Causal Graphs vs. Causal Programs: The Case of Conditional Branching

Sam Witty
College of Information and Computer Sciences
University of Massachusetts Amherst
Amherst, MA, United States
switty@cs.umass.edu

David Jensen
College of Information and Computer Sciences
University of Massachusetts Amherst
Amherst, MA, United States
jensen@cs.umass.edu

Abstract

We evaluate the performance of graph-based causal discovery algorithms when the generative process is a probabilistic program with conditional branching. Using synthetic experiments, we demonstrate empirically that graph-based causal discovery algorithms are able to learn accurate associational distributions for probabilistic programs with context-sensitive structure, but that those graphs fail to accurately model the effects of interventions on the programs.

Keywords Structure learning, context-sensitivity, causal discovery, causal inference.

1 Introduction

Randomized experiments are often considered to be the gold standard for estimating causal effects, but experiments are often prohibitively expensive, unethical, or otherwise impractical. A set of techniques has been developed over the past 25 years that analyze observational data to learn causal models in the form of directed graphical models [1, 2, 5]. However, these algorithms implicitly assume that a graphical model will accurately represent the true data-generating process. This assumption is not always valid, and this is the key motivation behind our work.

We contrast causal graphical models with a more expressive framework: causal probabilistic programs. Causal probabilistic programs are imperative programs specified using compositions of primitive programming constructs; including deterministic assignment, stochastic assignment, conditional branching, and loops. Causal probabilistic programs extend probabilistic programs to represent causal dependencies in exactly the same way probabilistic graphical models are extended to causal graphical models: via introduction of an intervention semantics [4]. Applying an intervention $do(X = x)$ to a causal program is accomplished by replacing all assignments of X in the program with the deterministic assignment statement $X = x$. *Intervening* on X differs from *conditioning* on X in that conditioning can influence estimates of the distribution of X's ancestors and induce dependence between two ancestors which are marginally independent, whereas intervention cannot result in either of these outcomes. Much like causal graphical models, causal probabilistic programs include a semantics for generating samples and evaluating probability densities of marginal, conditional, and post-intervention distributions of random variables. Alternatively, causal probabilistic programs can be thought of as a generalization of causal graphical models, encapsulating a broader space of generative processes. In this work we are particularly interested in causal probabilistic programs with context-sensitive structure, i.e. programs where there exists at least one random variable such that the set of its parents differ for two possible executions of the program. Programs with conditional branching can have context-sensitive structure, as demonstrated in Algorithm 2.

While there has been some work on extending probabilistic graphical models to incorporate conditional branching for parameter inference [3], we are unaware of any work on the causal implications of branching in graphical models or on structure learning for generative processes with context-sensitive structure. We hope that our work will help motivate researchers to consider more expressive representations for causal models, as well as to provide preliminary insight into methods for detecting conditional branching from samples of probabilistic programs.

2 Synthetic Experiments

To evaluate the performance of graph-based structure learning algorithms we: (1) generate samples from each of the probabilistic programs in Algorithms 1 and 2; (2) learn a Markov equivalence class of graphical models using the max-min hill climbing algorithm [6]; (3) estimate local conditional probability distributions[1]; and (4) generate post-intervention samples from both the given probabilistic program and the learned graphical model for the intervention $do(A = a)$. For these synthetic experiments all prior parameters are sampled independently as follows: $\theta \sim Normal(0, 3), \mu \sim Normal(0, 1), \sigma \sim \gamma^{-1}(3, 1), p \sim \beta(5, 5)$. The post-intervention distributions are based on the intervention $do(A = 5)$.

[1]Learned local conditional probability distributions for each variable are defined as Gaussian random variables, where the conditional mean and variance are estimated using random forest regression models with respect to the random variable's parents.

PROBPROG'18, October 03–05, 2018, Boston, MA, USA
2018. ACM ISBN …$15.00
https://doi.org/

Algorithm 1 Causal Graphical Model

$A \leftarrow Normal(\mu_A, \sigma_A^2)$
$B \leftarrow Normal(\mu_B + A * \theta_{AB}, \sigma_B^2)$
$C \leftarrow Normal(\mu_C + A * \theta_{AC}, \sigma_C^2)$
$D \leftarrow Normal(\mu_D + B * \theta_{BD} + C * \theta_{CD}, \sigma_D^2)$

Algorithm 2 Branching Program

if $Bernoulli(p)$ **then**
  $A \leftarrow Normal(\mu_A, \sigma_A^2)$
  $C \leftarrow Normal(\mu_C + A * \theta_{AC}, \sigma_C^2)$
else
  $C \leftarrow Normal(\mu_C, \sigma_C^2)$
  $A \leftarrow Normal(\mu_A + C * \theta_{CA}, \sigma_A^2)$
$B \leftarrow Normal(\mu_B + A * \theta_{AB}, \sigma_B^2)$
$D \leftarrow Normal(\mu_D + B * \theta_{BD} + C * \theta_{CD}, \sigma_D^2)$

Importantly, for any given execution of the *Branching Program* the set of C's and A's parents depends on a draw from a Bernoulli random variable. Therefore, an intervention $do(A = a)$ indirectly changes the distribution of C for some subset of executions and an intervention $do(C = c)$ indirectly changes the distribution of A for all other executions. A graphical model corresponding to this probabilistic program would require that A is an ancestor of C and that C is an ancestor of A. Note also that the set of V-structures is consistent between the two branches, implying that the set of conditional independencies relating A, B, C and D does not depend on the Bernoulli random variable. We ran similar experiments for probabilistic programs where C is a mediator if $Bernoulli(p)$ and a collider otherwise (which results in different V-structures between the two branches). This produced similar empirical findings.

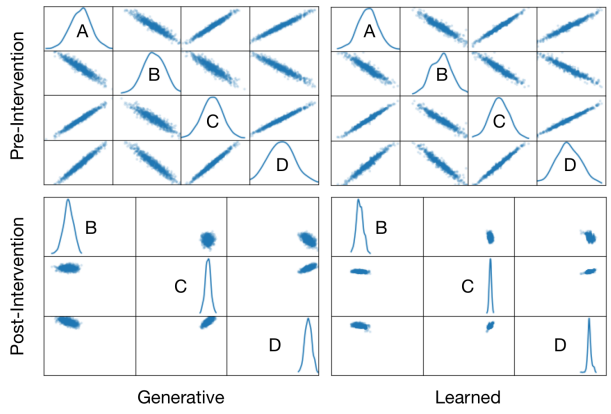

Figure 1. Causal Graphical Model Pairwise and Marginal Distributions.

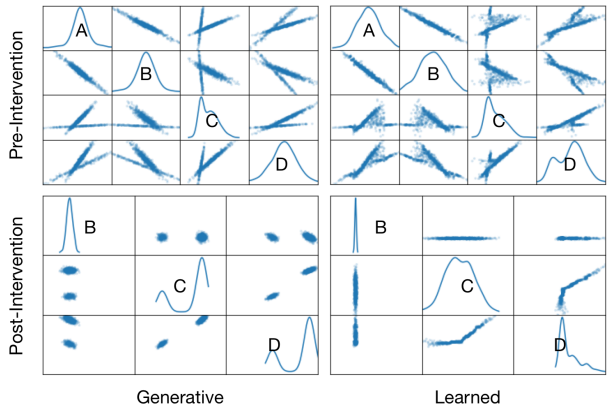

Figure 2. Branching Program Pairwise and Marginal Distributions.

3 Results and Discussion

As shown in Figure 1, samples from the learned model are qualitatively similar to samples from the generative model. With the exception of some roughness in the learned post-intervention distribution, the learned model captures the linear pairwise relationships as well as the unimodal marginal distributions. However, Figure 2 shows that while the learned pre-intervention distribution is similar to the generative pre-intervention distribution for the branching program, this is not true for the post-intervention distributions. In this setting, the learned model fails to capture the multi-modality of the generative model's post-intervention distributions. The learned model has high probability density in regions where the generative model has low probability density. For example, observe the marginal distribution of C.

These results provide evidence that learning the structure of some causal probabilistic programs cannot be achieved using established graph-based structure learning algorithms with observational data. However, we believe that interventional data may help to disambiguate between candidate probabilistic programs. Specifically, we conjecture that a promising approach to detecting context-sensitive causal structure will involve mixture modeling of post-intervention distributions. We speculate that the number of additional mixture components after intervention is closely related to the number of code blocks with differing causal structure. We expect this line of thinking to be promising for future work on learning the structure of causal probabilistic programs.

Acknowledgements

Thanks to Javier Burroni and Vikash Mansinghka for thoughtful contributions and comments. This material is based upon work supported by the United States Air Force under Contract No. FA8750-17-C-0120. Any opinions, findings and conclusions or recommendations expressed in this material are

those of the author(s) and do not necessarily reflect the views of the United States Air Force.

References

[1] David Maxwell Chickering. 2002. Optimal structure identification with greedy search. *Journal of machine learning research* 3, Nov (2002), 507–554.

[2] Dimitris Margaritis. 2003. *Learning Bayesian network model structure from data.* Technical Report. CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE.

[3] Tom Minka and John Winn. 2008. *Gates: A Graphical Notation for Mixture Models.* Technical Report. https://www.microsoft.com/en-us/research/publication/gates-a-graphical-notation-for-mixture-models/

[4] J. Pearl. 2009. *Causality.* Cambridge University Press. https://books.google.com/books?id=f4nuexsNVZIC

[5] P. Spirtes, C. Glymour, and R. Scheines. 2012. *Causation, Prediction, and Search.* Springer New York. https://books.google.com/books?id=oUjxBwAAQBAJ

[6] Ioannis Tsamardinos, Laura E. Brown, and Constantin F. Aliferis. 2006. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning* 65, 1 (01 Oct 2006), 31–78. https://doi.org/10.1007/s10994-006-6889-7