Balancing Computational Cost and Accuracy in Inference of Continuous Bayesian Networks

A Discretization and Knowledge Compilation Approach

Maarten C. Vonk[1,2] MAARTENCVONK@GMAIL.COM
Sebastiaan Brand[1]
Ninoslav Malekovic[2]
Thomas Bäck[1]
Alfons Laarman[1]
Anna V. Kononova[1]
[1] *Leiden Institute of Advanced Computer Science, Leiden University*
[2] *The Hague Centre for Strategic Studies*

Editors: J.H.P. Kwisthout & S. Renooij

Abstract

Bayesian networks allow a parsimonious encoding of joint probability distributions via directed acyclic graphs. While discrete Bayesian network inference is well-established, conducting inference on continuous Bayesian networks often requires discretization. In this paper, continuous Bayesian networks are subjected to various supervised and unsupervised discretization methods. Subsequently, the discretized Bayesian networks are encoded into decision diagrams, facilitating efficient inference. The trade-off between the quality of discretization/inference and the computational cost of inference with decision diagrams is explored by contrasting both metrics on a Pareto front. Through empirical evaluation across a range of causal and non-causal Bayesian networks, we investigate the impact of different discretization methods on this trade-off. We corroborate the significantly improved scalability of using decision diagrams for inference as opposed to traditional inference methods and extend this finding to discretized continuous networks. Coupled with insights on the accuracy-compute cost trade-off, we advocate for discretization as a viable method for Bayesian network inference on continuous networks.

Keywords: Bayesian networks; Decision Diagrams; Knowledge Compilation; Causal Inference; Bayesian Network Inference; Discretization.

1. Introduction

Bayesian networks have permeated multiple research fields such as environmental science (Kelly (Letcher) et al., 2013), defense studies (Johansson and Falkman, 2008), and biology (Su et al., 2013). While many applications require the accommodation of continuous variables (Delgado-Hernández et al., 2014; Morales-Nápoles and Steenbergen, 2015), state-of-the-art methods for continuous or hybrid (combination of discrete and continuous) Bayesian network inference are still underdeveloped. Algorithms have been developed to conduct inference on hybrid Bayesian networks when a conditional Gaussian distribution among the variables is assumed (Koller and Friedman, 2009). However, assuming the parametric form of the distribution is costly, which is why much research has been dedicated to approximation by either discretizing Bayesian networks (Beuzen et al., 2018; Nojavan

et al., 2017; Neil et al., 2007) or by approximating the distribution of the variables in the Bayesian network with a linear combination of exponentials (Rumí and Salmerón, 2007) or polynomials (Shenoy and West, 2011), which both allow inference. Nonetheless, the latter two approaches pose computational challenges as the number of regression coefficients in the functions grows linearly in the domain size of the discrete variables (Mori and Mahalec, 2016).

Discretization of the continuous variables enables the use of established discrete Bayesian network inference methods. Variable elimination and belief propagation are well-developed exact inference methods for discrete Bayesian networks that exploit the structure of the Bayesian network to substantially reduce the computational burden. Nevertheless, even with these effective algorithms, the computational cost increases exponentially as the number of parent nodes within the network grows. Therefore, researchers often employ approximate methods such as sampling or variational inference approaches for more complex Bayesian networks. These methods are summarized by Koller and Friedman (2009).

While discretization allows the use of discrete Bayesian network inference algorithms, it may lead to a loss of information, resulting in a lower accuracy of the inference query. At the same time, the computational cost of inference depends heavily on the number and positioning of bins that result from the discretization process. Or, as stated in (Koller and Friedman, 2009), "discretization provides a trade-off between the accuracy of the approximation and cost of computation."

To address the computational challenges of Bayesian network inference after discretization, knowledge compilation (Darwiche and Marquis, 2002) can be used. In knowledge compilation, information (such as the probability distribution given by a Bayesian network) is translated without loss into a format that can be queried efficiently. One of the motivations behind knowledge compilation is that by first performing a potentially computationally expensive 'compilation' step, which takes exponential time in the worst case, afterwards the result of many queries (such as inference queries) can be computed quickly. Different formats have been used for such encodings of Bayesian networks. Often, conjunctive normal form (CNF) is used as either the final target format (Sang et al., 2004, 2005a,b) or as an intermediate step in translating the Bayesian network to some other format. In those cases where CNF is not the final target format, the CNF formula is typically translated to a format where computing the result of inference queries is easier, at the cost of a larger representation. These representations include the decomposable negation normal form (DNNF) (Darwiche, 2002; Chavira and Darwiche, 2008), and decision diagrams (DDs) (Dal and Lucas, 2017; Dal et al., 2018, 2021). Other representations, such as probabilistic decision graphs (PDGs), have been shown to be no larger in their smallest form than the smallest junction tree for the same distribution (Jaeger, 2004). We specifically consider the compilation of Bayesian networks into (binary) decision diagrams (BDDs) (Bryant, 1986), as they have been shown to perform well compared to other methods such as DNNF (Dal et al., 2021). Translating Bayesian networks into decision diagrams has been done successfully for Bayesian networks which are natural discrete (Dal and Lucas, 2017; Dal et al., 2018, 2021), but has not been applied to Bayesian networks with discretized continuous variables until now.

In this paper, we study the trade-offs between the quality of the discretization/inference and the computational cost of both traditional and DD-based inference algorithms following the methodology of Figure 1. This methodology is applied to sample data derived

from causal as well as non-causal Bayesian networks, ranging in probabilistic relations, network structure and sample size. To allow a different number and positioning of discretized bins, we consider different types of discretization methods: two unsupervised approaches, equal width (EW) and equal frequency (EF) binning, as well as the supervised minimum description length principle (MDLP) binning method, class-attribute interdependence maximization (CAIM) discretization (Kurgan and Cios, 2004), ChiMerge (CM) discretization (Kerber, 1992) and dynamic discretization (DDN) (Neil et al., 2007). Conditional probability tables (CPTs) of discretized Bayesian networks are inferred via a maximum likelihood estimation (MLE) as well as via a Bayesian method with adjusted empirical Bayes priors (EBP). Subsequently, we encode the discretized Bayesian networks as CNF formulas, which are then compiled into BDDs. To assess the trade-off between the quality of discretization/inference and the cost of knowledge compilation, we use a concept known in multi-objective optimization as the Pareto front to visualize the results of various considered approaches. A Pareto front represents the set of non-dominated solutions where improving one objective would result in degrading another. The evaluation involves measuring discretization quality in terms of the earth mover's distance (EMD) and quantifying knowledge compilation cost by considering the number of nodes in the BDD. Additionally, for non-causal networks, we assess the quality of conditional queries (if ground truth is available) using the weighted root mean squared error (WRMSE). For causal Bayesian networks, such additional quality evaluation is done via the percentage error of the average treatment effect (ATE), for which we introduce the do-operator in the subsequent section.

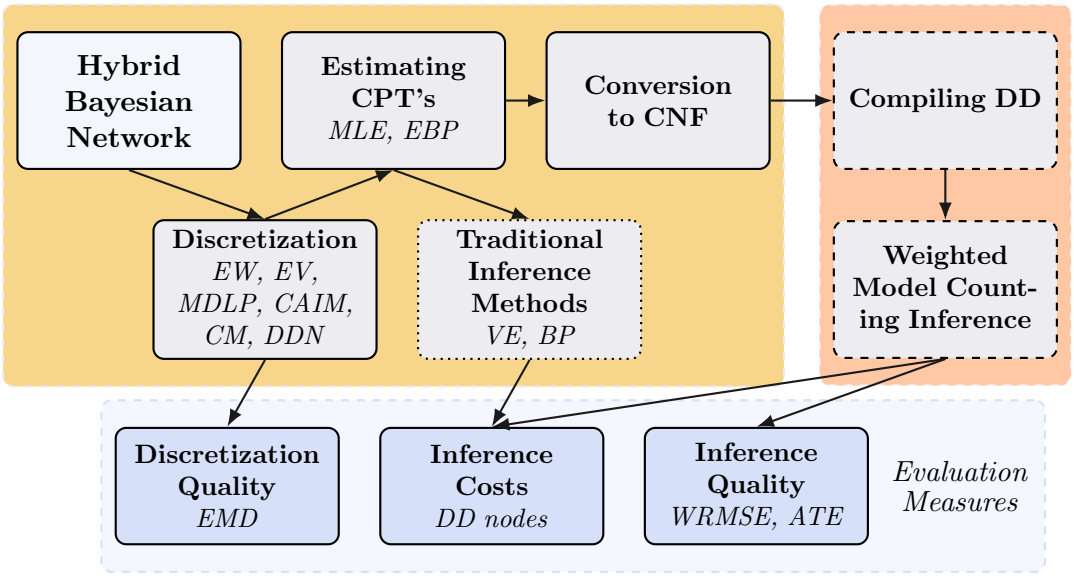


Figure 1: The implementation of the different tasks (highlighted in light gray) within the methodology together with the evaluation measures (depicted in blue), applied to hybrid Bayesian networks (indicated in white). The tasks implemented in Python comprise the vast yellow block on the left and the vast orange block on the right consists of the tasks implemented in C. The three selected measures of relevance are contrasted in terms of Pareto dominance. This methodology is applied across a wide array of Bayesian networks.

The main contributions of this work are:

- A technique to facilitate inference in hybrid Bayesian networks through discretization and BDDs, including implementation.[1]
- A detailed insight into the trade-off between inference quality and inference cost for a variety of hybrid Bayesian networks.
- Experimental evidence demonstrating the scalability advantage of BDDs compared to traditional inference methods.

The paper is structured as follows. We start by introducing the preliminaries of Bayesian networks and inference in Section 2. We then continue with the encoding of Bayesian networks to decision diagrams and discuss how the so-called weighted model counting approach can be used to compute inference queries in Section 3. In Section 4, we introduce supervised and unsupervised discretization methods and elaborate on different methods for inferring the conditional probability tables. The experimental setup is described in Section 5. Section 5.1 discusses the measures used to express the quality of discretization, the quality of inference and the computational cost of knowledge compilation with decision diagrams. After briefly introducing the different Bayesian networks in Section 5.2, some Pareto fronts are highlighted, and all results are discussed in Section 5.3. Finally, we summarize our work and propose future research directions in Sections 6 and 7.

2. Preliminaries

In this section, we introduce preliminaries and notation used throughout the paper.

The set of random variables is denoted by $X = \{X_1, \ldots, X_n\}$ where random variable X_i takes values x_i in corresponding state space Ω_{X_i}. A graph is denoted by $G = (V, E)$ with nodes $V = \{V_1, \ldots, V_n\}$ and edges $E \subseteq V \times V$. The graph is called *directed* when every edge in the graph has a direction and it is called *cyclic* when there exists a directed path from a node to itself. A directed and not cyclic graph is called a *directed acyclic graph*.

A Bayesian network (BN) represents random variables as the nodes of a directed acyclic graph. The probabilistic dependencies of the random variables are represented by the edges of such a graph. Let $P(x_1, \ldots, x_n)$ be the joint probability distribution of random variable X_i corresponding to nodes $V_i \in V$ in the directed acyclic graph $G = (V, E)$. The joint probability can be factorized according to the structure of the Bayesian network:

$$P(x_1, \ldots x_n) = \prod_{i=1}^{n} P(x_i \mid pa_i)$$

where pa_i represents the assignment of the random variables that correspond to parents of V_i. Throughout this paper, we focus specifically on *marginal inference*, which is concerned with the probability that a random variable X_n takes value x_n when marginalizing other variables out:

$$P(x_n) = \sum_{x_1} \cdots \sum_{x_{n-1}} P(x_1, \ldots x_n) = \sum_{x_1} \cdots \sum_{x_{n-1}} \prod_{i=1}^{n} P(x_i \mid pa_i).$$

1. The open-source implementation is available at https://github.com/sebastiaanbrand/bn-dd.

Using a similar expression, *conditional inference queries* can be performed. These queries compute the probability of a random variable assuming a specific value, given the observation of other random variables. In addition to observations, causal Bayesian networks distinguish themselves from non-causal Bayesian networks by their ability to facilitate causal interventions within the graph, achieved through the utilization of the *do-operator*. Therefore, the behaviour of the do-operator in the context of the Bayesian networks is also assumed, leading to a truncated factorization of the distribution (Vonk et al., 2023):

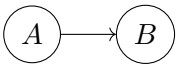
$$P(x_1, \dots x_{j-1}, x_{j+1} \dots x_n \mid do(x_j)) = \prod_{i \neq j} P(x_i \mid pa_i). \tag{1}$$

In case the conditional probability distributions resulting from the factorization are discrete, they can be expressed in the form of conditional probability tables (CPTs). Methods for estimating the CPTs from data are discussed in Section 4.

3. BDD Encoding and Inference

Binary decision diagrams (BDDs) (Bryant, 1986) are rooted directed acyclic graphs which represent Boolean functions $f : \{0,1\}^n \to \{0,1\}$, although by storing additional information outside the DD they can also be used to represent pseudo-Boolean functions $f : \{0,1\}^n \to \mathbb{R}$. Two important properties of BDDs are their ability to compactly represent many functions by identifying redundancies, and their support for efficient operations (i.e. polynomial-time in the size of the DD), such as computing marginal probabilities.

The joint probability distribution given by a BN is effectively a function of the form $f : \{0,1\}^n \to \mathbb{R}$ and can thus be encoded in a BDD. This is done by encoding each CPT

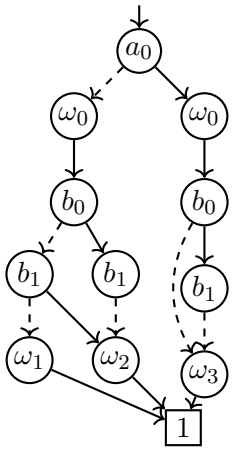


(a) BN structure

$P(A=0)$	$P(A=1)$
0.5	0.5

A	$P(B=0\|A)$	$P(B=1\|A)$	$P(B=2\|A)$
0	0.2	0.4	0.4
1	0.33	0.33	0.33

(b) Conditional probability tables



(c) BDD

ω_i	$\mathsf{val}(\omega_i)$
ω_0	0.5
ω_1	0.2
ω_2	0.4
ω_3	0.33

(d) ω_i values

Figure 2: An example Bayesian network (a,b) and the corresponding BDD (c). The actual probabilities corresponding to the Boolean variables ω_i are stored separately (d). In the BDD solid (dashed) edges correspond to positive (negative) assignments to the variables. For ease of visualization, in (c), all arrows pointing to the 0 leaf have been omitted.

entry in a small Boolean expression, from which a BDD can then be built using primitive BDD operations for logical and ($\wedge$), or ($\vee$), not ($\neg$), etc. As an example, consider the BN given in Figure 2(a)-2(b). To capture the (integer) values of A and B, Boolean variables $\{a_0, b_0, b_1\}$ are introduced, while unique probabilities are related to Boolean variables ω_i. As an example of the encoding of specific CPT entry, $P(B = 2 \mid A = 0) = 0.4$ is encoded as $(\neg a_0 \wedge b_1 \wedge \neg b_0) \Rightarrow \omega_2$, where $\neg a_0$ corresponds to $A = 0$ and $b_1 \wedge \neg b_0$ corresponds to $B = 2_{\text{dec}} = 10_{\text{bin}}$. The relationship $\mathsf{val}(\omega_2) = 0.4$ is stored outside of the BDD.

Computing marginal or conditional probabilities from a BDD that encodes a joint probability distribution can be done using so-called *weighted model counting* (Chavira and Darwiche, 2008). During weighted model counting the DD is traversed, relevant probabilities are gathered along the way, and each node is visited at most once, resulting in a computation time linear in the size of the BDD.

4. Discretization and Parameter Learning Methods

The discretization process serves to partition the state space Ω_{X_i} of a continuous random variable X_i into disjoint bins $\{B_j \mid j = 1, \ldots, m\}$ such that $\biguplus_j B_j = \Omega_{X_i}$. Every bin B_j is associated with a real number $f(B_j)$ denoting the value of the interval. In real-world applications, the state space of the random variable is unknown but is based on the sample data. The value associated with each bin B_j corresponds to the sample mean of the samples that are included in the bins, $\frac{1}{|B_j|} \sum_{x_i \in B_j} x_i$, in which $|B_j|$ denotes the number of $x_i \in B_j$.

The equal width (EW) discretization method partitions the state spaces Ω_{X_i} into bins of equal width. The equal frequency (EF) discretization approach divides the samples into quantiles. Both are unsupervised methods and require a parameter specifying the number of bins into which the original state space should be partitioned.

In addition to these two unsupervised discretization methods, we use four supervised discretization methods. First, we consider the entropy error-based approach, dynamic discretization (DDN) (Neil et al., 2007)[2], specifically developed for Bayesian network inference. Second, we employ minimum description length principle discretization (MDLP) (Fayyad and Irani, 1993), which iterates through potential cut-points recursively to minimize information entropy with respect to a chosen target variable. Third, we apply ChiMerge (CM) (Kerber, 1992), a discretization technique that continuously merges fine intervals based on the χ^2 statistic. Fourth, we use class-attribute interdependence maximization (CAIM) (Kurgan and Cios, 2004), which discretizes the continuous variables intending to maximize interdependency with the target variable (Ching et al., 1995). The latter three supervised discretization methods have been chosen because they performed well on a variety of discretization tasks (García et al., 2013).

Discretization of a continuous Bayesian network is followed by parameter learning, which involves the estimation of the CPTs. In this paper, we consider the maximum likelihood estimate (MLE) and the Bayesian method with adjusted empirical Bayes type 2 maximum likelihood priors (EBP) (Ji et al., 2015; Good, 1980). In the latter, the prior is initially estimated through MLE but refined by substituting 0 probability values with a minimal value (0.0001). This adjusted prior is subsequently used to infer the posterior CPTs with the

2. We use the implementation available at https://github.com/PCiunkiewicz/dynamic-discretization, adopting the parameter settings deemed most optimal by the implementator.

data. While the maximum likelihood estimates are sufficient to conduct inference on non-causal datasets, the causal datasets require the Bayesian approach to prevent any positivity violations (Zhu et al., 2023). The differences in results between both methods are discussed together with all the results of the experiments in the next section.

5. Experiments

We apply the methodology of Figure 1 to a variety of Bayesian networks. Section 5.1 introduces the measures used to evaluate the quality of the discretization or inference and the measure used to assess the computational cost of inference with decision diagrams. The different non-causal and causal Bayesian networks are specified in Section 5.2. Finally, Section 5.3 presents the results together with a list of the key findings.

5.1. Evaluation Measures

We start by discussing different measures to assess the quality of discretization and inference and continue with a measure to evaluate the computational cost.

5.1.1. Measuring the Quality of Discretization and Inference.

While f-divergences measure differences between probability distributions on the same measurable space (Sason, 2018), they are unsuitable for comparing a discretized state space and its continuous counterpart. Instead, we use the Wasserstein distance, specifically the Euclidean first-moment Wasserstein distance or *earth mover's distance* (EMD), to assess discretization quality as it is a common metric to compare (multivariate) distributions (Wang et al., 2021; Rubner et al., 2000; Applegate et al., 2011). The earth mover's distance quantifies the dissimilarity between two probability distributions by measuring the minimum "cost" to transform one distribution into the other. A high-quality discretization does not necessarily imply that a query of interest can be computed accurately. Fortunately, since the synthetic BNs in the experiments have specific distributions for which there exist exact inference methods, we have access to the conditional inference queries. To evaluate inference quality, we compare the conditional expected value of the original Bayesian network ($\mathbb{E}[Y \mid X]$) to its discretized counterpart ($\mathbb{E}_{disc}[Y \mid X]$) using the *weighted root mean squared error* (WRMSE), where the weights adjust for the probability of the conditioned-on variables. For the causal Bayesian networks, the *percentage error in the average treatment effect* (ATE) is being used, which is computed by means of interventional queries as in Equation 1. We refer the reader to Appendix C for a detailed description of these measures.

5.1.2. Measuring the Computational Cost of Inference.

As outlined in Section 3, inference using decision diagrams (DD) reduces to weighted model counting, which takes time linear in the size of the DD. Therefore, the number of nodes of the DD is considered to be a *proxy for the computational cost of inference*. Although DDs can potentially grow exponentially in the size of the Bayesian network, they typically remain smaller, enabling more scalable inference compared to traditional methods like vari-

Dataset	Kind	Variants	Samples	Network		Parents[4]		Inference
				Nodes	Edges	Max	Mean	Comparison
LG	Synthetic	36	100-5000	5	4	2	0.8	WRMSE
NM	Synthetic	8	100-500	2	1	1	0.5	WRMSE
CQ	Synthetic	1	2500	3	3	2	1	ATE
Lalonde	Real	1	2676	10	17	9	1.7	ATE
MC	Synthetic	1	4000	12	15	6	1.25	ATE
Arth	Real	1	1000	23[5]	28	6	1.22	None

Table 1: Characteristics of the Bayesian networks.

able elimination (VE) or belief propagation (BP).[3] Figure 1 details the methodology and implementation languages used.

The reported inference time for DDs includes *both* compilation and weighted model counting. The runtime of inference with traditional methods (dotted block on the left of Figure 1) is compared to the runtime of inference with DDs (both dashed blocks on the right of Figure 1). Since VE and BP are implemented in Python and weighted model counting in C++, comparing their runtimes directly is inappropriate. Instead, scalability is assessed by measuring the time speed-up (seconds) as the number of bins in the Bayesian network increases. The results are presented in Section 5.3.

5.2. Bayesian Network Description

The specifications of the non-causal and causal Bayesian networks, that are subject to experimentation, are introduced in this section. A summary of their characteristics can be found in Table 1.

Linear Gaussian (LG) Bayesian network. Samples are drawn from a linear Gaussian Bayesian network (Ostwal, 2020) with random variables A, B, C, D, E. In total, 36 experiments were conducted, varying in sample size (N) and distribution parameters. To ensure a balanced experimental design, Sobol sequences were employed (Garud et al., 2017). Detailed experimental specifications are provided in Tables 4 and 5 of Appendix B. The computational cost in terms of the number of nodes in the DD is drawn against the WRMSE and against the earth mover's distance in Figure 4(a), 4(c) and Figure 4(b), respectively.

Normal mixture (NM) Bayesian network. Samples from a normal mixture Bayesian network are drawn using a two-node Gaussian mixture model. In this network, A follows a Bernoulli distribution and $P(B|A)$ is Gaussian, based on (Neil et al., 2007). Details on sample sizes and distribution parameters are listed in Table 6 of Appendix B.

3. We used the Python implementation of pgmpy for VE and BP (Ankan and Panda, 2015).
4. The maximum and mean number of parents (also called maximum and mean *in-degree*) are proxies for computational cost of inference.
5. The reported node and edge size pertain to the pruned Bayesian network.

Causal quadratic (CQ) Bayesian network. Data is sampled from a quadratic data-generating process (DGP). In this DGP, the confounder Z is distributed normally and has a quadratic effect on outcome variable Y while also affecting treatment variable T. For the full specifications of the distribution of the quadratic DGP (Parikh et al., 2022), the reader is referred to Appendix B. The computational cost of inference has been set out against the percentage error of the ATE in the Pareto front of Figure 4(d).

Lalonde causal Bayesian network. The Lalonde causal dataset is a real causal dataset where the effect of temporary employment on income is studied (LaLonde, 1986) given confounding variables. Since an observational (Dehejia and Wahba, 1999) as well as an experimental dataset (LaLonde, 1986) is available, we can compare our non-parametric estimates of the average treatment effect with the difference in means in the observational and experimental datasets. The comparative analysis of computational cost of inference is presented alongside the percentage error of the ATE in the Pareto front depicted in Figure 4(e).

Mixed Confounding (MC) Bayesian network. Samples are drawn from a mixed confounding dataset, characterized by both continuous and discrete variables that influence multiple variables in the graph in a non-linear way, as outlined in detail in the Csuite benchmarking causal datasets (Geffner and et al., 2022). Figure 4(f) draws the computational cost of inference against the percentage error of the ATE within a Pareto front.

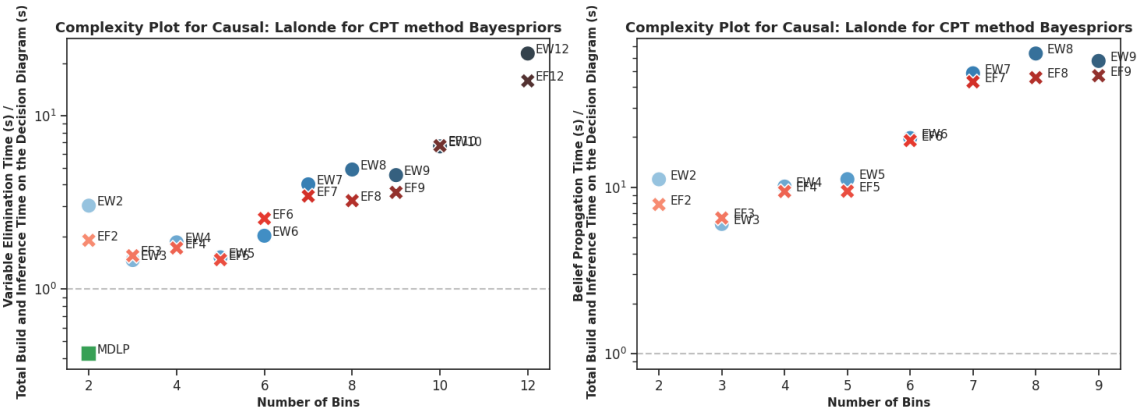
Arth Bayesian network This vast Bayesian network, sourced from the GeneNet package and featured in bnlearn, consists of plant expression data (Opgen-Rhein and Strimmer, 2007). Due to the enormous size of the network and our practical resource constraint of 64GB RAM, we have compiled a computationally equivalent pruned version of the network (Baker and Boult, 2013). Given the absence of a ground truth for conditional queries, the network is solely assessed based on the earth mover's distance.

5.3. Results

First, the speed-up scalability results are introduced. This is followed by some Pareto fronts epitomizing the trade-off between computational cost and discretization/inference quality. Finally, the key findings are presented.

5.3.1. Scalability of Decision Diagrams for Inference

In Figure 3, we have compared the speedup of Bayesian network inference via decision diagrams to Bayesian network inference with variable elimination (VE) (Figure 3(a)) and belief propagation (BP) (Figure 3(b)) for the Lalonde Bayesian network. The Lalonde network has been chosen since it has the highest maximum in-degree, a *proxy* for the computational cost of inference. The fact that inference with decision diagrams becomes at least over 10 times faster than VE or BP as the number of bins increases underscores a notable improvement in scalability (in fact, for BP this is true for over 5 bins).



(*a*) BDD speedup versus VE (*b*) BDD speedup versus BP

Figure 3: The speedup plots for using decision diagrams as opposed to VE (3(*a*)) or BP (3(*b*)) for inference for the Lalonde experiment. The red crosses refer to EF binning, the blue circles represent the EW binning and the MDLP binning is indicated by a green square. As the number of bins increases, using decision diagrams is more than 10 times as fast as both VE and BP.

5.3.2. General Results

While all Pareto fronts are available at Zenodo[6], a representative selection across all Bayesian networks and errors is highlighted in Figure 4. These Pareto fronts clearly demonstrate that increasing the number of bins results in a reduction of the earth mover's distance but an increase in computational cost. Simultaneously, the WRMSE and the percentage error of the ATE decrease as the number of bins rises, up to a certain number of bins.

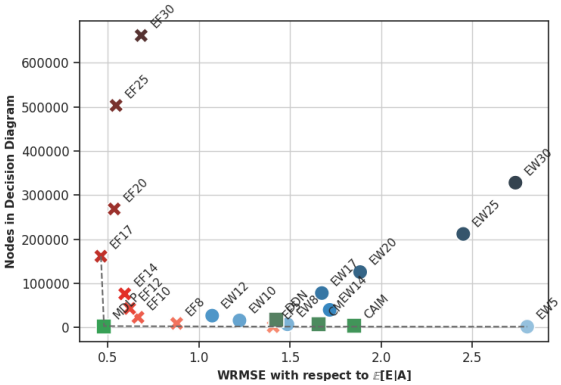
To facilitate the interpretation of results across various experiments, the Pareto fronts have been condensed into the heatmaps presented in Tables 2 and 3 of Appendix A. All the experiments yield the following four key findings.

First, the solutions with the lowest earth mover's distance to the original BN are the most-binned solutions as can be observed in Figure 4(*b*). In general, we observe that the earth mover's distance decreases when the number of bins used to discretize the BN increases, but the distance reduction becomes lower as the number of bins grows larger.

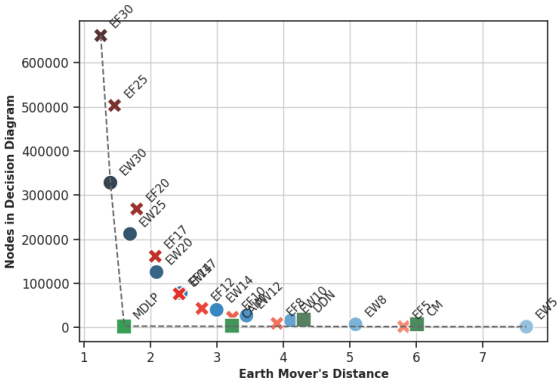
Second, the difference between inference results when estimating the CPTs with maximum likelihood estimate or the Bayesian method with adjusted empirical Bayes type 2 maximum likelihood priors is negligible. This similarity is evident from the plots in Figure 4(*a*) and 4(*c*), and supported by the data in Table 2 in Appendix A. Additional discrepancy plots on Zenodo further illustrate this negligible difference.

Third, the WRMSE and the PE decrease when adding bins up to a certain number of bins whereafter it increases again, indicating overfitting in data-sparse areas of the root variable. The Pareto fronts of Figure 4(*c*), 4(*d*), 4(*e*) and 4(*f*) show that the bending point differs per experiment. Generally, more available samples or simpler BN structures lead to the solution with the lowest error being often a more intensely-binned solution.

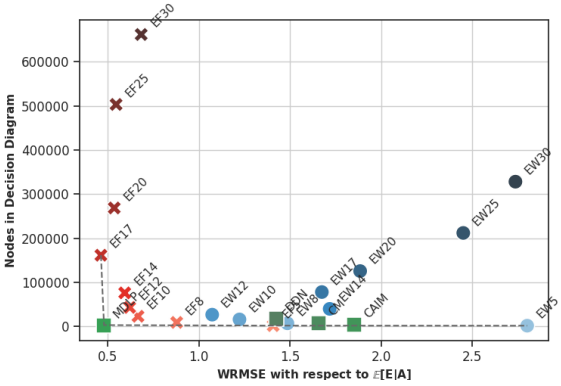
6. https://zenodo.org/record/11202314



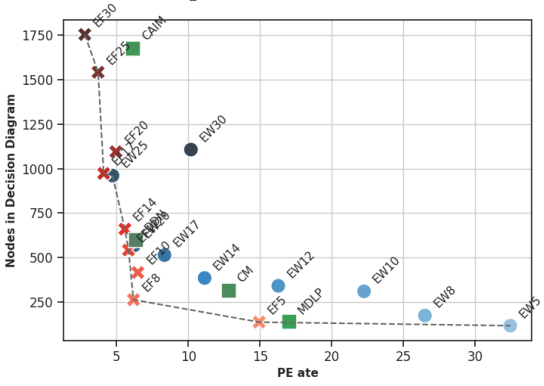
(*a*) WRMSE for the linear Gaussian experiment 9 with CPT method MLE.



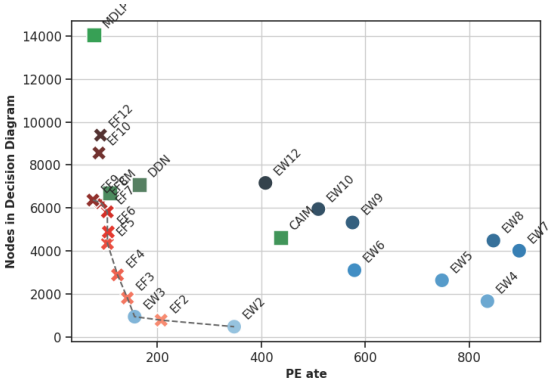
(*b*) Earth mover's distance for the linear Gaussian experiment 9



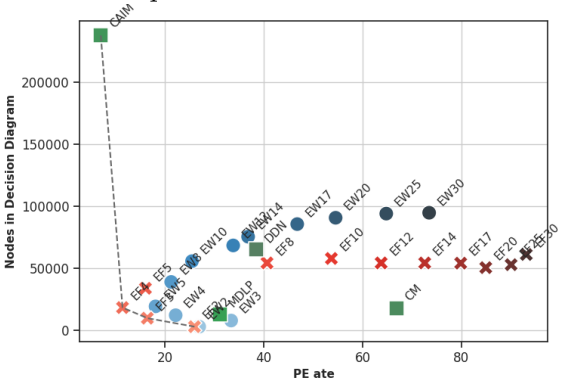
(*c*) WRMSE for the linear Gaussian experiment 9 with CPT method EBP.



(*d*) Percentage error of the ATE for the causal quadratic DGP



(*e*) Percentage error of the ATE for the Lalonde dataset



(*f*) Percentage error of the ATE for the mixed confounding dataset

Figure 4: Number of nodes in the BDDs versus various evaluation measures for several discretization approaches with different parameter settings, per dataset. Approaches representing trade-offs between objectives (axes, both to be minimized) lie on the Pareto front (dashed line).

Finally, it can incidentally be observed that one of the supervised discretization methods dominates the other solutions (Figure 4(f)). However, no supervised discretization method performs exceptionally well across all experiments on the considered measures.

6. Discussion

This paper presented a method for performing inference on continuous Bayesian networks using discretization and a computationally efficient knowledge compilation approach on decision diagrams. Unlike previous evaluations of discretizing methods of BNs (Nojavan et al., 2017; Beuzen et al., 2018), we applied many supervised and unsupervised discretization methods to a diverse set of non-causal and causal BNs. Thereby, we have explored the trade-off between the computational cost of inference and the quality of the discretization in terms of distance and inference results

Our contribution is threefold. First, our research underscores the significant scalability advantage of inference through knowledge compilation with decision diagrams, which becomes over 10 times faster as the number of bins increases, compared to traditional approaches. Second, not only do our findings confirm that increasing the number of bins reduces the earth mover's distance, they also highlight that the required number of bins needed to minimize errors in inference queries depends on the sample size and the complexity of the BN structure. Lastly, we explore addressing positivity violations in Bayesian networks by estimating CPTs with adjusted empirical Bayes priors, achieving results comparable to those obtained through maximum likelihood estimation, thus extending applicability to causal networks

We propose the following three avenues for future research: given the diverse origins of the continuous BNs used in our studies (Geffner and et al., 2022; Scutari, 2010), there is a pressing need for a standardized set of continuous BNs, complete with ground truths, to serve as benchmarks across the research domain. Additionally, while our work has primarily focused on BDDs, investigating the effects of discretization methods on other types of decision diagrams, such as Weighted Positive Binary Decision Diagrams (WPBDDs) and Affine Algebraic Decision Diagrams (AADDs), could offer valuable insights. Finally, the proposed methodology could be benchmarked against other known approximate inference methods for continuous Bayesian networks, such as sampling methods or variational inference (Koller and Friedman, 2009).

7. Conclusion

As demonstrated in this paper, discretizing continuous Bayesian networks does not necessarily lead to the explosion of computational cost nor to an extreme loss of accuracy. Therefore, we encourage researchers to carefully consider discretization as a viable option before making parametric assumptions in the network.

References

A. Ankan and A. Panda. pgmpy: Probabilistic graphical models using python. In *Proceedings of the 14th Python in Science Conference (SCIPY 2015)*. Citeseer, 2015.

D. Applegate, T. Dasu, S. Krishnan, and S. Urbanek. Unsupervised clustering of multi-dimensional distributions using earth mover distance. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 636–644, 2011.

M. Baker and T. E. Boult. Pruning bayesian networks for efficient computation. *arXiv preprint arXiv:1304.1112*, 2013.

T. Beuzen, L. Marshall, and K. D. Splinter. A comparison of methods for discretizing continuous variables in bayesian networks. *Environmental modelling & software*, 108: 61–66, 2018.

R. E. Bryant. Graph-based algorithms for Boolean function manipulation. *Transactions on Computers*, 100(8):677–691, 1986.

M. Chavira and A. Darwiche. On probabilistic inference by weighted model counting. *Artificial Intelligence*, 172(6-7):772–799, 2008.

J. Y. Ching, A. K. C. Wong, and K. C. C. Chan. Class-dependent discretization for inductive learning from continuous and mixed-mode data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(7):641–651, 1995.

G. H. Dal and P. J. Lucas. Weighted positive binary decision diagrams for exact probabilistic inference. *International Journal of Approximate Reasoning*, 90:411–432, 2017.

G. H. Dal, A. W. Laarman, and P. J. Lucas. Parallel probabilistic inference by weighted model counting. In *International Conference on Probabilistic Graphical Models*, pages 97–108. PMLR, 2018.

G. H. Dal, A. W. Laarman, A. Hommersom, and P. J. Lucas. A compositional approach to probabilistic knowledge compilation. *International Journal of Approximate Reasoning*, 138:38–66, 2021. ISSN 0888-613X. doi: https://doi.org/10.1016/j.ijar.2021.07.007. URL https://www.sciencedirect.com/science/article/pii/S0888613X21001122.

A. Darwiche. A logical approach to factoring belief networks. *KR*, 2:409–420, 2002.

A. Darwiche and P. Marquis. A knowledge compilation map. *Journal of Artificial Intelligence Research*, 17:229–264, 2002.

R. H. Dehejia and S. Wahba. Causal effects in nonexperimental studies: Reevaluating the evaluation of training programs. *Journal of the American statistical Association*, 94(448): 1053–1062, 1999.

D.-J. Delgado-Hernández, O. Morales-Nápoles, D. De-León-Escobedo, and J.-C. Arteaga-Arcos. A continuous bayesian network for earth dams' risk assessment: an application. *Structure and Infrastructure Engineering*, 10(2):225–238, 2014.

U. M. Fayyad and K. B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *International Joint Conference on Artificial Intelligence*, 1993.

S. García, J. Luengo, J. A. Sáez, V. López, and F. Herrera. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 25(4):734–750, 2013. doi: 10.1109/TKDE.2012.35.

S. S. Garud, I. A. Karimi, and M. Kraft. Design of computer experiments: A review. *Computers & Chemical Engineering*, 106:71–95, 2017.

T. Geffner and et al. Deep end-to-end causal inference. *arXiv preprint arXiv:2202.02195*, 2022.

I. J. Good. Some history of the hierarchical bayesian methodology. *Trabajos de estadística y de investigación operativa*, 31:489–519, 1980.

M. Jaeger. Probabilistic decision graphs—combining verification and ai techniques for probabilistic inference. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 12(supp01):19–42, 2004.

Z. Ji, Q. Xia, and G. Meng. A review of parameter learning methods in bayesian network. In *Advanced Intelligent Computing Theories and Applications: 11th International Conference, ICIC 2015, Fuzhou, China, August 20-23, 2015. Proceedings, Part III 11*, pages 3–12. Springer, 2015.

F. Johansson and G. Falkman. A bayesian network approach to threat evaluation with application to an air defense scenario. In *2008 11th International Conference on Information Fusion*, pages 1–7, 2008.

R. A. Kelly (Letcher), A. J. Jakeman, O. Barreteau, M. E. Borsuk, S. ElSawah, S. H. Hamilton, H. J. Henriksen, S. Kuikka, H. R. Maier, A. E. Rizzoli, H. van Delden, and A. A. Voinov. Selecting among five common modelling approaches for integrated environmental assessment and management. *Environmental Modelling & Software*, 47:159–181, 2013. ISSN 1364-8152. doi: https://doi.org/10.1016/j.envsoft.2013.05.005. URL https://www.sciencedirect.com/science/article/pii/S1364815213001151.

R. Kerber. Chimerge: Discretization of numeric attributes. In *Proceedings of the tenth national conference on Artificial intelligence*, pages 123–128, 1992.

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

L. Kurgan and K. Cios. Caim discretization algorithm. *IEEE transactions on knowledge and data engineering*, 16(2):145–153, 2004. ISSN 1041-4347.

R. J. LaLonde. Evaluating the econometric evaluations of training programs with experimental data. *The American economic review*, pages 604–620, 1986.

O. Morales-Nápoles and R. D. Steenbergen. Large-scale hybrid bayesian network for traffic load modeling from weigh-in-motion system data. *Journal of Bridge Engineering*, 20(1): 04014059, 2015.

J. Mori and V. Mahalec. Inference in hybrid bayesian networks with large discrete and continuous domains. *Expert Systems with Applications*, 49:1–19, 2016.

M. Neil, M. Tailor, and D. Marquez. Inference in hybrid bayesian networks using dynamic discretization. *Statistics and Computing*, 17(3):219–233, 2007.

F. Nojavan, S. S. Qian, and C. A. Stow. Comparative analysis of discretization methods in bayesian networks. *Environmental Modelling & Software*, 87:64–71, 2017.

R. Opgen-Rhein and K. Strimmer. From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data. *BMC systems biology*, 1(1):1–10, 2007.

P. Ostwal. ostwalprasad/lgnpy: v1.0.0, 2020. URL https://zenodo.org/record/3902122.

H. Parikh, C. Varjao, L. Xu, and E. T. Tchetgen. Validating causal inference methods. In *International Conference on Machine Learning*, pages 17346–17358. PMLR, 2022.

Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99, 2000.

R. Rumí and A. Salmerón. Approximate probability propagation with mixtures of truncated exponentials. *International Journal of Approximate Reasoning*, 45(2):191–210, 2007.

T. Sang, F. Bacchus, P. Beame, H. A. Kautz, and T. Pitassi. Combining component caching and clause learning for effective model counting. *SAT*, 4:7th, 2004.

T. Sang, P. Beame, and H. Kautz. Heuristics for fast exact model counting. In *Theory and Applications of Satisfiability Testing: 8th International Conference, SAT 2005, St Andrews, UK, June 19-23, 2005. Proceedings 8*, pages 226–240. Springer, 2005a.

T. Sang, P. Beame, and H. Kautz. Solving bayesian networks by weighted model counting. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05)*, volume 1, pages 475–482. AAAI Press, 2005b.

I. Sason. On f-divergences: Integral representations, local behavior, and inequalities. *Entropy*, 20(5):383, 2018.

M. Scutari. Learning bayesian networks with the bnlearn R package. *Journal of Statistical Software*, 35(3):1–22, 2010. doi: 10.18637/jss.v035.i03.

P. P. Shenoy and J. C. West. Inference in hybrid bayesian networks using mixtures of polynomials. *International Journal of Approximate Reasoning*, 52(5):641–657, 2011. doi: 10.1016/j.ijar.2010.09.003.

C. Su, A. Andrew, M. R. Karagas, and M. E. Borsuk. Using bayesian networks to discover relations between genes, environment, and disease. *BioData mining*, 6(1):1–21, 2013.

M. C. Vonk, N. Malekovic, T. Bäck, and A. V. Kononova. Disentangling causality: assumptions in causal discovery and inference. *Artificial Intelligence Review*, pages 1–37, 2023.

J. Wang, P. Wang, and P. Shafto. Efficient discretizations of optimal transport. *arXiv preprint arXiv:2102.07956*, 2021.

A. Y. Zhu, N. Mitra, and J. Roy. Addressing positivity violations in causal effect estimation using gaussian process priors. *Statistics in Medicine*, 42(1):33–51, 2023.

Appendix A. Heatmap Results

Table 2: Heatmap summarizing all the non-causal results.

Error measure		EMD	WRMSE			
CPT method		NA	EBP	MLE	EBP	MLE
Binning method		All	EF	EF	EW	EW
Experiment	N					
LG9	5000	EF30	EF17	EF17	EW12	EW12*
LG10	3000	EF30	EF17*	EF17*	EW10*	EW10*
LG11	2000	EF30	EF12*	EF12*	EW8*	EW8*
LG12	1000	EF30	EF12	EF12	EW8*	EW8*
LG13	800	EF30	EF12	EF12	EW5*	EW5*
LG14	600	EW30	EF8	EF8	EW8*	EW8*
LG15	500	EW30	EF8	EF8	EW8*	EW8*
LG16	400	EW30	EF8	EF8	EW5*	EW5*
LG17	300	EW30	EF8	EF8	EW5*	EW5*
LG18	200	EW30	EF5	EF5	EW5-	EW5-
LG19	100	EW30	EF5	EF5	EW5-	EW5-
LG101	100	EW30	EF5	EF5	EW8*	EW8*
LG102	1050	EF30	EF12	EF12	EW8	EW8
LG103	1525	EF30	EF12	EF12	EW10*	EW10*
LG104	575	EF30	EF5*	EF5*	EW5*	EW5*
LG105	812	EF30	EF12	EF12	EW8*	EW8*
LG106	1762	EF30	EF17	EF17	EW8*	EW8*
LG107	1288	EF30	EF12	EF12	EW8*	EW8*
LG108	338	EW30	EF8	EF8	EW5-	EW5-
LG109	456	EW30	EF8	EF8	EW5*	EW5*
LG110	1406	EF30	EF10	EF10	EW8-	EW8-
LG111	1881	EF30	EF10*	EF10*	EW12*	EW12*
LG112	931	EF30	EF10	EF10	EW5*	EW5*
LG113	694	EW30	EF10	EF10	EW8*	EW8*
LG114	1644	EW30	EF14	EF14	EW8-	EW8-
LG115	1169	EF30	EF12	EF12	EW10*	EW10*
LG116	219	EW30	EF5	EF5	EW5-	EW5-
LG117	278	EW30	EF5	EF5	EW5-	EW5-
LG118	1228	EF30	EF10	EF10	EW8*	EW8*
LG119	1703	EF30	EF17	EF17	EW10*	EW10*
LG120	753	EF30	EF12	EF12	EW8-	EW8-
LG121	991	EF30	EF12*	EF12*	EW8*	EW8*
LG122	1941	EF30	EF10	EF10	EW8*	EW8*
LG123	1466	EF30	EF12	EF12	EW8*	EW8*
LG124	516	EW30	EF10	EF10	EW5*	EW5*
LG125	397	EW30	EF8	EF8	EW5*	EW5*
NM1	500	EW30+	EF25+	EF25+	EW20+	EW20+
NM2	500	EF30+	EF30+	EF30+	EW30+	EW30+
NM3	500	EF30+	EF25+	EF25+	EW17+	EW17+
NM4	500	EF30+	EF30+	EF30+	EW25+	EW25+
NM5	100	EF30+	EF8+	EF8+	EW17+	EW17+
NM6	100	EF30+	EF30+	EF30+	EW25+	EW25+
NM7	100	EF30+	EF17+	EF17+	EW30+	EW30+
NM8	100	EF25+	EF25+	EF25+	EW30+	EW30+
Arth	1000	EW30				

Table 3: Heatmap summarizing the causal results: every box refers to a Pareto front corresponding to discretization methods EF and EB, evaluation measure EMD, WRMSE and PE ATE, and inferring CPT method MLE or EBP. The color indicates the number of bins in the best approach for the corresponding experiment with respect to the chosen evaluation measure: light blue stands for a small number of bins and dark blue means a large number of bins. In the heatmap, a star indicates MDLP dominance over all solutions for that binning strategy, a plus signifies CAIM dominance, a minus denotes ChiMerge dominance and a tilde denotes dynamic discretization dominance.

Error measure		EMD	PE ATE	
CPT method		All	EBP	
Binning method		–	EF	EW
Experiment	N			
CQ DGP	2500	EF30	EF30	EW25
Lalonde	2675	EF12	EF9*	EW3*
MC	4000	EW30	EF4+	EW5+

As the computational cost becomes generally higher when the number of bins in the discretization process increases, these heatmaps focus on the quality of discretization and inference. For example, the discretization EF9 in Pareto front of Figure 4(e) dominates all other EF discretizations in terms of the error. Therefore, EF9 returns in the heatmap of Table 3 in the corresponding error and CPT inference method column.

Appendix B. Experimental Set-up

The experimental setup outlines the specifications of each of the experiments in terms of their distribution and sample size.

Table 4: The first 25 experiments involving linear Gaussian BNs parameterized by Sobol sequences for the number of samples N and standard deviations. Each variable follows a normal distribution $\mathcal{N}(\mu, \sigma)$, with mean μ specified in the header and standard deviation σ listed in the table.

Experiment	N	$P(A)$ $\mu = 20$ σ	$P(B)$ $\mu = 20$ σ	$P(C)$ $\mu = 15$ σ	$P(D \mid A, B)$ $\mu = 2A + 3B$ σ	$P(E \mid C, D)$ $\mu = 3C + 3D$ σ
LG101	100	1	1	1	1	1
LG102	1050	5.5	5.5	5.5	5.5	5.5
LG103	1525	3.25	3.25	3.25	3.25	3.25
LG104	575	7.75	7.75	7.75	7.75	7.75
LG105	813	4.38	4.38	4.38	4.38	4.38
LG106	1763	8.88	8.88	8.88	8.88	8.88
LG107	1288	2.13	2.13	2.13	2.13	2.13
LG108	338	6.63	6.63	6.63	6.63	6.63
LG109	456	3.81	3.81	3.81	3.81	3.81
LG110	1406	8.31	8.31	8.31	8.31	8.31
LG111	1881	1.56	1.56	1.56	1.56	1.56
LG112	931	6.06	6.06	6.06	6.06	6.06
LG113	694	2.69	2.69	2.69	2.69	2.69
LG114	1644	7.19	7.19	7.19	7.19	7.19
LG115	1169	4.94	4.94	4.94	4.94	4.94
LG116	219	9.44	9.44	9.44	9.44	9.44
LG117	278	5.22	5.22	5.22	5.22	5.22
LG118	1228	9.72	9.72	9.72	9.72	9.72
LG119	1703	2.97	2.97	2.97	2.97	2.97
LG120	754	7.47	7.47	7.47	7.47	7.47
LG121	991	1.84	1.84	1.84	1.84	1.84
LG122	1941	6.34	6.34	6.34	6.34	6.34
LG123	1466	4.09	4.09	4.09	4.09	4.09
LG124	516	8.59	8.59	8.59	8.59	8.59
LG125	397	2.41	2.41	2.41	2.41	2.41

Table 5: Number of samples and parametrization of the experiments with the 11 extra linear Gaussian Bayesian networks. These experiments are meant to isolate the effect of the sample size on the Pareto front. Note that the samples in lower sample-sized experiments are contained in the samples of experiments with higher sample sizes.

$P(A)$	$P(B)$	$P(C)$	$P(D \mid A, B)$	$P(E \mid C, D)$
$\mathcal{N}(20, 2)$	$\mathcal{N}(20, 2)$	$\mathcal{N}(15, 2)$	$\mathcal{N}(2A + 3B, 2)$	$\mathcal{N}(3C + 3D, 2)$

(a) Parametrization of all of the experiments

Experiment	**LG9**	**LG10**	**LG11**	**LG12**	**LG13**	**LG14**	**LG15**	**LG16**	**LG17**	**LG18**	**LG19**
N	5000	3000	2000	1000	800	600	500	400	300	200	100

(b) Name and sample size of experiments

Table 6: Number of samples and parametrization of the experiments with the Normal mixture model

Experiment	N	$P(A)$	$P(B \mid A = 1)$	$P(B \mid A = 0)$
NM1	500	$B(1, \frac{1}{2})$	$\mathcal{N}(21, 10)$	$\mathcal{N}(25, 1)$
NM2	500	$B(1, \frac{4}{5})$	$\mathcal{N}(21, 10)$	$\mathcal{N}(25, 1)$
NM3	500	$B(1, \frac{1}{2})$	$\mathcal{N}(6, 2)$	$\mathcal{N}(4, 2)$
NM4	500	$B(1, \frac{4}{5})$	$\mathcal{N}(6, 2)$	$\mathcal{N}(4, 2)$
NM5	100	$B(1, \frac{1}{2})$	$\mathcal{N}(21, 10)$	$\mathcal{N}(25, 1)$
NM6	100	$B(1, \frac{4}{5})$	$\mathcal{N}(21, 10)$	$\mathcal{N}(25, 1)$
NM7	100	$B(1, \frac{1}{2})$	$\mathcal{N}(6, 2)$	$\mathcal{N}(4, 2)$
NM8	100	$B(1, \frac{4}{5})$	$\mathcal{N}(6, 2)$	$\mathcal{N}(4, 2)$

The following causal quadratic (CQ) linear experiment is adopted from (Parikh et al., 2022):

$$
\begin{aligned}
&X_i \sim \mathcal{N}(0, 1) \\
&Y_i(0) = \beta^T X_i + \epsilon_0 && \text{where } \epsilon_0 \sim \mathcal{N}(0, 1) \\
&Y_i(1) = Y_i(0)^2 + \alpha^T X_i + \epsilon_1 && \text{where } \epsilon_1 \sim \mathcal{N}(0, 1) \\
&T_i = \text{expit}(\mathbf{1}^T X_i))
\end{aligned}
$$

A maximum number of 30 bins was allowed for all experiments except for the Lalonde dataset (12 bins maximum).

Appendix C. Evaluation Measures

We compare the conditional expected value of the target variable $\mathbb{E}[Y \mid X]$ with respect to conditioning on one of the root nodes X of the original Bayesian network with its counterpart on the discretized Bayesian network, denoted by $\mathbb{E}_{disc}[Y \mid X]$. In order to compensate for the probability of the conditioned-on variable, we evaluate the accuracy of the discretized conditional expected value by means of the weighted root mean squared error (WRMSE):

$$WRMSE = \sqrt{\sum_x P(X = x)(\mathbb{E}[Y \mid X = x] - \mathbb{E}_{disc}[Y \mid X = x])^2}$$

where $P(X = x)$ denotes the discretized probability that X takes value x. Note that the number of values involved in WRMSE depends on the discretization of the root node X.

For the causal Bayesian networks in Section 5.2, we have access to the true average treatment effect:

$$\begin{aligned} ATE &= \mathbb{E}[Y \mid do(T = 1)] - \mathbb{E}[Y \mid do(T = 0)] \\ &= \mathbb{E}_Z[\mathbb{E}[Y \mid T = 1, Z] - \mathbb{E}[Y \mid T = 0, Z]] \end{aligned}$$

for confounding variables Z, treatment variable T and target variable Y. Therefore, we consider the percentage error (PE) of the average treatment effect (ATE) as the object of investigation:

$$PE = 100 \times \left| \frac{ATE_{true} + ATE_{disc}}{ATE_{true}} \right|.$$