS imposes hard acyclicity constraint in the continuous optimization). Although it is to be noted here that solving an acyclicity constrained optimization problem does not guarantee to return a DAG and Ng et al. (Ng et al., 2022) in their study illustrates on this behavior and proposes the convergence guarantee with a DAG solution.

4.4 Real Data

We evaluate the implication of incorporating expert knowledge on the dataset from study in Ref. (Sachs et al., 2005), which is largely used in the literature of probabilistic graphical models with a consensus network accepted by the biological community. This dataset contains the expression levels of phosphorylated proteins and phospholipids in human cells under different conditions. The dataset has $d = 11$ cell types along with $n = 7466$ samples of expression levels. As for the ground truth of the underlying causal graph, we considered $s_0 = 20$ active edges as suggested by the study (Sachs et al., 2005). We have opted for ΔTPR , the percentage difference of edges in agreement (higher is better), and the percentage difference of reversed edges (lower is better) as the evaluation metrics since the performance on these metrics would indicate the significance more distinctively. Similar

to the synthetic data analysis, we had 10 trials that we used to summarize our evaluation. Our empirical result (Mean \pm Stderr.) shows: Δ TPR as 0.020 ± 0.004 , the percentage difference of edges in agreement as 0.393 ± 0.086 , and the percentage difference of reversed edges as -0.073 ± 0.030 . We have found that with the help of induced knowledge the model shows statistically significant improvement by correctly identifying more active edges and by reducing the number of edges identified in the reverse direction. Due to the limitation of having access only to a subset of the true active edges, our analyses could not include a comparative study on *known inactive* edges as in the synthetic data case. We assume the performance could have been improved by fine-tuning the model's parameters but since our main focus of this study is entirely based on the analyses regarding the impact of induced knowledge of different types and from different sources on structure learning, we kept the parameter setup similar for all consecutive steps in the knowledge induction process.

5 CONCLUSIONS

We have studied the impact of expert causal knowledge on causal structure learning and provided a set of comparative analyses of biasing the model using different types of knowledge. Our findings show that knowledge that corrects model's mistakes yields significant improvements and it does no harm even in the case of redundant knowledge that results in redundant constraints. This suggests that the practitioners should consider incorporating domain knowledge whenever available. More importantly, we have found that knowledge related to active edges has a larger positive impact on causal discovery than knowledge related to inactive edges which can mostly be attributed to the difference between the number of degrees of freedom each case reduces. This finding suggests that the practitioners may want to prioritize incorporating knowledge regarding presence of an edge whenever applicable. Furthermore, our experimental analysis shows that the induced knowledge does not correct on average more incorrect active and/or inactive edges than expected. This finding is rather surprising to us, as we have expected that every constraint based on a known active/inactive edge to impact and correct more than one edge on average.

Our work points to the importance of the human-in-the-loop in causal discovery that we would like to further explore in our future studies. Also, we would like to mention that in our study we adopted hard constraints to accommodate the prior knowledge since we

have assumed our priors to be correct. An interesting future direction would be to accommodate the continuous optimization with functionality to allow different levels of confidence on the priors.

ACKNOWLEDGEMENTS

Research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-22-1-0035. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

REFERENCES

- Alrajeh, D., Chockler, H., and Halpern, J. Y. (2020). Combining experts' causal judgments. *Artificial Intelligence*, 288:103355.
- Andrews, B., Spirtes, P., and Cooper, G. F. (2020). On the completeness of causal discovery in the presence of latent confounding with tiered background knowledge. In *International Conference on Artificial Intelligence and Statistics*, pages 4002–4011. PMLR.
- Bertsekas, D. P. (1997). Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334.
- Bhattacharjya, D., Gao, T., Mattei, N., and Subramanian, D. (2021). Cause-effect association between event pairs in event datasets. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 1202–1208.
- Bradley, R., Dietrich, F., and List, C. (2014). Aggregating causal judgments. *Philosophy of Science*, 81(4):491–515.
- Chickering, D. M. (1996). Learning Bayesian networks is NP-complete. In *Learning from data*, pages 121–130. Springer.
- Chickering, D. M. (2002). Optimal structure identification with greedy search. *Journal of machine learning research*, 3(Nov):507–554.
- Chickering, M., Heckerman, D., and Meek, C. (2004). Large-sample learning of Bayesian networks is NP-hard. *Journal of Machine Learning Research*, 5.
- Colombo, D., Maathuis, M. H., Kalisch, M., and Richardson, T. S. (2012). Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pages 294–321.
- Cooper, G. (1995). Causal discovery from data in the presence of selection bias. In *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, pages 140–150.

- Elkan, C. (2001). The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd.
- Fang, Z., Zhu, S., Zhang, J., Liu, Y., Chen, Z., and He, Y. (2020). Low rank directed acyclic graphs and causal structure learning. *arXiv preprint arXiv:2006.05691*.
- Gencoglu, O. and Gruber, M. (2020). Causal modeling of twitter activity during Covid-19. *Computation*, 8(4):85.
- Heindorf, S., Scholten, Y., Wachsmuth, H., Ngonga Ngomo, A.-C., and Potthast, M. (2020). Causenet: Towards a causality graph extracted from the web. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 3023–3030.
- Holzinger, A. (2016). Interactive machine learning for health informatics: when do we need the human-in-the-loop? *Brain Informatics*, 3(2):119–131.
- Jaber, A., Zhang, J., and Bareinboim, E. (2018). Causal identification under Markov equivalence. *arXiv preprint arXiv:1812.06209*.
- Lachapelle, S., Brouillard, P., Deleu, T., and Lacoste-Julien, S. (2019). Gradient-based neural DAG learning. *arXiv preprint arXiv:1906.02226*.
- Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and brain sciences*, 40.
- Li, Z., Ding, X., Liu, T., Hu, J. E., and Van Durme, B. (2021). Guided generation of cause and effect. *arXiv preprint arXiv:2107.09846*.
- Liu, J., Chen, Y., and Zhao, J. (2021). Knowledge enhanced event causality identification with mention masking generalizations. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3608–3614.
- Magliacane, S., van Ommen, T., Claassen, T., Bongers, S., Versteeg, P., and Mooij, J. M. (2017). Domain adaptation by using causal inference to predict invariant conditional distributions. *arXiv preprint arXiv:1707.06422*.
- Ng, I., Fang, Z., Zhu, S., Chen, Z., and Wang, J. (2019). Masked gradient-based causal structure learning. *arXiv preprint arXiv:1910.08527*.
- Ng, I., Ghassami, A., and Zhang, K. (2020). On the role of sparsity and DAG constraints for learning linear DAGs. *arXiv preprint arXiv:2006.10201*.
- Ng, I., Lachapelle, S., Ke, N. R., Lacoste-Julien, S., and Zhang, K. (2022). On the convergence of continuous constrained optimization for structure learning. In *International Conference on Artificial Intelligence and Statistics*, pages 8176–8198. PMLR.
- O’Donnell, R. T., Nicholson, A. E., Han, B., Korb, K. B., Alam, M. J., and Hope, L. R. (2006). Causal discovery with prior information. In *Australasian Joint Conference on Artificial Intelligence*, pages 1162–1167. Springer.
- Pearl, J. (2009). *Causality*. Cambridge university press.
- Pearl, J. and Verma, T. S. (1995). A theory of inferred causation. In *Studies in Logic and the Foundations of Mathematics*, volume 134, pages 789–811. Elsevier.
- Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of causal inference: foundations and learning algorithms*. The MIT Press.
- Ramsey, J., Glymour, M., Sanchez-Romero, R., and Glymour, C. (2017). A million variables and more: the Fast Greedy Equivalence Search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International journal of data science and analytics*, 3(2):121–129.
- Reisach, A., Seiler, C., and Weichwald, S. (2021). Beware of the simulated dag! causal discovery benchmarks may be easy to game. *Advances in Neural Information Processing Systems*, 34:27772–27784.
- Sachs, K., Perez, O., Pe’er, D., Lauffenburger, D. A., and Nolan, G. P. (2005). Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529.
- Scutari, M. (2009). Learning bayesian networks with the bnlearn r package. *arXiv preprint arXiv:0908.3817*.
- Sharma, A. and Kiciman, E. (2020). Dowhy: An end-to-end library for causal inference. *arXiv preprint arXiv:2011.04216*.
- Spirtes, P., Glymour, C. N., Scheines, R., and Heckerman, D. (2000). *Causation, prediction, and search*. MIT press.
- Wei, D., Gao, T., and Yu, Y. (2020). Dags with no fears: A closer look at continuous optimization for learning bayesian networks. *arXiv preprint arXiv:2010.09133*.
- Xin, D., Ma, L., Liu, J., Macke, S., Song, S., and Parameswaran, A. (2018). Accelerating human-in-the-loop machine learning: Challenges and opportunities. In *Proceedings of the second workshop on data management for end-to-end machine learning*, pages 1–4.
- Yang, Y., Kandogan, E., Li, Y., Sen, P., and Lasecki, W. S. (2019). A study on interaction in human-in-the-loop machine learning for text analytics. In *IUI Workshops*.
- Yu, Y., Chen, J., Gao, T., and Yu, M. (2019). DAG-GNN: DAG structure learning with graph neural networks. In *International Conference on Machine Learning*, pages 7154–7163. PMLR.
- Zadrozny, B. (2004). Learning and evaluating classifiers under sample selection bias. In *Proceedings of the twenty-first international conference on Machine learning*, page 114.
- Zhang, K., Zhu, S., Kalander, M., Ng, I., Ye, J., Chen, Z., and Pan, L. (2021). gcastle: A python toolbox for causal discovery. *arXiv preprint arXiv:2111.15155*.
- Zheng, X., Aragam, B., Ravikumar, P., and Xing, E. P. (2018). DAGs with no tears: Continuous optimization for structure learning. *arXiv preprint arXiv:1803.01422*.
- Zheng, X., Dan, C., Aragam, B., Ravikumar, P., and Xing, E. (2020). Learning sparse nonparametric DAGs. In *International Conference on Artificial Intelligence and Statistics*, pages 3414–3425. PMLR.

APPENDIX

We illustrate here the detailed performance with summary statistics of induced knowledge from our empirical evaluation (Δ FDR, Δ TPR, Δ FPR, and Δ SHD for both $\Delta=1$ and $\Delta=2$). Similar to one additional knowledge ($\Delta=1$), we calculate the impact of inducing two additional piece of knowledge ($\Delta=2$) based on our model’s prediction i.e. on the metric SHD (Δ SHD $^2_{pred}$) as follows:

$$\Delta\text{SHD}^2_{pred} = \text{SHD}(\mathcal{G}_{pred}^{k+2}) - \text{SHD}(\mathcal{G}_{pred}^k) \quad (10)$$

Table 6 shows the results for inducing redundant knowledge or knowledge that is correctly classified by NOTEARS-MLP.

A. Threshold Incorporation and Slack Variables.

In Eq. 5, we have seen that our inequality constraint takes the following form:

$$h_{ineq}^p(W(\theta)) > 0$$

where p enumerates over each induced knowledge associated with true active edge (*known active*) $X_i \rightarrow X_j$ imposing $[W(\theta)]_{ij} \neq 0$. NOTEARS uses a thresholding step that reduces false discoveries where any edge

weight below the threshold value, w_{thresh} in its absolute value is set to zero. Thus, for any induction from true active edges ($X_i \rightarrow X_j$) we have the following constraint:

$$[W(\theta)]_{ij}^2 \geq W_{thresh}^2.$$

We convert inequality constraints in our optimization to equality by introducing a set of slack variables y_p such that:

$$-[W(\theta)]_{ij}^2 + W_{thresh}^2 + y_p = 0 \quad \text{s.t. } y_p \geq 0 \quad (11)$$

In a similar manner, using the threshold value, W_{thresh} our equality constraints (associated with *known inactive* edges) take the form as:

$$[W(\theta)]_{ij}^2 - W_{thresh}^2 + y_q = 0 \quad \text{s.t. } y_q \geq 0 \quad (12)$$

where q enumerates over each induction associated with true inactive edge $X_i \nrightarrow X_j$ imposing $[W(\theta)]_{ij} = 0$.

B. Additional Results and Summary Statistics.

Table 7 shows the detailed results for inducing knowledge that corrects model’s mistake. Table 8 shows the

Table 6: Full results for inducing redundant knowledge (Sanity Check).

Metric	Δ	Mean \pm Stdev.	p-value	t-stat	Remarks
Δ FDR	1	-0.00030 \pm 0.00017	0.076	-1.770	No harm
Δ FDR	2	-0.00060 \pm 0.00021	0.004	-2.850	No harm
Δ TPR	1	-0.00035 \pm 0.00027	0.205	-1.260	No harm
Δ TPR	2	-0.00036 \pm 0.00029	0.227	-1.210	No harm
Δ FPR	1	-0.00097 \pm 0.00059	0.100	-1.630	No harm
Δ FPR	2	-0.00183 \pm 0.00069	0.008	-2.660	No harm
Δ SHD	1	-0.00154 \pm 0.00167	0.356	-0.920	No harm
Δ SHD	2	-0.00357 \pm 0.00188	0.050	-1.900	No harm

Table 7: Full results for inducing knowledge that corrects model’s mistake (Section 4.1).

Metric	Δ	Knowledge	Mean \pm Stdev.	p-value	t-stat	Improvement
Δ FDR	1	inactive	-0.018, 0.002	3.41E-14	-7.800	Significant
Δ FDR	1	active	-0.008, 0.001	2.51E-08	-5.657	Significant
Δ FDR	2	inactive	-0.023, 0.003	2.74E-15	-8.221	Significant
Δ FDR	2	active	-0.011, 0.002	9.06E-08	-5.448	Significant
Δ TPR	1	inactive	-0.007, 0.003	3.10E-02	-2.164	Not significant
Δ TPR	1	active	0.024, 0.003	8.58E-19	9.191	Significant
Δ TPR	2	inactive	-0.001, 0.003	8.25E-01	-0.222	Not significant
Δ TPR	2	active	0.035, 0.004	1.16E-19	9.580	Significant
Δ FPR	1	inactive	-0.023, 0.004	3.81E-08	-5.583	Significant
Δ FPR	1	active	-0.008, 0.003	1.21E-02	-2.517	Significant
Δ FPR	2	inactive	-0.021, 0.003	1.04E-08	-5.845	Significant
Δ FPR	2	active	-0.015, 0.005	6.73E-03	-2.724	Significant
Δ SHD	1	inactive	-0.032, 0.012	9.74E-03	-2.594	Significant
Δ SHD	1	active	-0.071, 0.011	1.61E-10	-6.522	Significant
Δ SHD	2	inactive	-0.082, 0.012	1.93E-10	-6.533	Significant
Δ SHD	2	active	-0.126, 0.016	3.41E-14	-7.875	Significant

Table 8: Full results of comparison between the impact of inducing knowledge regarding inactive vs active edges. (Section 4.2).

Metric	Δ	Inactive	Active	p-value	t-stat	Better
Δ FDR	1	-0.019 ± 0.002	-0.008 ± 0.001	1.30E-04	-3.85	Inactive
Δ FDR	2	-0.023 ± 0.002	-0.011 ± 0.001	5.58E-04	-3.47	Inactive
Δ TPR	1	-0.007 ± 0.003	0.024 ± 0.003	8.13E-14	-7.57	Active
Δ TPR	2	-0.001 ± 0.003	0.035 ± 0.004	2.84E-13	-7.43	Active
Δ FPR	1	-0.023 ± 0.004	-0.009 ± 0.004	7.28E-03	-2.69	Inactive
Δ FPR	2	-0.021 ± 0.004	-0.015 ± 0.005	3.23E-01	-0.99	No difference
Δ SHD	1	-0.033 ± 0.013	-0.072 ± 0.011	1.90E-02	2.35	Active
Δ SHD	2	-0.082 ± 0.013	-0.126 ± 0.016	3.28E-02	2.14	Active

Table 9: Full results of comparison between the empirical performance vs expectation (Section 4.3).

Metric	Δ	Knowledge	Empirical	Expected	p-value	t-stat	Remarks
Δ FDR	1	inactive	-0.019 ± 0.002	-0.016 ± 0.002	0.51	-0.65	No difference
Δ FDR	1	active	-0.008 ± 0.001	-0.006 ± 0.001	0.21	-1.25	No difference
Δ FDR	2	inactive	-0.023 ± 0.002	-0.025 ± 0.002	0.60	0.53	No difference
Δ FDR	2	active	-0.011 ± 0.002	-0.010 ± 0.002	0.75	-0.32	No difference
Δ TPR	1	inactive	-0.007 ± 0.003	-0.002 ± 0.003	0.22	-1.23	No difference
Δ TPR	1	active	0.024 ± 0.003	0.022 ± 0.002	0.48	0.70	No difference
Δ TPR	2	inactive	-0.001 ± 0.003	-0.006 ± 0.003	0.24	1.17	No difference
Δ TPR	2	active	0.035 ± 0.004	0.028 ± 0.004	0.18	1.34	No difference
Δ FPR	1	inactive	-0.023 ± 0.004	-0.021 ± 0.004	0.62	-0.50	No difference
Δ FPR	1	active	-0.009 ± 0.003	-0.007 ± 0.003	0.79	-0.27	No difference
Δ FPR	2	inactive	-0.021 ± 0.004	-0.030 ± 0.005	0.18	1.34	No difference
Δ FPR	2	active	-0.015 ± 0.005	-0.018 ± 0.005	0.61	0.51	No difference
Δ SHD	1	inactive	-0.033 ± 0.013	-0.047 ± 0.010	0.36	0.91	No difference
Δ SHD	1	active	-0.072 ± 0.011	-0.056 ± 0.010	0.30	-1.04	No difference
Δ SHD	2	inactive	-0.082 ± 0.013	-0.086 ± 0.013	0.82	0.23	No difference
Δ SHD	2	active	-0.126 ± 0.016	-0.100 ± 0.017	0.28	-1.09	No difference

Table 10: Full results for inducing knowledge in real data (Section 4.4).

Metric	Δ	Mean \pm Stderr.	p-value	t-stat	Remarks
Δ TPR	1	0.020 ± 0.004	8.10E-06	4.60	Improvement
Δ TPR	2	0.036 ± 0.005	1.77E-12	7.62	Improvement
Δ % edge in agreement	1	0.393 ± 0.086	8.10E-06	4.60	Improvement
Δ % edge in agreement	2	0.714 ± 0.094	1.77E-12	7.62	Improvement
Δ % edge reversed	1	-0.073 ± 0.030	1.54E-02	-2.45	Improvement
Δ % edge reversed	2	-0.107 ± 0.033	1.29E-03	-3.27	Improvement

detailed results of the difference between the impact of ‘known inactive’ (knowledge induced from inactive edges) and ‘known active’ (knowledge induced from active edges) using misclassified edge set as the knowledge source. Table 9 shows the detailed results of the difference between empirical improvements due to knowledge induction vs expected outcomes using misclassified edge set as the knowledge source. Table 10 shows the detailed results for inducing knowledge on the real dataset (from (Sachs et al., 2005)).