On conditional truncated densities Bayesian networks

Santiago Cortijo, Christophe Gonzales

On Conditional Truncated Densities Bayesian Networks

Santiago Cortijo, Christophe Gonzales

Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6, UMR 7606, Paris, France.
firstname.lastname@lip6.fr

Abstract

The majority of Bayesian networks learning and inference algorithms rely on the assumption that all random variables are discrete, which is not necessarily the case in real-world problems. In situations where some variables are continuous, a trade-off between the expressive power of the model and the computational complexity of inference has to be done: on one hand, conditional Gaussian models are computationally efficient but they lack expressive power; on the other hand, mixtures of exponentials (MTE), basis functions (MTBF) or polynomials (MOP) are expressive but this comes at the expense of tractability. In this paper, we introduce an alternative model called a ctdBN that lies in between. It is composed of a "discrete" Bayesian network (BN) combined with a set of univariate conditional truncated densities modeling the uncertainty over the continuous random variables given their discrete counterpart resulting from a discretization process. We prove that ctdBNs can approximate (arbitrarily well) any Lipschitz mixed probability distribution. They can therefore be exploited in many practical situations. An efficient inference algorithm is also provided and its computational complexity justifies theoretically why inference computation times in ctdBNs are very close to those in discrete BNs. Experiments confirm the tractability of the model and highlight its expressive power, notably by comparing it with BNs on classification problems and with MTEs and MOPs on marginal distributions estimations.

Keywords: Bayesian network, continuous random variable, mixed probability distribution, inference

1. Introduction

For several decades, Bayesian networks (BN) [1] have been successfully exploited for dealing with uncertainties. Their popularity has stimulated the

development of many efficient learning and inference algorithms [2, 3, 4, 5, 6]. Whilst these algorithms are relatively well understood when they involve only discrete variables, their ability to cope with continuous variables is often unsatisfactory. Dealing with continuous random variables is much more complicated than dealing with discrete ones and one actually has to trade-off between the expressive power of the uncertainty model and the computational complexity of its learning and inference mechanisms. Conditional Gaussian models and their mixing with discrete variables [7, 8, 9] lie on one side of the spectrum. They compactly represent multivariate Gaussian distributions. Their inference mechanisms are computationally very efficient but their main drawback is that they lack expressive power. Indeed, although Conditional Linear Gaussian (CLG) models can easily encode conditional independences between random variables, the density functions of their continuous random variables are required to be Normal distributions whose parameters depend linearly on the values of their parents. They are therefore unable to represent models where dependences between the continuous random variables are nonlinear. In addition, they are not very well suited to represent models in which random variables are not distributed w.r.t. Normal distributions. On the other side of the spectrum, there are more expressive models like mixtures of exponentials (MTE) [10, 11, 12], mixtures of truncated basis functions (MTBF) [13, 14] and mixtures of polynomials (MOP) [15, 16, 17]. Those can approximate very well density functions but this comes at the expense of tractability: their exact inference computation times tend to grow exponentially with the number of continuous variables, which makes them unusable when they contain hundreds of random variables.

In this paper, we propose an alternative model that lies in between these two extremes. The key idea is to discretize the random variables, thereby mapping each (continuous) value of their domain into an interval within a *finite* set of intervals. Of course, whenever some discretization is performed, some information about the continuous random variables is lost. But this can be significantly alleviated by modeling the distribution of the continuous values within each discretization interval by a density function which may not necessarily be a uniform distribution (which is the implicit assumption when using a classical discretization). The set of density functions over all the intervals of a continuous variable constitutes its "*conditional truncated density*" given its discretized counterpart. Now, our uncertainty model is a (discrete) BN over the set of discrete and discretized random variables combined with the set of conditional truncated densities assigned to the continuous random variables that were discretized. This model model

is therefore called "*conditional truncated densities Bayesian network*", or ctdBN for short. It represents compactly mixed probability distributions. The model is derived from the result of an algorithm for learning BNs from datasets containing both discrete and continuous random variables [18].

By assigning conditional truncated densities to continuous variables, our model gains expressive power over a BN in which all continuous variables are discretized. As we show in this paper, this assertion is justified theoretically by the fact that any Lipschitz mixed probability distribution can be (arbitrarily well) approximated by a ctdBN. For inference, the density functions need only be included in the BN as discrete evidence (computed by integrations) over the discretized variables and, then, only a classical inference over discrete variables is needed to complete the process. As, in our model, the density functions are univariate, integrations can be performed efficiently. So the inference times are very close to those of inferences in classical BNs, which makes inference tractable in ctdBNs. The theoretical computational complexity of our inference algorithm supports this assertion. In addition, the experiments performed in the paper also highlight this point as well as the expressive power of the model.

The paper is organized as follows. In the next section, we recall some related works on CLGs, MTEs, MTBFs and MOPs. Then, in Section 3, we present our model, we study theoretically its expressive power, i.e., its capacity to approximate mixed probability distributions, and we propose an inference algorithm as well as its computational complexity. Next, the efficiency and effectiveness of ctdBN's inferences are highlighted through a set of experiments. Finally, a conclusion and some perspectives are provided in the last section.

2. Related Works

In the rest of the paper, capital letters (possibly subscripted) refer to random variables and boldface capital letters to sets of variables. To distinguish continuous random variables from discrete ones, we denote by $\mathring{X}_i$ a continuous variable and by X_i a discrete one. Without loss of generality, for any $\mathring{X}_i$, variable X_i represents its discretized counterpart. Throughout the paper, let $\mathbf{X_D} = \{X_1, \ldots, X_d\}$ and $\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{X}_{d+1}, \ldots, \mathring{X}_n\}$ denote the set of discrete and continuous random variables respectively. We denote by $\mathcal{X} = \mathbf{X_D} \cup \mathring{\mathbf{X}}_\mathbf{C}$ the set of all random variables. In addition, for any set of indices $\mathbf{I} = \{i_1, \ldots, i_k\}$, $\mathbf{X_I}$ denotes the set of random variables $\{X_{i_1}, \ldots, X_{i_k}\}$. Finally, for any variable X or set of random variables $\mathbf{Y}$ or $\mathring{\mathbf{Y}}$, let Ω_X (resp. $\Omega_\mathbf{Y}$ or $\Omega_{\mathring{\mathbf{Y}}}$) denote the domain of X (resp. $\mathbf{Y}$ or $\mathring{\mathbf{Y}}$).

As mentioned in the introduction, a Conditional Linear Gaussian (CLG) model represents a mixed probability distribution [7]. Like in a BN, in the CLG model, to each discrete variable X_i in $\mathbf{X_D}$ is assigned its conditional probability table (CPT) $P(X_i|\mathbf{Pa}(X_i))$ given its parents (the latter are all discrete). In addition, to each continuous variable $\mathring{X}_i \in \mathring{\mathbf{X}}_\mathbf{C}$ is assigned the conditional distribution:

$$f(\mathring{x}_i|\mathbf{X_{D_i}} = \mathbf{x_{D_i}}, \mathring{\mathbf{X}}_\mathbf{C_i} = \mathring{\mathbf{x}}_\mathbf{C_i}) = \mathcal{N}(\mathring{x}_i|\alpha(\mathbf{x_{D_i}}) + \beta(\mathbf{x_{D_i}})^T \mathring{\mathbf{x}}_\mathbf{C_i}, \sigma(\mathbf{x_{D_i}})),$$

where $\mathbf{X_{D_i}}$ and $\mathring{\mathbf{X}}_\mathbf{C_i}$ are the set of discrete and continuous parents of $\mathring{X}_i$ respectively. $\alpha(\mathbf{x_{D_i}})$ and $\beta(\mathbf{x_{D_i}})$ are the coefficients of a linear regression model of $\mathring{X}_i$ given its continuous parents. These coefficients depend on the values $\mathbf{x_{D_i}}$ of the discrete parents. The product of all the CPTs and the conditional distributions represent the joint mixed distribution over $\mathcal{X}$. Our ctdBN model shares some similarities with CLGs: it represents mixed probability distributions using a DAG whose nodes represent random variables, the parents of the discrete ones being also necessarily discrete. The main difference between CLGs and ctdBNs lies in the conditional density functions assigned to the continuous random variables. Unlike CLGs, in ctdBNs, they are not limited to normal distributions and any conditional truncated density function can be used. In this sense, ctdBNs are more general than CLGs. The dependences between the continuous variables are also different: in CLGs, those are necessarily linear (the coefficients of the mean vector result from a linear regression) whereas, in ctdBNs, linearity is not required. Of course, nonlinearity can be approximated by piecewise linear functions and can therefore be taken into account in CLGs introducing latent variables and using deterministic relationships, like in [19]. In ctdBNs, this is taken into account directly through the relationships between the discretized random variables. There is no need to introduce latent variables, which complexifies the learning of the model. Finally, in CLGs, the mean vector of the normal distribution assigned to $\mathring{X}_i$ can vary with the values of $\mathring{\mathbf{X}}_\mathbf{C_i}$ whereas, in ctdBNs, this interaction between $\mathring{X}_i$ and $\mathring{\mathbf{X}}_\mathbf{C_i}$ is limited to the interaction between the discretized counterpart of $\mathring{X}_i$ and that of $\mathring{\mathbf{X}}_\mathbf{C_i}$.

The introduction into the ctdBN model of these discretized counterparts makes ctdBNs similar to mixture distributions, which explains their high expressive power. Mixture distributions have been studied in the literature, notably from the learning perspective, see e.g., [20]. However, unlike [20] in which the number of components of the mixture is implicitly assumed to be small, in ctdBNs, this number, which is equal to the product of the domain sizes of the discretized random variables, is potentially very high. Yet, the way ctdBNs are defined, they remain as tractable as fully discrete

BNs. In terms of mixture models, the closest works related to our model are probably MTEs, MOPs and MTBFs. In MTE [10], the distribution over the set of all random variables $\mathcal{X}$ is specified by a density function f such that:

- $\sum_{\mathbf{x_D} \in \Omega_{\mathbf{X_D}}} \int_{\Omega_{\mathring{\mathbf{X}}_\mathbf{C}}} f(\mathbf{x_D}, \mathring{\mathbf{x}}_\mathbf{C})\, d\mathring{\mathbf{x}}_\mathbf{C} = 1$,
- f is an MTE potential over $\mathcal{X}$, i.e.:

Definition 1 (MTE potential).
Let $\mathbf{Y} = \{X_{r_1}, \ldots, X_{r_p}\}$ and $\mathring{\mathbf{Z}} = \{\mathring{X}_{s_1}, \ldots, \mathring{X}_{s_q}\}$ be sets of discrete and continuous variables respectively. A function $\phi : \Omega_{\mathbf{Y} \cup \mathring{\mathbf{Z}}} \mapsto \mathbb{R}_0^+$ is a MTE potential if one of the two following conditions holds:

1. *ϕ can be written as:*
$$\phi(\mathbf{y}, \mathring{\mathbf{z}}) = a_0 + \sum_{i=1}^{m} a_i \exp\left\{ \sum_{j=1}^{p} b_i^{(j)} x_{r_j} + \sum_{k=1}^{q} b_i^{(p+k)} \mathring{x}_{s_k} \right\} \qquad (1)$$
for all $(x_{r_1}, \ldots, x_{r_p}) \in \mathbf{Y}$, $(\mathring{x}_{s_1}, \ldots, \mathring{x}_{s_q}) \in \mathring{\mathbf{Z}}$, where a_i, $i = 0, \ldots, m$ and $b_i^{(j)}$, $i = 1, \ldots, m$, $j = 1, \ldots, p+q$, are real numbers.
2. *There exists a partition $\Omega_1, \ldots, \Omega_k$ of $\Omega_{\mathbf{Y} \cup \mathring{\mathbf{Z}}}$ such that the domain of the continuous variables, $\Omega_{\mathring{\mathbf{Z}}}$, is divided into hypercubes, the domain $\Omega_\mathbf{Y}$ of the discrete variables is divided into arbitrary sets, and such that ϕ is defined as:*
$$\phi(\mathbf{y}, \mathring{\mathbf{z}}) = \phi_i(\mathbf{y}, \mathring{\mathbf{z}}) \quad \text{if } (\mathbf{y}, \mathring{\mathbf{z}}) \in \Omega_i,$$
where each ϕ_i, $i = 1, \ldots, k$, can be written in the form of Equation (1), i.e., it is a MTE potential on Ω_i.

MTEs present attractive features. First, they are expressive in the sense that they can approximate (w.r.t. the Kullback-Leibler distance) any continuous density function [11, 21]. Second, they are easy to learn from datasets [22, 23]. Finally, they satisfy Shafer-Shenoy's propagation axioms [24] and inference can thus be performed using a junction tree-based algorithm [10, 21].

This algorithm can be described as follows. An undirected graph called a Markov network is first created: its nodes correspond to the variables of $\mathcal{X}$ and its edges are such that, for every MTE potential ϕ_i, all the nodes involved in ϕ_i are linked together. This graph is then triangulated by eliminating sequentially all the nodes. A node elimination consists i) in adding edges

to the Markov network in order to create a clique (a complete subgraph) containing the eliminated node and all its neighbors; and ii) in removing the eliminated node and its adjacent edges from the Markov network. The cliques created during this process constitute the nodes of the junction tree. They are linked in order to satisfy a "running intersection" property [4]. Finally, each MTE potential ϕ_i is inserted into a clique containing all its variables.

A collect-distribute message-passing algorithm can then be performed in this junction tree, hence enabling to compute *a posteriori* marginal distributions of all the random variables. As usual, the message passed from one clique $\mathcal{C}_i$ to a neighbor $\mathcal{C}_j$ is the projection onto the variables in $\mathcal{C}_i \cap \mathcal{C}_j$ of the combination of the MTE potentials stored in $\mathcal{C}_i$ with the messages received by $\mathcal{C}_i$ from all its neighbors except $\mathcal{C}_j$. By Equation (1), combinations and projections are Algebraic operations over sums of exponentials. Unfortunately, these operations have a serious shortcoming: when propagating messages from one clique to another, the number of a_i/exp terms in Equation (1) tends to grow exponentially, hence limiting the use of this exact inference mechanism to problems with only a small number of cliques.

To overcome this issue, approximate algorithms based on MCMC [10] or on the Penniless algorithm [12] are provided in the literature.

Mixtures of polynomials (MOP) are similar to MTE except that functions $\phi : \Omega_{\mathbf{Y} \cup \mathring{\mathbf{Z}}} \mapsto \mathbb{R}_0^+$ of Equation (1) are substituted by polynomials over the variables in $\mathbf{Y} \cup \mathring{\mathbf{Z}}$ [15, 16]. MOPs have several advantages over MTEs: their parameters for approximating density functions are easier to determine than those of MTEs. They are also applicable to a larger class of deterministic functions in hybrid BNs. As MTE, the MOP model satisfies Shafer-Shenoy's propagation axioms and inference can thus be performed by message-passing in a junction tree. But, similarly to Equation (1), the number of terms these messages involve tends to grow exponentially with the number of cliques in the junction tree, thereby limiting the use the message-passing algorithm to junction trees with a small number of cliques/random variables.

Finally, mixtures of truncated basis functions (MTBF) generalize both MTEs and MOPs [13]. The definition of an MTBF is the same as Definition 1 except that Equation (1) is substituted by:

$$\phi(\mathbf{y}, \mathring{\mathbf{z}}) = \sum_{i=0}^{m} \prod_{k=1}^{q} a_{i,\mathbf{y}}^{(k)} \psi_i(\mathring{x}_{s_k}), \tag{2}$$

where potentials $\psi_i : \mathbb{R} \mapsto \mathbb{R}$ are basis functions. MTBFs are defined so that

the potentials are closed under combination and projection which, again, ensures that inference can be performed by message-passing in a junction tree. By exploiting cleverly factorizations of terms in Equation (2), inference in MTBFs can be more efficient than in MTEs [14]. But, like all the other aforementioned models, the sizes of the messages tend to grow with the number of cliques in the junction tree.

In the next section, we propose an alternative model that overcomes this issue while still being expressive.

3. Conditional Truncated Densities Bayesian Networks

In this section, we propose a new graphical model called "*conditional truncated densities Bayesian network*". This is a combination of a classical discrete Bayesian network with some conditional truncated densities. Before describing it in details, we therefore need to recall what are Bayesian networks and conditional densities.

Definition 2 (Bayesian network (BN)). *A (discrete) BN* $\mathcal{B}$ *is a pair* $(\mathcal{G}, \boldsymbol{\theta})$ *where* $\mathcal{G} = (\mathbf{X}, \mathbf{A})$ *is a directed acyclic graph (DAG),* $\mathbf{X} = \{X_1, ..., X_n\}$ *represents a set of discrete and/or discretized random variables*[1]*,* $\mathbf{A}$ *is a set of arcs, and* $\boldsymbol{\theta} = \{P(X_i|\mathbf{Pa}(X_i))\}_{i=1}^n$ *is the set of the conditional probability tables/distributions (CPT) of the variables* X_i *in* $\mathcal{G}$ *given their parents* $\mathbf{Pa}(X_i)$ *in* $\mathcal{G}$*. The BN encodes the joint probability over* $\mathbf{X}$ *as* $P(\mathbf{X}) = \prod_{i=1}^n P(X_i|\mathbf{Pa}(X_i))$*.*

Our model is intended to be used in situations where uncertain variables are mixed discrete/continuous. BNs are unable to model such situations because, by their very definition, they assign CPTs to the nodes of their graphical structure, which requires all random variables to be discrete. A first idea to exploit BNs is therefore to discretize the continuous random variables, thereby creating new discrete variables, and to express the uncertainties as a BN over the set of discrete and discretized variables. In some sense, our model is a refinement of this process.

Definition 3 (Discretization). *A discretization of a continuous variable* $\mathring{X}_i$ *is a function* $d_{\mathring{X}_i} : \Omega_{\mathring{X}_i} \to \{0, \ldots, g_i\}$ *defined by an increasing sequence*

[1]By abuse of notation, we use interchangeably $X_i \in \mathbf{X}$ to denote a node in the BN and its corresponding random variable.

of g_i cut points $\{t_1, t_2, ..., t_{g_i}\} \subset \Omega_{\mathring{X}_i}$ such that:

$$d_{\mathring{X}_i}(\mathring{x}_i) = \begin{cases} 0 & \text{if } \mathring{x}_i < t_1, \\ k & \text{if } t_k \leq \mathring{x}_i < t_{k+1}, \quad \text{for all } k \in \{1, \ldots, g_i - 1\} \\ g_i & \text{if } \mathring{x}_i \geq t_{g_i} \end{cases}$$

Thus the discretized variable X_i corresponding to $\mathring{X}_i$ has a finite domain of $\{0, \ldots, g_i\}$. Therefore, after discretizing all the continuous random variables, the uncertainty over all the discrete and discretized random variables $\mathbf{X} = \{X_1, ..., X_n\}$ can be represented by a classical BN in which very efficient exact message-passing inference mechanisms can be used, notably junction tree-based algorithms [2, 25, 3, 4] and weighted model counting methods [5, 6]. In this paper, we will exploit the former.

However, discretizing continuous random variables raises two major issues: i) which discretization function shall be used to minimize the loss of information? and ii) will the loss of information affect significantly the results of inference? A possible answer to the first question consists of exploiting "*conditional truncated densities*" [18]. The answer to the second question of course strongly depends on the discretization performed but, as we shall see, conditional truncated densities can limit the discrepancy between the exact *a posteriori* marginal density functions of the continuous random variables and the approximation they provide.

Definition 4 (Conditional truncated density). *Let $\mathring{X}_i$ be a continuous random variable. Let $d_{\mathring{X}_i}$ be a discretization of $\mathring{X}_i$ with set of cutpoints $\{t_1, t_2, ..., t_{g_i}\}$. Finally, let X_i be a discrete random variable with domain $\Omega_{X_i} = \{0, \ldots, g_i\}$. A conditional truncated density is a function $f(\mathring{X}_i|X_i) : \Omega_{\mathring{X}_i} \times \Omega_{X_i} \mapsto \mathbb{R}_0^+$ satisfying the following properties:*

1. *$f(\mathring{x}_i|x_i) = 0$ for all $x_i \in \Omega_{X_i}$ and $\mathring{x}_i \notin [t_{x_i}, t_{x_i+1})$ with, by abuse of notation $t_0 = \inf \Omega_{\mathring{X}_i}$ and $t_{g_i+1} = \sup \Omega_{\mathring{X}_i}$;*
2. *the following equation holds:*

$$\int_{t_{x_i}}^{t_{x_i+1}} f(\mathring{x}_i|x_i)\, d\mathring{x}_i = 1, \quad \text{for all } x_i \in \Omega_{X_i}. \tag{3}$$

In other words, $f(\mathring{x}_i|x_i)$ represents the truncated density function of random variable $\mathring{X}_i$ over the interval of discretization $[t_{x_i}, t_{x_i+1})$.

Lemma 1. *Let $P(X_i)$ be any probability distribution over the discrete random variable X_i of Definition 4. Then $f(\mathring{X}_i|X_i)P(X_i)$ is a mixed probability distribution over $\Omega_{\mathring{X}_i} \times \Omega_{X_i}$, i.e., it is a non-negative function such that:*

$$\sum_{x_i \in \Omega_{X_i}} \int_{\Omega_{\mathring{X}_i}} f(\mathring{x}_i|x_i)P(x_i)\ d\mathring{x}_i = 1. \tag{4}$$

All the proofs are given in the appendix.

We can now introduce "*Bayesian networks with conditional truncated densities*", which are Bayesian networks defined over discrete and discretized random variables, in which to each discretized variable is assigned its corresponding continuous random variable and its conditional truncated density:

Definition 5 (Conditional truncated densities Bayesian networks). (ctdBN) *Let $\mathbf{X_D} = \{X_1, \ldots, X_d\}$ and $\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{X}_{d+1}, ..., \mathring{X}_n\}$ be sets of discrete and continuous random variables respectively. Let $\mathbf{X_C} = \{X_{d+1}, ..., X_n\}$ be a set of discretized variables resulting from the discretization of the variables in $\mathring{\mathbf{X}}_\mathbf{C}$. Then, a ctdBN is a pair $(\mathcal{G}, \boldsymbol{\theta})$ where:*

- $\mathcal{G} = (\mathbf{X}, \mathbf{A})$ *is a directed acyclic graph,*
- $\mathbf{X} = \mathbf{X_D} \cup \mathbf{X_C} \cup \mathring{\mathbf{X}}_\mathbf{C}$,
- $\mathbf{A}$ *is a set of arcs such that nodes $\mathring{X}_i \in \mathring{\mathbf{X}}_\mathbf{C}$* have no children and exactly one parent equal to X_i. *This condition is the key to guarantee that inference in a ctdBN is as fast as that in a classical BN.*
- *Finally, $\boldsymbol{\theta} = \boldsymbol{\theta}_\mathbf{D} \cup \boldsymbol{\theta}_\mathbf{C}$, where $\boldsymbol{\theta}_\mathbf{D} = \{P(X_i|\mathbf{Pa}(X_i))\}_{i=1}^n$ is the set of the conditional probability tables of the discrete and discretized variables X_i in $\mathcal{G}$ given their parents $\mathbf{Pa}(X_i)$ in $\mathcal{G}$, and $\boldsymbol{\theta}_\mathbf{C} = \{f(\mathring{X}_i|X_i)\}_{i=d+1}^n$ is the set of the conditional truncated densities of the continuous random variables of $\mathring{\mathbf{X}}_\mathbf{C}$.*

Note that $\boldsymbol{\theta}_\mathbf{C}$ needs a very limited amount of memory compared to $\boldsymbol{\theta}_\mathbf{D}$ since truncated densities are univariate (e.g., a truncated normal distribution $f(\mathring{X}_i|X_i)$ is specified by only $2|\Omega_{X_i}|$ parameters). An example of ctdBN is given in Figure 1. The model contains 3 continuous variables, $\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{X}_1, \mathring{X}_3, \mathring{X}_5\}$ represented by blue dotted circles, which are discretized into $\mathbf{X_C} = \{X_1, X_3, X_5\}$. Nodes in pink solid circles $\mathbf{X_C}$ and $\mathbf{X_D}$ form a classical BN. Finally, all the continuous nodes $\mathring{X}_i \in \mathring{\mathbf{X}}_\mathbf{C}$ are children of their discretized counterpart X_i and none has any child. The key idea of ctdBNs

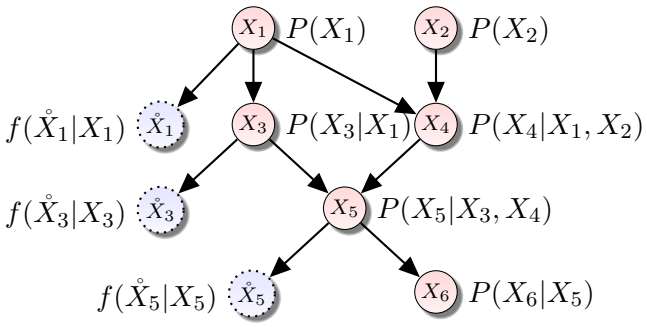

Figure 1: A BN with conditional truncated densities.

is thus to extend BNs by specifying the uncertainties over continuous random variables $\mathring{X}_i$ as 2-level functions: a "rough" probability distribution for discrete variable X_i and a finer-grain conditional density $f(\mathring{X}_i|X_i)$ for $\mathring{X}_i$. This idea can be somewhat related to second order probabilities [26].

Proposition 1. *In a ctdBN defined over* $\mathbf{X} = \mathbf{X_D} \cup \mathbf{X_C} \cup \mathring{\mathbf{X}}_\mathbf{C}$*, where* $\mathbf{X_D} = \{X_1, \ldots, X_d\}$*,* $\mathbf{X_C} = \{X_{d+1}, \ldots, X_n\}$ *and* $\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{X}_{d+1}, \ldots, \mathring{X}_n\}$*, function* $h : \mathbf{X} \mapsto \mathbb{R}_0^+$ *defined as:*

$$h(\mathbf{X}) = \prod_{i=1}^{n} P(X_i|\mathbf{Pa}(X_i)) \prod_{i=d+1}^{n} f(\mathring{X}_i|X_i) \tag{5}$$

is a mixed probability distribution over $\mathbf{X}$*.*

The above proposition therefore shows that ctdBNs encode compactly mixed probability distributions.

3.1. Faithfulness with ctdBNs

It was shown in [11, 21] that MTEs can approximate standard probability density functions (w.r.t. the Kullback-Leibler distance). One may wonder whether ctdBN are also faithful, i.e., whether they can provide good approximations of densities or mixed probability distributions. The propositions provided in this subsection show that the answer to this question is positive and that ctdBNs can actually approximate very general functions.

Proposition 2. *Let* $\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{X}_1, ..., \mathring{X}_n\}$ *be a set of continuous real-valued random variables of respective domains* $\mathring{\Omega}_1, ..., \mathring{\Omega}_n$ *such that none of the* $\mathring{\Omega}_i$ *is a singleton. Let* $\mathring{\mathbf{\Omega}}_\mathbf{C} = \prod_{i=1}^{n} \mathring{\Omega}_i$ *be the domain of* $\mathring{\mathbf{X}}_\mathbf{C}$*. Let* $f : \mathring{\mathbf{\Omega}}_\mathbf{C} \mapsto \mathbb{R}$

be a probability density function. Assume that f is Lipschitz, i.e., there exists a constant $M > 0$ such that, for every pair $(\mathring{x}, \mathring{y})$ of elements of $\mathring{\mathbf{\Omega}}_{\mathbf{C}}$, $|f(\mathring{x}) - f(\mathring{y})| \leq M||\mathring{x} - \mathring{y}||$, where $||\mathring{x} - \mathring{y}||$ represents the L2-norm of vector $(\mathring{x} - \mathring{y})$.

Then, for every strictly positive real number $\epsilon < 1$, there exists a ctdBN $\mathcal{B} = (\mathcal{G}, \boldsymbol{\theta})$ that approximates f up to ϵ, i.e.:

- *the nodes of Graph $\mathcal{G}$ are $\mathbf{X} = \mathbf{X_D} \cup \mathring{\mathbf{X}}_{\mathbf{C}}$, where $\mathbf{X_D} = \{X_1, \ldots, X_n\}$ is a set of the discretized variables corresponding to $\mathring{\mathbf{X}}_{\mathbf{C}}$; in addition, let $\mathbf{\Omega_D} = \prod_{i=1}^{n} \Omega_i$ and $\mathbf{\Omega} = \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_{\mathbf{C}}$ be the domains of $\mathbf{X_D}$ and $\mathbf{X}$ respectively;*

- *$\mathcal{B}$ represents a mixed probability distribution $g : \mathbf{\Omega} \mapsto \mathbb{R}$ such that, for every $\mathring{x} \in \mathring{\mathbf{\Omega}}_{\mathbf{C}}$, $|g(x, \mathring{x}) - f(\mathring{x})| \leq \epsilon$, where x corresponds to the discretized counterpart of $\mathring{x}$.*

The above proposition shows that any Lipschitz multivariate probability density function can be (arbitrarily well) approximated by a ctdBN. Note that, to do so, the conditional truncated densities used by such ctdBN need not be "complex": in the proof of this proposition, only uniform and conditional truncated normal distributions were used. An obvious corollary of this proposition is that standard density functions can be approximated by ctdBNs:

Corollary 1. *Standard distributions like, e.g., univariate and multivariate Normal distributions, Beta distributions $B(\mathring{x}, \alpha, \beta)$, with $\alpha, \beta \geq 2$, Gamma distribution $\Gamma(\mathring{x}, \alpha, \beta)$ with $\alpha > 2$, as well as their combinations by mutually independent random variables, can be approximated up to $\epsilon < 1$ by ctdBNs.*

But ctdBNs represent compactly the uncertainties over both discrete and continuous random variables. So, they may also provide good approximations of mixed probability distributions and the following proposition justifies this intuition:

Proposition 3. *Let $\mathbf{X_D} = \{X_1, \ldots, X_d\}$ be a set of discrete random variables of respective domains $\{\Omega_1, \ldots, \Omega_d\}$ and let $\mathbf{\Omega_D} = \prod_{i=1}^{d} \Omega_i$ be the domain of $\mathbf{X_D}$. Let $\mathring{\mathbf{X}}_{\mathbf{C}} = \{\mathring{X}_{d+1}, ..., \mathring{X}_n\}$ be a set of continuous random variables of respective domains $\mathring{\Omega}_{d+1}, ..., \mathring{\Omega}_n$ such that none of the $\mathring{\Omega}_i$ is a singleton. Let $\mathring{\mathbf{\Omega}}_{\mathbf{C}} = \prod_{i=d+1}^{n} \mathring{\Omega}_i$ be the domain of $\mathring{\mathbf{X}}_{\mathbf{C}}$. Finally, let $\mathbf{X} = \mathbf{X_D} \cup \mathring{\mathbf{X}}_{\mathbf{C}}$ and $\mathbf{\Omega} = \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_{\mathbf{C}}$.*

Let $f : \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_\mathbf{C} \mapsto \mathbb{R}$ *be a mixed probability distribution. Assume that* f *is Lipschitz w.r.t. the continuous variables of* $\mathring{\mathbf{X}}_\mathbf{C}$*, i.e., there exists a constant* $M > 0$ *such that, for every pair* $(\mathring{x}, \mathring{y})$ *of elements of* $\mathring{\mathbf{\Omega}}$ *such that* $x_i = y_i$ *for all* $i \in \{1, \ldots, d\}$*,* $|f(\mathring{x}) - f(\mathring{y})| \leq M||\mathring{x} - \mathring{y}||$*, where* $||\mathring{x} - \mathring{y}||$ *represents the L2-norm of vector* $(\mathring{x} - \mathring{y})$*.*

Then, for every strictly positive real number $\epsilon < 1$*, there exists a ctdBN* $\mathcal{B} = (\mathcal{G}, \boldsymbol{\theta})$ *that approximates* f *up to* ϵ*, i.e.:*

- *the nodes of* $\mathcal{G}$ *are* $\mathbf{X} = \mathbf{X_D} \cup \mathbf{X_C} \cup \mathring{\mathbf{X}}_\mathbf{C}$*, where* $\mathbf{X_C} = \{X_{d+1}, \ldots, X_n\}$ *is a set of discretized variables corresponding to* $\mathring{\mathbf{X}}_\mathbf{C}$*; in addition, let* $\mathbf{\Omega_C} = \prod_{i=d+1}^{n} \Omega_i$ *and* $\mathbf{\Omega} = \mathbf{\Omega_D} \times \mathbf{\Omega_C} \times \mathring{\mathbf{\Omega}}_\mathbf{C}$ *be the domains of* $\mathbf{X_C}$ *and* $\mathbf{X}$ *respectively;*

- $\mathcal{B}$ *represents a mixed probability density function* $g : \mathbf{\Omega} \mapsto \mathbb{R}$ *such that, for every* $(y, \mathring{x}) \in \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_\mathbf{C}$*,* $|g(y, x, \mathring{x}) - f(y, \mathring{x})| \leq \epsilon$*, where* x *corresponds to the discretized counterpart of* $\mathring{x}$*.*

CtdBNs also have some decomposability properties. For instance, the following proposition shows that, if the mixed probability distribution to be approximated is decomposable, then so is also the approximating ctdBN:

Proposition 4. *Let* $\mathbf{X_D} = \{X_1, \ldots, X_d\}$ *and* $\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{X}_{d+1}, ..., \mathring{X}_n\}$ *be sets of discrete and continuous random variables respectively. Let* $f : \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_\mathbf{C} \mapsto \mathbb{R}$ *be a mixed probability distribution. Assume that sets of variables* $\mathbf{X_D}$ *and* $\mathring{\mathbf{X}}_\mathbf{C}$ *can be partitioned into sets* $\{\mathbf{X}_{D_1}, \ldots, \mathbf{X}_{D_k}\}$ *and* $\mathring{\mathbf{X}}_{C_1}, \ldots, \mathring{\mathbf{X}}_{C_k}$ *respectively, and that there exist some non-negative functions* $f_i : \mathbf{\Omega}_{D_i} \times \mathring{\mathbf{\Omega}}_{C_i} \mapsto \mathbb{R}$*,* $i = 1, \ldots, k$*, such that* $f(x, \mathring{x}) = \prod_{i=1}^{k} f_i(x_{D_i}, \mathring{x}_{C_i})$ *for all* $(x, \mathring{x}) \in \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_\mathbf{C}$*. Then if* f *is Lipschitz w.r.t. the continuous variables of* $\mathring{\mathbf{X}}_\mathbf{C}$*, it can be approximated up to* ϵ *by a ctdBN which has the same decomposition, i.e., sets* $(\mathbf{X}_{D_i} \cup \mathbf{X}_{C_i} \cup \mathring{\mathbf{X}}_{C_i})$*,* $i = 1, \ldots, k$*, where* $\mathbf{X}_{C_i}$ *are the discretized counterparts of* $\mathring{\mathbf{X}}_{C_i}$*, form the connected components of the ctdBN's graphical structure.*

All these theoretical results will be confirmed in practice in the experimental section of the paper. Now, we will focus on inferences with ctdBNs, in particular on the efficiency of junction tree-based algorithms.

3.2. Inference in ctdBNs

The terms in Equation (5) satisfy Shafer-Shenoy's propagation axioms [24], so we can rely on a message-passing algorithm in a junction tree to perform inference. The latter is constructed by node eliminations from the

Markov network, as described in the preceding section. It was proved that first eliminating all simplicial nodes, i.e., nodes that, together with their neighbors in the Markov network, constitute a clique (a complete maximal subgraph), cannot prevent obtaining a junction tree that is optimal w.r.t. inference [27]. By the definition of ctdBNs, all the continuous nodes $\mathring{X}_i \in \mathring{\mathbf{X}}_\mathbf{C}$ constitute a clique with their parent X_i (for instance, in Figure 1, $\{X_3, \mathring{X}_3\}$ is a complete maximal subgraph and is thus a clique). As a consequence, the junction tree of a ctdBN is simply the junction tree of its discrete BN part defined over $\mathbf{X_C} \cup \mathbf{X_D}$ to which cliques $\{X_i, \mathring{X}_i\}$, for $\mathring{X}_i \in \mathring{\mathbf{X}}_\mathbf{C}$, have been added (linked to a clique containing X_i in order to satisfy the running intersection property). Figure 2 shows a junction tree related to the ctdBN of Figure 1. All the CPTs $P(X_i|\mathbf{Pa}(X_i))$, $i = 1, \ldots, n$, are inserted into cliques not containing any continuous node of $\mathring{\mathbf{X}}_\mathbf{C}$. Of course, conditional truncated densities are inserted into cliques $\{X_i, \mathring{X}_i\}$, $\mathring{X}_i \in \mathring{\mathbf{X}}_\mathbf{C}$.

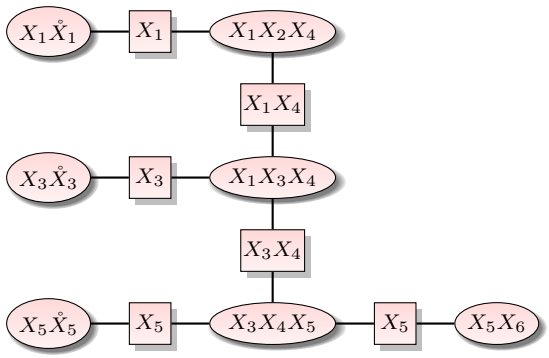

Figure 2: A junction tree for the ctdBN of Figure 1.

The inference process can now be performed by message passing within the junction tree, for instance using a usual collect-distribute algorithm in a Shafer-Shenoy-like architecture [2], sending messages in both directions on all the edges of the junction tree.

There remains to show how to compute the messages sent on the separators in the Shafer-Shenoy collect-distribute algorithm and how to encode and insert evidence into junction tree $\mathcal{T}$. First, let us address the second problem. Of course evidence on discrete random variables X_i are handled in a usual manner by multiplying the joint mixed probability distribution $g(X, \mathring{X})$ represented by the ctdBN with discrete beliefs of the type $P(e_{X_i}|X_i)$. This corresponds to adding probability table $P(e_{X_i}|X_i)$ into a clique of $\mathcal{T}$ containing X_i. For continuous random variables, two cases can occur: first, it may be the case that the available evidence on continuous

random variable $\mathring{X}_i$ can be encoded as an evidence on its discretized random variable, then we do so. For instance, if $\{t_1, \ldots, t_{g_i}\}$ are the cutpoints of the discretization function applied to $\mathring{X}_i$, then evidence "$\mathring{X}_i$ is known to belong $[t_j, t_k]$" can be encoded as a vector $P(e_{X_i}|X_i)$ whose values are 1 for the indices in $\{j, \ldots, k-1\}$, else 0. Second, it may be impossible to enter evidence $e_{\mathring{X}_i}$ on $\mathring{X}_i$ into X_i. In this case, $e_{\mathring{X}_i}$ can be of the type "$\mathring{X}_i$ belongs to some interval $[a, b]$", with $a, b \notin \{t_1, \ldots, t_{g_i}\}$. Such an evidence can be represented by function $f_i(e_{\mathring{X}_i}|\mathring{X}_i) : \Omega_{\mathring{X}_i} \mapsto [0, 1]$ equal to 1 when $\mathring{X}_i \in [a, b]$ and 0 otherwise. As function f_i is defined only over $\mathring{X}_i$, it can be entered into the clique $\mathcal{C}_i = \{X_i, \mathring{X}_i\}$ of junction tree $\mathcal{T}$. More generally, beliefs about $\mathring{X}_i$ can be entered as any $[0, 1]$-valued function $f_i(e_{\mathring{X}_i}|\mathring{X}_i)$ into clique $\mathcal{C}_i$. It is easy to see that the product of the evidence functions $f_i(e_{\mathring{X}_i}|\mathring{X}_i)$ and $P(e_{X_i}|X_i)$ with $g(X, \mathring{X})$ defines, up to a proportional constant equal to the probability of all the evidence, a new mixed probability distribution.

Now, there remains to show how to compute the messages sent from one clique, say $\mathcal{C}_i$, to one of its neighbor $\mathcal{C}_j$. Two cases can occur. First, assume that $\mathcal{C}_i$ contains a continuous random variable $\mathring{X}_i$. Then, by construction of ctdBNs, $\mathcal{C}_i = \{X_i, \mathring{X}_i\}$, with X_i the discretized variable corresponding to $\mathring{X}_i$. By construction, clique $\mathcal{C}_i$ has only one neighbor clique, say $\mathcal{C}_j$, and the separator between $\mathcal{C}_i$ and $\mathcal{C}_j$ is necessarily $S_{ij} = \{X_i\}$. Clique $\mathcal{C}_i$ contains only conditional truncated density $g_i(\mathring{X}_i|X_i)$ and, potentially, some evidence belief $f_i(e_{\mathring{X}_i}|\mathring{X}_i)$. So, in order to remove variable $\mathring{X}_i$ from the equations, it must be marginalized out as:

$$\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}(x_i) = \int_{\Omega_{\mathring{X}_i}} g_i(\mathring{x}_i|x_i) f_i(e_{\mathring{X}_i}|\mathring{x}_i)\, d\mathring{x}_i, \text{ for all } x_i \in \Omega_{X_i}. \tag{6}$$

Assume that $\{t_1, \ldots, t_{g_i}\}$ are the cutpoints of the discretization function applied to $\mathring{X}_i$. Then message $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}$ is a real-valued vector of size $g_i + 1$. So, messages sent from cliques containing continuous random variables to their neighbor are necessarily vectors of finite size. In addition, whether $\mathring{X}_i$ received evidence or not, note that message $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}$ is computed by integrating a *univariate* function, which, in practice, is not time consuming (it can be done either exactly in closed-form formula or approximately using a MCMC algorithm or well-known tables like for normal distributions).

The second case for computing messages concerns situations in which clique $\mathcal{C}_i$ contain only discrete random variables. Then, by construction, the separator $\mathcal{S}_{ij} = \mathcal{C}_i \cap \mathcal{C}_j$ contains only discrete random variables. Message $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}$ can therefore be computed as usual by first multiplying all the

messages sent to $\mathcal{C}_i$ by all $\mathcal{C}_i$'s neighbors except $\mathcal{C}_j$ with the product of the CPTs stored into $\mathcal{C}_i$, and then by marginalizing out all the variables in $\mathcal{C}_i \backslash \mathcal{C}_j$.

Proposition 5. *Let e represent all the evidence entered into junction tree $\mathcal{T}$. Assume that Shafer-Shenoy's message-passing algorithm has been performed, with messages computed as described above.*

Let $\mathcal{C}_k$ be any clique containing only discrete variables and let $\mathcal{C}_{i_1}, \ldots, \mathcal{C}_{i_r}$ be the neighbors of $\mathcal{C}_k$. Then the CPT resulting from the normalization of the product of all the messages $\mathcal{M}_{\mathcal{C}_{i_j} \to \mathcal{C}_k}$, $j = 1, \ldots, r$, with the CPTs stored into $\mathcal{C}_k$ is equal to the joint posterior distribution of the variables of $\mathcal{C}_k$ given evidence e. The posterior of any variable in $\mathcal{C}_k$ can be obtained by marginalizing out the other variables from this CPT.

Let $\mathcal{C}_k$ be any clique containing a continuous random variable, say $\mathring{X}_k$. Let g_k+1 be the domain size of the corresponding discretized random variable X_k. Finally, let $\mathcal{C}_j$ be the neighbor clique of $\mathcal{C}_k$. Then:

$$g_k(\mathring{x}_k|e) \propto \sum_{x_k=0}^{g} \mathcal{M}_{\mathcal{C}_j \to \mathcal{C}_k}(x_k) g_k(\mathring{x}_k|x_k) f_k(e_{\mathring{X}_k}|\mathring{x}_k) \tag{7}$$

is the posterior density of variable $\mathring{X}_k$.

As shown above, ctdBNs allow for the computation of the marginal *a posteriori* distributions of the continuous and discrete random variables. In addition, as shown in the next proposition, the algorithm proposed in this paper is very efficient for performing these computations. Notably, when the integrals of Equation (6) can be computed in O(1), the complexity of inference in ctdBN is exactly the same as that in classical discrete BNs. As a consequence, when tables for these integrals are available, like, e.g., when the ctdBN's conditional truncated densities are truncated normal distributions, inference in ctdBNs is as fast as that in discrete BNs.

Proposition 6. *Let w be the treewidth of $\mathcal{T}$ and let k denote the maximum domain size of the discrete and discretized random variables. Finally, let n be the number of random variables in the ctdBN and let $\overline{I}$ be the average complexity of computing one integral of Equation (6) (i.e., an integral for a given value of x_i) and $\overline{J}$ the average complexity of computing the product in Equation (7). Then the complexity of computing all the marginal posterior distributions of all the random variables is in $O(nk(k^w + \overline{I} + \overline{J}))$.*

Overall, inference in ctdBNs is fast because i) by construction, most of the inference's complexity lies in computations performed on discrete variables; and ii) whenever computations concern densities, either they correspond to compute a mixture of univariate conditional truncated densities (like in Equation (7)) or to compute the integral of a *univariate* function (like in Equation (6)).

4. Experiments

In this section, we provide two sets of experiments. The first one is intended to show the gain brought by ctdBNs over classical BNs. For this purpose, we illustrate the discrepancies between both models on classification problems derived from UCI datasets [28]. The second set of experiments is devoted to the comparison between ctdBNs and MTBFs in order to highlight the inference scalability of ctdBNs compared to that of MTBFs.

4.1. Comparisons with discrete BNs

In order to compare BNs and ctdBNs on real-world problems, we base our experiments on the real-world datasets of the UCI repository [28] reported in Table 1. In these datasets, all records with missing values are discarded. In each resulting dataset, there exists a discrete random variable, call it X_0, representing a classification variable. The other random variables can be either discrete (variables $\mathbf{X_D} = \{X_1, \ldots, X_d\}$) or continuous ($\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{X}_{d+1}, \ldots, \mathring{X}_n\}$). Our classification problem consists of estimating the most probable value of X_0 given some observation on the values of variables in $\mathbf{X_D} \cup \mathring{\mathbf{X}}_\mathbf{C}$.

dataset	#attributes	#classes	#instances	#continuous attr.
australian	14	2	690	6
cleve	14	2	296	13
crx	16	2	653	6
glass2	10	2	163	9
iris	5	3	150	4
pima	9	2	768	8
shuttle small	10	7	3866	9
vehicle	19	4	846	18

Table 1: UCI datasets used for BN/ctdBN comparisons in classification tasks.

To address such a problem with Bayesian networks, we must first discretize all the continuous random variables. To do so, we exploit Friedman's discretization algorithm [29]. After performing these discretizations,

variables in $\mathring{\mathbf{X}}_{\mathbf{C}} = \{\mathring{X}_{d+1}, \ldots, \mathring{X}_n\}$ are mapped into discretized variables in $\mathbf{X_C} = \{X_{d+1}, \ldots, X_n\}$ and the mixed discrete/continuous dataset $\mathring{\mathcal{D}}$ of the UCI dataset is mapped into a fully discrete dataset $\mathcal{D}$. A BN $\mathcal{B}$ over $(X_0, X_1, \ldots, X_n)$ is then learnt from $\mathcal{D}$ using a hill climbing algorithm with an MDL score. To do so, we use the aGrUM library (`http://www.agrum.org`). This BN is then exploited for a classification task as follows: given some observation $e_{\mathring{X}_i}$ (resp. e_{X_i}) on each continuous random variable $\mathring{X}_i$ (resp. discrete variable X_i), we enter belief $P(e_{\mathring{X}_i}|X_i)$ (resp. $P(e_{X_i}|X_i)$) into $\mathcal{B}$, so that the latter represents:

$$P(X_0, \ldots, X_n, e_{X_1}, \ldots, e_{X_d}, e_{\mathring{X}_{d+1}}, \ldots, e_{\mathring{X}_n}) = \\ P(X_0|\mathbf{Pa}(X_0)) \prod_{i=1}^{n} P(X_i|\mathbf{Pa}(X_i)) \prod_{i=1}^{d} P(e_{X_i}|X_i) \prod_{i=d+1}^{n} P(e_{\mathring{X}_i}|X_i).$$

From this distribution, by means of a Shafer-Shenoy (exact) inference, we compute the posterior distribution $P(X_0|e_{X_1}, \ldots, e_{X_d}, e_{\mathring{X}_{d+1}}, \ldots, e_{\mathring{X}_n})$ so that the most probable value for class variable X_0 is determined as:

$$x_0^* = \mathrm{Argmax}_{X_0} P(X_0|e_{X_1}, \ldots, e_{X_d}, e_{\mathring{X}_{d+1}}, \ldots, e_{\mathring{X}_n}).$$

To highlight the gain brought by ctdBNs over simple BNs, for each dataset, we construct our ctdBN as follows: we start from BN $\mathcal{B}$ computed in the preceding paragraphs and we add to it its respective conditional truncated densities $g_i(\mathring{X}_i|X_i)$, $i = d+1, \ldots, n$, defined as follows: let $\mathring{\Omega}_i^{obs} = \{\mathring{x}_{i,1}, \mathring{x}_{i,2}, \ldots, \mathring{x}_{i,N'}\}$ be the set of distinct observed values of $\mathring{X}_i$ in the dataset, sorted by increasing order. The midpoints of $\mathring{\Omega}_i^{obs}$ are defined as:

$$m_{i,j} = \begin{cases} \mathring{x}_{i,1} - \frac{\mathring{x}_{i,2} - \mathring{x}_{i,1}}{2} & \text{if } j = 0, \\ \frac{\mathring{x}_{i,j} + \mathring{x}_{i,j+1}}{2} & \text{if } 1 \leq j < N', \\ \mathring{x}_{i,N'} + \frac{\mathring{x}_{i,N'} - \mathring{x}_{i,N'-1}}{2} & \text{if } j = N'. \end{cases}$$

Let $h_i : \Omega_{\mathring{X}_i} \mapsto \mathbb{R}$ be the histogram of $\mathring{X}_i$ whose bins correspond to intervals $[m_{i,j}, m_{i,j+1})$. Assume that $\mathring{X}_i$ has been discretized into X_i using cutpoints $\{t_i^1, \ldots, t_i^{g_i}\}$. Then we define conditional truncated densities $g_i(\mathring{X}_i|X_i = j)$, $j = 0, \ldots, g_i$, as the normalized histogram of h_i over $[t_i^j, t_i^{j+1})$, i.e.,

$$g_i(\mathring{x}_i|X_i = j) = \begin{cases} \dfrac{h_i(\mathring{x}_i)}{\int_{t_i^j}^{t_i^{j+1}} h_i(\mathring{x}) d\mathring{x}} & \text{if } \mathring{x}_i \in [t_i^j, t_i^{j+1}), \\ 0 & \text{otherwise.} \end{cases}$$

The ctdBN therefore represents the following mixed probability distribution:

$$g(X_0,\ldots,X_n,\mathring{X}_{d+1},\ldots,\mathring{X}_n) = \\ P(X_0|\mathbf{Pa}(X_0))\prod_{i=1}^{n}P(X_i|\mathbf{Pa}(X_i))\prod_{i=d+1}^{n}g_i(\mathring{X}_i|X_i).$$

The same evidence e_{X_i} and $e_{\mathring{X}_i}$ as those of the BN are entered into the ctdBN. However, the latter are included into the ctdBN as beliefs $f_i(e_{\mathring{X}_i}|\mathring{X}_i)$ as ctdBNs can cope with more precise evidence than mere beliefs $P(e_{\mathring{X}_i}|X_i)$ about discretized random variables X_i. Therefore, after entering evidence, the ctdBN represents:

$$g(X_0,\ldots,X_n,\mathring{X}_{d+1},\ldots,\mathring{X}_n,e_{X_1},\ldots,e_{X_d},e_{\mathring{X}_{d+1}},\ldots,e_{\mathring{X}_n}) = \\ P(X_0|\mathbf{Pa}(X_0))\prod_{i=1}^{n}P(X_i|\mathbf{Pa}(X_i))\prod_{i=1}^{d}P(e_{X_i}|X_i)\prod_{i=d+1}^{n}g_i(\mathring{X}_i|X_i)f_i(e_{\mathring{X}_i}|\mathring{X}_i).$$

From this distribution, using the algorithm provided in Section 3, we compute $g(X_0|e_{X_1},\ldots,e_{X_d},e_{\mathring{X}_{d+1}},\ldots,e_{\mathring{X}_n})$ and the most probable value for class variable X_0 is $x_0^* = \text{Argmax}_{X_0}g(X_0|e_{X_1},\ldots,e_{X_d},e_{\mathring{X}_{d+1}},\ldots,e_{\mathring{X}_n})$.

Finally, to perform our experiments, each dataset of Table 1 is randomly shuffled 100 times. Each resulting dataset is splitted into a learning set (70%) and a test set (30%). So, overall, for each UCI dataset, 100 different learning sets and their respective 100 test sets are created. From each learning set, we learn a BN and a ctdBN and, then, for each record of the corresponding test set, we estimate the most probable values of class variable X_0 given observations on $X_1,\ldots,X_d,\mathring{X}_{d+1},\ldots,\mathring{X}_n$ according to these two models. These estimations are then compared with the true values of X_0 observed in the test set and the accuracy of the model (BN,ctdBN) is defined as the proportion of correct estimations performed. The latter depend on the kind of observations available, so we shall now describe those used in the experiments.

First, all observations over discrete variables are supposed to be precise. Now, assume that observation $e_{\mathring{X}_i}$ over continuous variable $\mathring{X}_i$ is also precise, i.e., it is equal to the observed value of $\mathring{X}_i$ in the record. Then, for the ctdBN, $f_i(e_{\mathring{X}_i}|\mathring{X}_i)$ is a Dirac function, which means that message $\mathcal{M}_{\mathcal{C}_i\to\mathcal{C}_j}$, as defined in Eq. (6), is a zero filled vector, bringing no information. In addition, the BN is unable to handle such precise information. What can be handled by the BN is the (less precise) observation that "the discretized value of the observed value of $\mathring{X}_i$ is equal to j". Such information is exactly equivalent

to "$\mathring{X}_i$ belongs to interval $[t_i^j, t_i^{j+1})$", where $[t_i^j, t_i^{j+1})$ is the discretization interval containing the observed value of $\mathring{X}_i$. In this case, $P(e_{\mathring{X}_i}|X_i)$, is a vector filled with zeros except for a 1 in the cell corresponding to the discretized value of $\mathring{X}_i$. If we enter the same information into the ctdBN, Message $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}$ is exactly proportional to $P(e_{\mathring{X}_i}|X_i)$ and, therefore, the BN and the ctdBN provide the same estimation for X_0.

The advantage of ctdBNs over standard BNs becomes visible when observations are imprecise. So, in our experiments, all the continuous variables $\mathring{X}_i$ are imprecisely observed and the belief $f_i(e_{\mathring{X}_i}|\mathring{X}_i)$ is always expressed as a normal distribution centered on the observed value of $\mathring{X}_i$. For the BN, vector $P(e_{\mathring{X}_i}|X_i)$ can then simply be computed as:

$$P(e_{\mathring{X}_i}|X_i = j) = \int_{t_i^j}^{t_i^{j+1}} f_i(e_{\mathring{X}_i}|\mathring{x}_i)\, d\mathring{x}_i \quad \text{for all } j. \tag{8}$$

When the standard deviations of the normal distributions are sufficiently small, in the BN, $P(e_{\mathring{X}_i}|X_i)$ is approximately equal to a vector filled with zeros except for one cell equal to 1 and, in the ctdBN, Message $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}$ of Eq. (6) is approximately proportional to $P(e_{\mathring{X}_i}|X_i)$. So, both models are equivalent. However, when standard deviations are higher, i.e., when observations are less precise, $P(e_{\mathring{X}_i}|X_i)$ can contain several non-zero cells and, from Eq. (6), it is clear that Message $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}$ can differ from $P(e_{\mathring{X}_i}|X_i)$ and bring more refined information than $P(e_{\mathring{X}_i}|X_i)$. In our experiments, all the standard deviations of the continuous variables are kept sufficiently small

Dataset	standard deviations and variables' domain sizes
australian	$\mathring{X}_7 : 19$ (28.5), $\mathring{X}_{10} : 27$ (68), $\mathring{X}_{13} : 98$ (2000), $\mathring{X}_{14} : 83.5$ (100000)
cleve	$\mathring{X}_8 : 21$ (133)
crx	$\mathring{X}_{11} : 37$ (68), $\mathring{X}_{15} : 235$ (100000)
glass	$\mathring{X}_3 : 0.55$ (4.5), $\mathring{X}_4 : 0.1$ (3.8), $\mathring{X}_5 : 0.6$ (5.6), $\mathring{X}_6 : 0.55$ (6.2)
	$\mathring{X}_8 : 0.35$ (3.15)
iris	$\mathring{X}_3 : 0.03$ (6), $\mathring{X}_4 : 0.085$ (2)
pima	$\mathring{X}_1 : 6.6$ (18), $\mathring{X}_2 : 15.4$ (200), $\mathring{X}_4 : 0.85$ (100), $\mathring{X}_6 : 1.3$ (67)
shuttle_small	$\mathring{X}_1 : 3.5$ (74)
vehicle	$\mathring{X}_6 : 3$ (54), $\mathring{X}_{11} : 34$ (191)

Table 2: The standard deviations for the beliefs on the observations of the $\mathring{X}_i$'s. The domain sizes of the $\mathring{X}_i$'s are given inside parentheses. In each dataset, the first variable/column is called X_1 or $\mathring{X}_1$, the second one X_2 or $\mathring{X}_2$, etc. Class variable X_0 is always located in the last column of the dataset.

that $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j} \propto P(e_{\mathring{X}_i}|X_i)$, except for the few variables mentioned in Table 2 in which the standard deviations displayed introduce discrepancies between $\mathcal{M}_{\mathcal{C}_i \to \mathcal{C}_j}$ and $P(e_{\mathring{X}_i}|X_i)$. Note that these standard deviations are often much smaller than the range of the random variable.

With the observations as described above, the average accuracies for the different UCI datasets over the 100 corresponding test sets, as well as their standard deviations, are reported in Table 3. From this table, it is clear that ctdBNs outperform BNs for classification tasks. The way we constructed the ctdBNs from the BNs, this improvement is necessarily due to the conditional truncated densities contained in the ctdBNs.

Dataset	% BN Acc.	% ctdBN Acc.	Gain Acc.	p-value
australian	84.36 ± 3.08	$\mathbf{85.72 \pm 2.12}$	1.34 ± 2.45	0.2912
cleve	82.89 ± 3.67	$\mathbf{83.22 \pm 3.50}$	0.34 ± 0.96	0.3632
crx	85.36 ± 2.48	$\mathbf{86.38 \pm 2.08}$	1.02 ± 2.12	0.3156
glass	89.94 ± 3.56	$\mathbf{91.88 \pm 2.98}$	1.94 ± 2.61	0.2236
iris	94.73 ± 2.62	$\mathbf{95.49 \pm 2.82}$	0.76 ± 1.45	0.3015
pima	73.64 ± 2.67	$\mathbf{74.37 \pm 2.69}$	0.74 ± 1.18	0.2676
shuttle small	84.92 ± 5.93	$\mathbf{92.66 \pm 3.41}$	7.74 ± 4.21	0.0336
vehicle	35.63 ± 6.02	$\mathbf{52.21 \pm 7.58}$	16.57 ± 7.97	0.0000

Table 3: Comparisons between BNs and ctdBNs for classification tasks. The first two columns present the average accuracies and the accuracies' standard deviations over the 100 datasets constructed from each UCI dataset. The third column displays the gain in accuracy of using ctdBNs instead of BNs (the subtraction of the first column from the second one). Assuming that accuracies are distributed w.r.t. normal distributions of parameters the average BN accuracy and the variance of the BN accuracy, the last column shows the p-value obtained by the ctdBN accuracy.

4.2. Comparisons with MTBFs

In this subsection, we highlight the faithfulness of ctdBNs by comparing them with MTBFs, more precisely with MOPs [15] and MTEs [10]. We show that ctdBNs can approximate effectively these MTBFs while being more efficient in terms of inference. For this purpose, we generate MTBFs $\Phi(\mathring{X}_1, \ldots, \mathring{X}_n)$ as products of bivariate MTBF potentials:

$$\Phi(\mathring{X}_1, \ldots, \mathring{X}_n) = \Phi_1(\mathring{X}_1, \mathring{X}_2) \times \Phi_2(\mathring{X}_2, \mathring{X}_3) \times \cdots \times \Phi_{n-1}(\mathring{X}_{n-1}, \mathring{X}_n).$$

This decomposition has been chosen in order to make inference in MTBFs as fast as possible. Indeed, the corresponding junction tree contains only cliques of size two, having at most two neighbors, which limits the combinatorics of the algebraic operations performed during inference. All these

MTBF potentials are defined over some continuous variables $\mathring{X}_i$ whose domain is $\Omega_{\mathring{X}_i} = [0, 10]$. To make MTEs as efficient as possible for inference, we express MTE potentials $\Phi_i^{MTE}(\mathring{X}_i, \mathring{X}_{i+1})$ by only one exponential term:

$$\Phi_i^{MTE}(\mathring{X}_i, \mathring{X}_{i+1}) = a_{i,0} + a_{i,1} \exp\left(b_{i,0}\mathring{X}_i + b_{i,1}\mathring{X}_{i+1}\right),$$

with $a_{i,0}, a_{i,1} \in [0, 1]$ and $b_{i,0}, b_{i,1} \in [0.5, 1]$ chosen randomly.

In order to make the shape of MOP potentials not easily captured by affine distributions[2] (which we use in our ctdBNs) while guaranteeing that inferences in MOPs are as fast as possible, we define MOP potentials as polynomials of degree 6, more precisely as polynomials of degree 3 in each of their variables, i.e.,:

$$\Phi_i^{MOP}(\mathring{X}_i, \mathring{X}_{i+1}) = \sum_{j=0}^{3}\sum_{k=0}^{3} c_{i,j,k}\mathring{X}_i^j\mathring{X}_{i+1}^k,$$

with $c_{i,j,k}$ chosen randomly in interval $[0, 1]$.

Our goal is to approximate these MTBFs by ctdBNs $\mathcal{B}$ encoding a mixed probability distribution $g(X_1, \ldots, X_n, \mathring{X}_1, \ldots, \mathring{X}_n)$. To construct $\mathcal{B}$, we first discretize every continuous variable $\mathring{X}_i$ in 50 equally-sized intervals, hence resulting in discretized random variables X_i. Then, we construct a discrete BN $\mathcal{B}_d$ over $X_1, \ldots, X_n$ whose independence structure corresponds to that of the MTBF, i.e., to the structure shown in Figure 3. Finally, ctdBN $\mathcal{B}$ is defined as $\mathcal{B}_d$ to which are added conditional truncated densities $g_i(\mathring{X}_i|X_i)$. Therefore, the ctdBN encodes the following mixed distribution:

$$g(X_1, \ldots, X_n, \mathring{X}_1, \ldots, \mathring{X}_n) = P_{\mathcal{B}_d}(X_1)\prod_{i=2}^{n} P_{\mathcal{B}_d}(X_i|X_{i-1})\prod_{i=1}^{n} g_i(\mathring{X}_i|X_i).$$

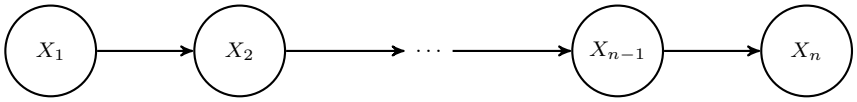

Figure 3: The BN structure used for approximating MTBFs.

As a result, the junction tree used for inferences with ctdBN $\mathcal{B}$ always contains only cliques of size two, exactly like that of the MTBF. Since inference complexity with junction trees is exponential in the treewidth, imposing

[2]The conditional truncated densities we use in our ctdBNs are in fact Beta rectangular distributions with parameters $\alpha = 2$ and $\beta = 1$.

the structure of Figure 3 allows to perform a fair comparison of how well ctdBNs' and MTBFs' inferences scale with increasing numbers of random variables.

As the structure of $\mathcal{B}$ is imposed, only $\mathcal{B}$'s parameters (CPTs $P_{\mathcal{B}_d}(X_i|X_{i-1})$ and functions $g_i(\mathring{X}_i|X_i)$) need be learnt. This learning is performed iteratively, i.e., in a first step, functions $P_{\mathcal{B}_d}(X_1)$, $P_{\mathcal{B}_d}(X_2|X_1)$, $g_i(\mathring{X}_1|X_1)$ and $g_i(\mathring{X}_2|X_2)$ are determined. Then, assuming that ctdBN $\mathcal{B}$ has been constructed for variables $X_1, \mathring{X}_1$ up to $X_{i-1}, \mathring{X}_{i-1}$, we learn $P_{\mathcal{B}_d}(X_i|X_{i-1})$ and $g_i(\mathring{X}_i|X_i)$. More precisely, to learn the parameters related to $X_i, \mathring{X}_i$, we first perform an inference in the MTBF in order to compute:

$$\Phi(\mathring{X}_{i-1}, \mathring{X}_i) = \sum_{\{\mathring{X}_1,\ldots,\mathring{X}_n\}\setminus\{\mathring{X}_{i-1},\mathring{X}_i\}} \Phi(\mathring{X}_1, \ldots, \mathring{X}_n).$$

We also compute $\Phi(\mathring{X}_i) = \sum_{\mathring{X}_{i-1}} \Phi(\mathring{X}_{i-1}, \mathring{X}_i)$. Then we sample 500 000 times the potentials $\Phi(\mathring{X}_{i-1}, \mathring{X}_i)$ and $\Phi(\mathring{X}_i)$ using the Metropolis-Hastings algorithm [30] and the Inverse Transform Sampling's method, respectively. Discretizing these samples using the discretizations of $\mathring{X}_{i-1}$ and $\mathring{X}_i$ defined above results in new discrete samples from which we determine by maximum likelihood some probability distributions $P(X_{i-1}|X_i)$ and $P(X_i)$ respectively. Note that, due to the finite size of the samples, the estimated distributions $P(X_{i-1}|X_i)$ and $P(X_i)$ are not necessarily equal to $P_{\mathcal{B}_d}(X_{i-1}|X_i)$ and $P_{\mathcal{B}_d}(X_i)$. From these two distributions, we compute $P(X_{i-1}, X_i) = P(X_{i-1}|X_i) \times P(X_i)$. The goal of constructing joint distribution $P(X_{i-1}, X_i)$ in this two-step process rather than directly from the sample of $\Phi(\mathring{X}_{i-1}, \mathring{X}_i)$ is to ensure that, in this joint, the marginal distribution $P(X_i)$ is as close as possible to distribution $\Phi(\mathring{X}_i)$, which may not be the case when sampling with Metropolis-Hastings uniquely on pairs $(\mathring{X}_{i-1}, \mathring{X}_i)$, due to the finite size of the samples. Finally, we define $P_{\mathcal{B}_d}(X_1) = P(X_1)$ and, for every $i > 1$, $P_{\mathcal{B}_d}(X_i|X_{i-1}) = P(X_{i-1}, X_i)/P_{\mathcal{B}_d}(X_{i-1})$, where $P_{\mathcal{B}_d}(X_{i-1})$ is computed by inference in the ctdBN constructed so far.

In our experiments, for every j, conditional density $g_i(\mathring{X}_i|X_i = j)$ is an affine function on discretization interval $[t_i^j, t_i^{j+1})$ and is equal to 0 everywhere else. Therefore, it is of the form $g_i(\mathring{x}_i|X_i = j) = \alpha_{i,j}\mathring{x}_i + \beta_{i,j}$ on this interval and, since this is a probability density function, we have that:

$$\int_{t_i^j}^{t_i^{j+1}} g_i(\mathring{x}_i|X_i = j)d\mathring{x}_i = 1.$$

As a consequence, the following equation holds for all $\mathring{x}_i \in \Omega_{\mathring{X}_i}$:

$$g_i(\mathring{x}_i|X_i = j) = \alpha_{i,j}\left(\mathring{x}_i - \frac{t_i^j + t_i^{j+1}}{2}\right) + \frac{1}{t_i^{j+1} - t_i^j}.$$

Using the projection over $\mathring{X}_i$ of the 500 000-record sample of $\Phi(\mathring{X}_{i-1}, \mathring{X}_i)$ previously used for estimating $P_{\mathcal{B}_d}(X_i|X_{i-1})$, we can now estimate $\alpha_{i,j}$ by maximum likelihood under the constraint that $g_i(\cdot)$ is non-negative on $[t_i^j, t_i^{j+1})$. As there is no closed-form formula for the optimal value of $\alpha_{i,j}$, we solve this constrained optimization problem by the Newton-Raphson method.

The experiments are conducted as follows: we generate MTBFs (both MTEs and MOPs) with $n = 4$, 8, 16, 32, 64, 128 and 256 variables. For each number of variables, 25 MTEs and 25 MOPs are generated. Exact inferences are performed in all these models using Variable Elimination [3] in order to determine the distribution $\Phi(\mathring{X}_n)$ of the last random variable. For each MOP and each MTE, we also learn a ctdBN as described above and execute the inference algorithm described in Section 3 to determine the distribution $g(\mathring{X}_n)$ of $\mathring{X}_n$.

The inference times are reported in Tables 4 and 5. They highlight the scalability of ctdBNs. The ratios of the average inference time by the number of variables, i.e., the last two columns of the tables, are displayed in Figures 4 and 5. It is clear that the MTBFs' inference times increase exponentially with the number of variables, even though the largest clique always remains of size 2. This results from the multiplications of the algebraic functions performed during the inferences that tend to produce new functions with an ever increasing number of parameters. Even marginalizations cannot restrain this increase. Unlike in MTBFs, in ctdBNs, the number of operations performed on each clique remains always the same during inference. This explains the linear increase in computation times when the number of vari-

n	$T_{mte}(ms)$	$T_{ctdBN}(ms)$	T_{mte}/n	T_{ctdBN}/n
4	0.30 ± 0.03	1.59 ± 0.16	0.07	0.40
8	0.57 ± 0.08	3.53 ± 0.31	0.07	0.44
16	1.67 ± 1.10	7.14 ± 0.41	0.10	0.45
32	5.79 ± 0.87	13.99 ± 1.35	0.18	0.44
64	23.68 ± 2.01	29.22 ± 1.92	0.37	0.46
128	122.37 ± 10.99	50.87 ± 6.88	0.96	0.40
256	708.11 ± 37.12	96.02 ± 13.79	2.77	0.38

Table 4: Average inference times (plus standard deviations) for MTEs and ctdBNs. These averages are computed over the 25 different networks defined for each number n of variables.

n	$T_{mop}(ms)$	$T_{ctdBN}(ms)$	T_{mop}/n	T_{ctdBN}/n
4	0.79 ± 0.16	1.59 ± 0.19	0.20	0.40
8	1.76 ± 0.29	3.40 ± 0.26	0.22	0.42
16	5.04 ± 0.81	6.94 ± 0.78	0.32	0.43
32	16.41 ± 1.42	14.40 ± 0.89	0.51	0.45
64	50.96 ± 6.89	29.21 ± 1.55	0.80	0.46
128	217.07 ± 22.04	60.19 ± 4.87	1.70	0.47
256	1063.48 ± 51.82	111.34 ± 16.88	4.15	0.43

Table 5: Average inference times (plus standard deviations) for MOPs and ctdBNs. These averages are computed over the 25 different networks defined for each number n of variables.

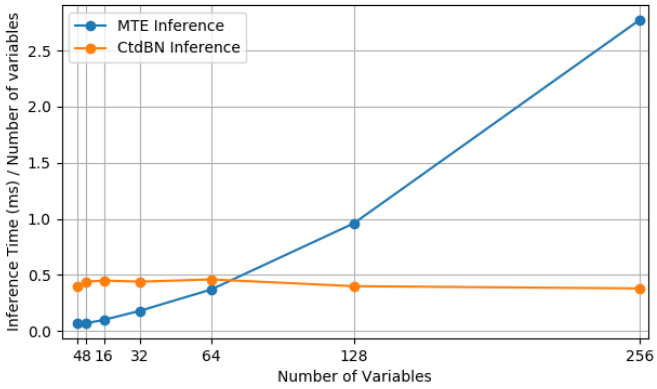

Figure 4: The ratio of inference times in MTEs and ctdBNs by the number of variables.

ables increase. This also corroborates the inference complexity provided in Proposition 4.

Of course, better inference times are attractive only if the estimations by ctdBNs approximate pretty well those of MTBFs. As a first hint that this is the case, Table 6 displays the average Jensen-Shannon Divergence (JSD) between the distributions $\Phi(\mathring{X}_n)$ and $g(\mathring{X}_n)$ computed previously. The low JSD values show that ctdBNs approximate effectively MTBFs. To precise these results, we add some small noise to distributions $\Phi(\mathring{X}_n)$ computed previously, hence resulting in new distributions $\Psi(\mathring{X}_n)$. Comparing the JSDs between $\Phi(\mathring{X}_n)$ and $\Psi(\mathring{X}_n)$ on one hand and between $\Phi(\mathring{X}_n)$ and $g(\mathring{X}_n)$ on the other, we show that ctdBNs provide better approximations than the slightly perturbed distributions, hence highlighting the ctdBN's faithfulness, while enabling much faster inferences than MTBFs, as shown above.

For the MTEs, the marginal distribution inferred for $\mathring{X}_n$ is of the form

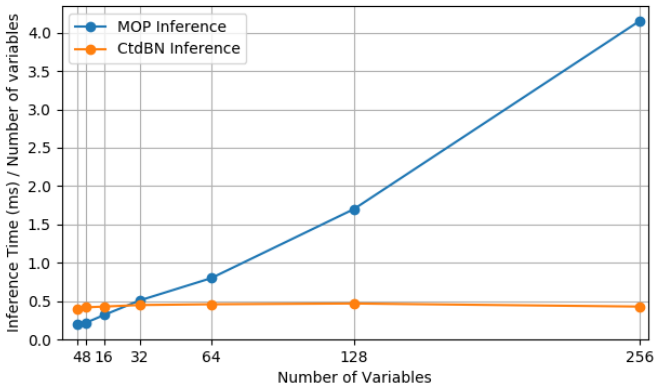

Figure 5: The ratio of inference times in MOPs and ctdBNs by the number of variables.

$\Phi^{MTE}(\mathring{X}_n) = a_0 + b_0 \exp(b_1 \mathring{X}_n)$. We perturb its parameters by ϵ, for different values of ϵ, as follows:

$$\Psi_\epsilon^{MTE}(\mathring{X}_n) \propto (1+\epsilon)a_0 + (1-\epsilon)b_0 \exp\big((1+\epsilon)b_1 \mathring{X}_n\big).$$

Note that $\Psi_\epsilon^{MTE}(\mathring{X}_n)$ is not strictly equal to the right hand side of the above equation, but only proportional to it, so that the integral of Ψ_ϵ^{MTE} over $\Omega_{\mathring{X}_n}$ is equal to one, hence ensuring that Ψ_ϵ^{MTE} is a probability density function. For MOPs, the marginal distribution of $\mathring{X}_n$ is of the form $\Phi^{MOP}(\mathring{X}_n) = c_0 + c_1 \mathring{X}_n + c_2 \mathring{X}_n^2 + c_3 \mathring{X}_n^3$. We perturb it up to ϵ as follows:

$$\Psi_\epsilon^{MOP}(\mathring{X}_n) \propto (1+\epsilon)c_0 + (1-\epsilon)c_1 \mathring{X}_n^{1+\epsilon} + (1+\epsilon)c_2 \mathring{X}_n^{2(1-\epsilon)} + (1-\epsilon)c_3 \mathring{X}_n^{3(1+\epsilon)}.$$

As for Ψ_ϵ^{MTE}, function Ψ_ϵ^{MOP} is normalized so that its integral over $\Omega_{\mathring{X}_n}$ is equal to one. The average JSDs between distributions $\Phi(\mathring{X}_n)$ and $\Psi_\epsilon(\mathring{X}_n)$ for $\epsilon = 0.1$, 0.05 and 0.01 are provided in Tables 7 and 8 for MTEs and MOPs respectively. As can be observed, for $\epsilon = 0.05$, both for MTEs and MOPs, and whatever the number of random variables, the JSDs between the true distribution $\Phi(\mathring{X}_n)$ and the distribution $g(\mathring{X}_n)$ inferred by ctdBN are smaller than those between $\Phi(\mathring{X}_n)$ and ϵ-perturbed distributions $\Psi_\epsilon(\mathring{X}_n)$. This supports the fact that ctdBNs approximate very well MTBFs. Indeed, in real-world applications, the parameters of MTEs and MOPs are learnt from datasets and perturbed probability density functions $\Psi_\epsilon(\mathring{X}_n)$ can be seen as the result of the imprecision on the values of these parameters due to this learning.

We can therefore conclude that ctdBNs outperform MTBFs in terms of scalability of inference. As shown above, the cost of this speed increase is

n	$JSD[\Phi^{MTE}(\mathring{X}_n);\ g(\mathring{X}_n)]$	$JSD[\Phi^{MOP}(\mathring{X}_n);\ g(\mathring{X}_n)]$
4	$2.47 \times 10^{-4} \pm 7.19 \times 10^{-5}$	$8.07 \times 10^{-5} \pm 7.40 \times 10^{-6}$
8	$2.33 \times 10^{-4} \pm 8.28 \times 10^{-5}$	$8.33 \times 10^{-5} \pm 4.24 \times 10^{-6}$
16	$2.32 \times 10^{-4} \pm 7.56 \times 10^{-5}$	$8.24 \times 10^{-5} \pm 6.13 \times 10^{-6}$
32	$2.56 \times 10^{-4} \pm 8.15 \times 10^{-5}$	$8.30 \times 10^{-5} \pm 5.48 \times 10^{-6}$
64	$2.58 \times 10^{-4} \pm 9.05 \times 10^{-5}$	$8.10 \times 10^{-5} \pm 7.11 \times 10^{-6}$
128	$2.07 \times 10^{-4} \pm 6.72 \times 10^{-5}$	$7.85 \times 10^{-5} \pm 7.83 \times 10^{-6}$
256	$2.39 \times 10^{-4} \pm 7.87 \times 10^{-5}$	$8.18 \times 10^{-5} \pm 5.11 \times 10^{-6}$

Table 6: Average Jensen-Shannon Divergences between $\Phi(\mathring{X}_n)$ and $g(\mathring{X}_n)$ for different numbers of variables.

	$JSD[\Phi^{MTE}(\mathring{X}_n);\ \Psi^{MTE}(\mathring{X}_n)]$		
n	$\epsilon = 0.1$	$\epsilon = 0.05$	$\epsilon = 0.01$
4	$1.09 \times 10^{-3} \pm 4.02 \times 10^{-5}$	$2.84 \times 10^{-4} \pm 1.14 \times 10^{-5}$	$1.18 \times 10^{-5} \pm 5.06 \times 10^{-7}$
8	$1.08 \times 10^{-3} \pm 5.39 \times 10^{-5}$	$2.81 \times 10^{-4} \pm 1.51 \times 10^{-5}$	$1.16 \times 10^{-5} \pm 6.64 \times 10^{-7}$
16	$1.11 \times 10^{-3} \pm 1.80 \times 10^{-4}$	$2.90 \times 10^{-4} \pm 5.12 \times 10^{-5}$	$1.20 \times 10^{-5} \pm 2.27 \times 10^{-6}$
32	$1.08 \times 10^{-3} \pm 5.32 \times 10^{-5}$	$2.83 \times 10^{-4} \pm 1.50 \times 10^{-5}$	$1.17 \times 10^{-5} \pm 6.58 \times 10^{-7}$
64	$1.08 \times 10^{-3} \pm 4.58 \times 10^{-5}$	$2.83 \times 10^{-4} \pm 1.30 \times 10^{-5}$	$1.17 \times 10^{-5} \pm 5.77 \times 10^{-7}$
128	$1.06 \times 10^{-3} \pm 4.89 \times 10^{-5}$	$2.77 \times 10^{-4} \pm 1.37 \times 10^{-5}$	$1.15 \times 10^{-5} \pm 6.04 \times 10^{-7}$
256	$1.12 \times 10^{-3} \pm 1.40 \times 10^{-4}$	$2.93 \times 10^{-4} \pm 3.87 \times 10^{-5}$	$1.21 \times 10^{-5} \pm 1.67 \times 10^{-6}$

Table 7: Average Jensen-Shannon Divergences between $\Phi^{MTE}(\mathring{X}_n)$ inferred by MTE and its perturbed distributions.

	$JSD[\Phi^{MOP}(\mathring{X}_n);\ \Psi^{MOP}(\mathring{X}_n)]$		
n	$\epsilon = 0.1$	$\epsilon = 0.05$	$\epsilon = 0.01$
4	$8.86 \times 10^{-4} \pm 1.37 \times 10^{-4}$	$2.31 \times 10^{-4} \pm 3.58 \times 10^{-5}$	$9.54 \times 10^{-6} \pm 1.48 \times 10^{-6}$
8	$8.17 \times 10^{-4} \pm 7.32 \times 10^{-5}$	$2.13 \times 10^{-4} \pm 1.92 \times 10^{-5}$	$8.80 \times 10^{-6} \pm 7.93 \times 10^{-7}$
16	$8.43 \times 10^{-4} \pm 8.58 \times 10^{-5}$	$2.19 \times 10^{-4} \pm 2.16 \times 10^{-5}$	$9.04 \times 10^{-6} \pm 8.65 \times 10^{-7}$
32	$8.33 \times 10^{-4} \pm 7.70 \times 10^{-5}$	$2.17 \times 10^{-4} \pm 1.98 \times 10^{-5}$	$8.96 \times 10^{-6} \pm 8.10 \times 10^{-7}$
64	$8.94 \times 10^{-4} \pm 1.17 \times 10^{-4}$	$2.33 \times 10^{-4} \pm 3.04 \times 10^{-5}$	$9.61 \times 10^{-6} \pm 1.25 \times 10^{-6}$
128	$9.24 \times 10^{-4} \pm 1.54 \times 10^{-4}$	$2.40 \times 10^{-4} \pm 3.91 \times 10^{-5}$	$9.88 \times 10^{-6} \pm 1.57 \times 10^{-6}$
256	$8.59 \times 10^{-4} \pm 1.06 \times 10^{-4}$	$2.24 \times 10^{-4} \pm 2.76 \times 10^{-5}$	$9.25 \times 10^{-6} \pm 1.14 \times 10^{-6}$

Table 8: Jensen-Shannon Divergences for marginalized distributions of $\mathring{X}_n$ in MOPs w.r.t. the marginals obtained using perturbed MOPs.

a slight imprecision in the results of the inferences. All experiments have been performed using the C++ aGrUM library (`http://www.agrum.org`) on a Linux box with an Intel Xeon at 2.40GHz and 128GB of RAM.

5. Conclusion

In this paper, ctdBNs, a new graphical model for handling uncertainty over sets of continuous and discrete variables, have been introduced. We have proved that ctdBNs can approximate (arbitrarily well) any Lipschitz mixed probability distribution. So, theoretically, most of the mixed probability distributions used in real-world situations can be approximated by

ctdBNs. Experiments highlight that this result is not only theoretic: in practice, ctdBNs are very expressive and can be exploited efficiently for diagnosis and classification tasks. A junction tree-based inference algorithm has also been provided. Its theoretical computational complexity has been given and it shows that inference in ctdBNs is essentially similar to that in classical discrete BNs. Here again, the experiments provided in the paper highlight the tractability of inference in practical situations.

For future works, we plan to enrich ctdBNs by allowing the conditional truncated densities assigned to the continuous nodes to depend not only on their discretized counterpart but also on other nodes, in particular their parents. This shall increase the expressive power of the model. In addition, if only parents are added, the conditional truncated densities are defined over the same discrete nodes as the CPTs of the discretized nodes. This shall ensure tractability of inference since the computational complexity of inference shall remain of the same order of that in a classical BN. But the expressive power shall be increased. In some sense, a ctdBN learning algorithm has already been provided in [18]. This algorithm raises some issues, notably the fact that computing discretizations conditionally to the nodes in the Markov blankets of each discretized node limits the ctdBNs that can be learnt. For instance, a discretized node with no parent and many children has a large Markov blanket, which may prevent discretization to be possible due to too high a memory requirement. Therefore, new algorithms are needed for learning ctdBNs from data. This is especially necessary if the set of parents of the continuous nodes $\mathring{X}_i$ is no more limited to the discretized counterpart X_i.

6. Acknowledgments

This work was supported by European project H2020-ICT-2014-1 #644425 Scissor.

Appendix: proofs

Proof of Lemma 1 By definition, $f(\mathring{X}_i|X_i)$ and $P(X_i)$ are non-negative real-valued functions, hence $f(\mathring{X}_i|X_i)P(X_i)$ is also a non-negative real-valued function. So, to prove that it is a mixed probability distribution, it is sufficient to show that:

$$\sum_{x_i \in \Omega_{X_i}} \int_{\Omega_{\mathring{X}_i}} f(\mathring{x}_i|x_i)P(x_i)\ d\mathring{x}_i = 1.$$

By Property 1., $f(\mathring{X}_i = \mathring{x}_i | X_i = x_i)P(X_i = x_i) = 0$ for all $x_i \in \Omega_{X_i}$ and $\mathring{x}_i \notin [t_{x_i}, t_{x_i+1})$. So, the above equation is equivalent to:

$$\sum_{x_i \in \Omega_{X_i}} \int_{t_{x_i}}^{t_{x_i+1}} f(\mathring{x}_i|x_i)P(x_i)\ d\mathring{x}_i = 1,$$

which, by the fact that x_i is a constant inside the integral and by Equation (3), is also equivalent to:

$$\sum_{x_i \in \Omega_{X_i}} P(x_i) \int_{t_{x_i}}^{t_{x_i+1}} f(\mathring{x}_i|x_i)\ d\mathring{x}_i = \sum_{x_i \in \Omega_{X_i}} P(x_i) = 1.$$

As a consequence, $f(\mathring{X}_i|X_i)P(X_i)$ is a mixed probability distribution. ■

Proof of Proposition 1 First, note that all the terms in the product are non-negative real-valued functions, hence h is also a non-negative real-valued function. Let

$$\begin{aligned}\alpha &= \sum_{x_1 \in X_1} \cdots \sum_{x_n \in X_n} \int_{\mathring{X}_{d+1}} \cdots \int_{\mathring{X}_n} \prod_{i=1}^{n} P(x_i|\mathbf{Pa}(x_i)) \prod_{i=d+1}^{n} f(\mathring{x}_i|x_i) d_{\mathring{x}_n} \cdots d_{\mathring{x}_{d+1}} \\ &= \sum_{x_1 \in X_1} \cdots \sum_{x_n \in X_n} \prod_{i=1}^{n} P(x_i|\mathbf{Pa}(x_i)) \times \\ &\qquad \left(\int_{\mathring{X}_{d+1}} f(\mathring{x}_{d+1}|x_{d+1}) d_{\mathring{x}_{d+1}} \right) \cdots \left(\int_{\mathring{X}_n} f(\mathring{x}_n|x_n) d_{\mathring{x}_n} \right).\end{aligned}$$

By Property 2 of Definition 4, each integral of a conditional truncated density is equal to 1, hence:

$$\alpha = \sum_{x_1 \in X_1} \cdots \sum_{x_n \in X_n} \prod_{i=1}^{n} P(x_i|\mathbf{Pa}(x_i)).$$

This formula is also equal to 1 since its terms constitute a discrete BN. Therefore, $h(\mathbf{X})$ is a mixed probability distribution. ■

Proof of Proposition 2 Without loss of generality, we will assume in the sequel that $\epsilon \leq \min\{|\Omega_i| : i \in \{1, \ldots, n\}\}$, where $|\Omega_i|$ denotes the size of domain Ω_i. In addition, let us denote by $\mathbf{B}(\mathring{x}, r)$ the intersection of the hyperball of radius r centered on $\mathring{x}$ with $\mathring{\mathbf{\Omega}}_\mathbf{C}$. First, let us show that there exists $\mathring{a} \in \mathring{\mathbf{\Omega}}_\mathbf{C}$ such that, for every $\mathring{x} \in \mathring{\mathbf{\Omega}}_\mathbf{C}$, we have $f(\mathring{x}) \leq \epsilon$ whenever

$||\mathring{x}|| \geq ||\mathring{a}||$. Proof by contradiction: assume that, for every $\mathring{a} \in \mathring{\mathbf{\Omega}}_{\mathbf{C}}$, there exists $\mathring{x} \in \mathring{\mathbf{\Omega}}_{\mathbf{C}}$ such that $||\mathring{x}|| \geq ||\mathring{a}||$ and $f(\mathring{x}) > \epsilon$. Then:

$$\int_{\mathring{\mathbf{\Omega}}_{\mathbf{C}}} f(\mathring{x})d\mathring{x} = \int_{\mathbf{B}(0,||\mathring{a}||)} f(\mathring{x})d\mathring{x} + \int_{\{\mathring{x}\in\mathring{\mathbf{\Omega}}_{\mathbf{C}}:||\mathring{x}||\geq||\mathring{a}||\}} f(\mathring{x})d\mathring{x}.$$

By hypothesis, there exists $\mathring{b} \in \mathring{\mathbf{\Omega}}_{\mathbf{C}}$ such that $||\mathring{b}|| \geq ||\mathring{a}|| + \frac{\epsilon}{4M}$ and $f(\mathring{b}) > \epsilon$. So, since f is a probability density function, i.e., it is a positive function, the last term of the above equation is such that:

$$\int_{\{\mathring{x}\in\mathring{\mathbf{\Omega}}_{\mathbf{C}}:||\mathring{x}||\geq||\mathring{a}||\}} f(\mathring{x})d\mathring{x} \geq \int_{\mathring{x}\in\mathbf{B}\left(\mathring{b},\frac{\epsilon}{4M}\right)} f(\mathring{x})d\mathring{x} + \int_{\{\mathring{x}\in\mathring{\mathbf{\Omega}}_{\mathbf{C}}:||\mathring{x}||\geq||\mathring{b}||+\frac{\epsilon}{4M}\}} f(\mathring{x})d\mathring{x}$$

and since f is Lipschitz, for every $\mathring{x}$ inside Ball $\mathbf{B}\left(\mathring{b}, \frac{\epsilon}{4M}\right)$, we have that $|f(\mathring{x}) - f(\mathring{b})| \leq M||\mathring{x} - \mathring{y}|| \leq 2M\frac{\epsilon}{4M} = \frac{\epsilon}{2}$. As $f(\mathring{b}) > \epsilon$, we can deduce that $f(\mathring{x}) > \epsilon/2$ for every $\mathring{x} \in \mathbf{B}\left(\mathring{b}, \frac{\epsilon}{4M}\right)$ and, therefore, that the middle term in the above equation is greater than $\frac{\epsilon}{2}\int_{\mathring{x}\in\mathbf{B}\left(\mathring{b},\frac{\epsilon}{4M}\right)} 1d\mathring{x}$. This last integral corresponds to the volume of the intersection of the n-dimensional hyperball of radius $\frac{\epsilon}{4M}$ centered on $\mathring{b} = (\mathring{b}_1, \ldots, \mathring{b}_n)$ with $\mathring{\mathbf{\Omega}}_{\mathbf{C}}$. As $\epsilon \leq \min\{|\Omega_i|\}$, for each random variable $\mathring{X}_i$, at least one interval among $(\mathring{b}_i - \frac{\epsilon}{4M}, \mathring{b}_i)$ and $(\mathring{b}_i, \mathring{b}_i + \frac{\epsilon}{4M})$ belongs to Ω_i. So the integral is greater than or equal to $1/2^n$ of the volume of an n-dimensional hyperball of radius r, which is equal to $\alpha = \pi^{n/2}r^n/\Gamma(\frac{n}{2}+1)$. Consequently,

$$\int_{\mathring{\mathbf{\Omega}}_{\mathbf{C}}} f(\mathring{x})d\mathring{x} > \int_{\mathbf{B}(0,||\mathring{a}||)} f(\mathring{x})d\mathring{x} + \frac{\alpha\epsilon}{2^{n+1}} + \int_{\{\mathring{x}\in\mathring{\mathbf{\Omega}}_{\mathbf{C}}:||\mathring{x}||\geq||\mathring{b}||+\frac{\epsilon}{4M}\}} f(\mathring{x})d\mathring{x}.$$

By our contradiction hypothesis, the same process can be applied to the last term and, by induction, it is possible to construct an infinite sequence $\{\mathring{b}(i)\}_{i\geq 0}$ such that $\mathring{b}(0) = \mathring{b}$ and, for all $i \geq 1$, $||\mathring{b}(i)|| \geq ||\mathring{b}(i-1)|| + \frac{\epsilon}{4M}$ and $f(\mathring{b}(i)) > \epsilon$. Thus, for all $k > 0$,

$$\int_{\mathring{\mathbf{\Omega}}_{\mathbf{C}}} f(\mathring{x})d\mathring{x} > \int_{\mathring{x}:||\mathring{x}||<||\mathring{a}||} f(\mathring{x})d\mathring{x} + k\frac{\alpha\epsilon}{2^{n+1}} + \int_{\{\mathring{x}\in\mathring{\mathbf{\Omega}}_{\mathbf{C}}:||\mathring{x}||\geq||\mathring{b}(k-1)||+\frac{\epsilon}{4M}\}} f(\mathring{x})d\mathring{x}.$$

So $\int_{\mathring{\mathbf{\Omega}}_{\mathbf{C}}} f(\mathring{x})d\mathring{x}$ tends toward $+\infty$, which is impossible since f is a probability density function (hence its integral over $\mathring{\mathbf{\Omega}}_{\mathbf{C}}$ is equal to 1). Therefore, there necessarily exists $\mathring{a} \in \mathring{\mathbf{\Omega}}_{\mathbf{C}}$ such that, for every $\mathring{x} \in \mathring{\mathbf{\Omega}}_{\mathbf{C}}$, we have $f(\mathring{x}) \leq \epsilon$ whenever $||\mathring{x}|| \geq ||\mathring{a}||$.

Now, for any continuous variable $\mathring{X}_i$ of $\mathring{\mathbf{X}}_\mathbf{C}$, let $t_i^- = \max\{\inf \mathring{X}_i, -||\mathring{a}||\}$ and $t_i^+ = \min\{\sup \mathring{X}_i, ||\mathring{a}||\}$. Define a discretization function d_i of $\mathring{X}_i$ by its set of cutpoints $\{t_i^k\}$:

$$\left\{ t_i^k = t_i^- + k\frac{\epsilon}{\sqrt{n}M} : k \in \{0, \ldots, g_i\} \right\} \quad \text{with} \quad g_i = 1 + \left\lfloor \frac{\sqrt{n}M(t_i^+ - t_i^-)}{\epsilon} \right\rfloor.$$

Applying discretization function d_i to $\mathring{X}_i$, we obtain a discretized random variable X_i of domain Ω_i. Let $\mathbf{X_D}$ be the set of all these discretized variables and let $\mathbf{\Omega_D} = \prod_{i=1}^n \Omega_i$. Finally, for any value x_i of discretized variable X_i, denote by $\mathring{\mathbf{\Omega}}_{i|x_i}$ the subdomain of variable $\mathring{X}_i$ compatible with x_i, i.e., $\mathring{\mathbf{\Omega}}_{i|x_i} = [t_i^{x_i}, t_i^{x_i+1})$ if $x_i \notin \{0, g_i\}$, $\mathring{\mathbf{\Omega}}_{i|x_i} = \{\mathring{x}_i < t_i^0\}$ if $x_i = 0$ and $\mathring{\mathbf{\Omega}}_{i|x_i} = \{\mathring{x}_i \geq t_i^{g_i}\}$ if $x_i = g_i$. Let $\mathring{\mathbf{\Omega}}_{|x} = \prod_{i=1}^n \mathring{\mathbf{\Omega}}_{i|x_i}$.

We can now construct a joint probability distribution over $\mathbf{\Omega_D}$ and conditional truncated densities as follows: for every $x = (x_1, \ldots, x_n) \in \mathbf{\Omega_D}$, partition the set of indices $\{1, \ldots, n\}$ into $L = \{i : \mathring{\mathbf{\Omega}}_{i|x_i} \text{ is bounded}\}$, $L_{-\infty} = \{i : \inf \mathring{\mathbf{\Omega}}_{i|x_i} = -\infty\}$ and $L_{+\infty} = \{i : \sup \mathring{\mathbf{\Omega}}_{i|x_i} = +\infty\}$. Fix the joint probability value of $x = (x_1, \ldots, x_n)$ to $P(x) = \int_{\mathring{\mathbf{\Omega}}_{|x}} f(\mathring{x})d\mathring{x}$ and define for all $\mathring{x}_i \in \mathring{\mathbf{\Omega}}_{|x_i}$ the truncated conditional density function $h(\mathring{x}_i|x_i)$ as:

$$h(\mathring{x}_i|x_i) = \begin{cases} \left(\frac{\epsilon}{\beta}\right)^n e^{-\left\{\frac{\pi}{4}\left(\frac{\epsilon}{\beta}\right)^{2n}(\mathring{x}_i - t_i^0)^2\right\}} & \text{if } i \in L_{-\infty} \\ \sqrt{n}M/\epsilon & \text{if } i \in L \\ \left(\frac{\epsilon}{\beta}\right)^n e^{-\left\{\frac{\pi}{4}\left(\frac{\epsilon}{\beta}\right)^{2n}(\mathring{x}_i - t_i^{g_i})^2\right\}} & \text{if } i \in L_{+\infty} \end{cases}$$

where $\beta = \max\{1, \sqrt{n}M\}$. Then, it is easy to see that $P(\cdot)$ is non-negative and that $\sum_{x \in \mathbf{\Omega_D}} P(x) = \int_{\mathring{\mathbf{\Omega}}_\mathbf{C}} f(\mathring{x})d\mathring{x} = 1$. In addition, $\int_{\mathring{\mathbf{\Omega}}_{|x_i}} h(\mathring{x}_i|x_i)d\mathring{x}_i = 1$ for all x_i since the formulas for $i \in L_{-\infty} \cup L_{+\infty}$ are nothing else than twice the density function of a Normal distribution of variance $\sqrt{\frac{2}{\pi}}\left(\frac{\beta}{\epsilon}\right)^n$. So for all x_i, since $h(\mathring{x}_i|x_i) \geq 0$, h is a truncated conditional density function. Consequently $P(x)\prod_{i=1}^n h(\mathring{x}_i|x_i)$ defines a mixed probability distribution.

Now, let us show that, for all $\mathring{x} \in \mathring{\mathbf{\Omega}}_\mathbf{C}$, $|f(\mathring{x}) - P(x)\prod_{i=1}^n h(\mathring{x}_i|x_i)| \leq \epsilon$, where x is the discretized value of $\mathring{x}$. Consider any element $\mathring{x} \in \mathring{\mathbf{\Omega}}_\mathbf{C}$. First, assume that $L_{+\infty} \cup L_{-\infty} \neq \emptyset$, then $||\mathring{x}|| \geq ||\mathring{a}||$ and, consequently, $f(\mathring{x}) \leq \epsilon$. By definition, $P(x) \leq 1$. In addition, for all $i \in L_{+\infty} \cup L_{-\infty}$, $h(\mathring{x}_i|x_i) \leq$

$\left(\frac{\epsilon}{\beta}\right)^n$. So, if $N_\infty = |L_{-\infty}| + |L_{+\infty}|$, then:

$$P(x)\prod_{i=1}^{n} h(\mathring{x}_i|x_i)| \leq \left(\frac{\epsilon}{\beta}\right)^{nN_\infty} \times \left(\frac{\sqrt{n}M}{\epsilon}\right)^{n-N_\infty}.$$

As $N_\infty \geq 1$, $nN_\infty \geq n \geq n - N_\infty + 1$. Hence, as $\beta = \max\{1, \sqrt{n}M\}$ and $\epsilon/\beta \leq \epsilon < 1$,

$$P(x)\prod_{i=1}^{n} h(\mathring{x}_i|x_i)| \leq \epsilon \times \left(\frac{\epsilon}{\beta}\right)^{n-N_\infty} \times \left(\frac{\sqrt{n}M}{\epsilon}\right)^{n-N_\infty} \leq \epsilon.$$

If, on the contrary, $L_{+\infty} \cup L_{-\infty} = \emptyset$, then all the $\Omega_{i|x_i}$ are bounded and their sizes are all equal to $\epsilon/(\sqrt{n}M)$, so $\mathring{\mathbf{\Omega}}_{|x}$ is also bounded. Let $f^- = \min_{\mathring{x}\in\mathring{\mathbf{\Omega}}_{|x}} f(\mathring{x})$ and $f^+ = \max_{\mathring{x}\in\mathring{\mathbf{\Omega}}_{|x}} f(\mathring{x})$. Then:

$$\int_{\Omega_{i|x_i}} f^- dx = \left(\frac{\epsilon}{\sqrt{n}M}\right)^n f^- \leq P(x) \leq \left(\frac{\epsilon}{\sqrt{n}M}\right)^n f^+ = \int_{\Omega_{i|x_i}} f^+ dx.$$

As all the $h(\mathring{x}_i|x_i)$'s are equal to $\sqrt{n}M/\epsilon$, we have $f^- \leq P(x)\prod_{i=1}^n h(\mathring{x}_i|x_i) \leq f^+$. Now, for any pair of elements $(\mathring{y}, \mathring{z})$ of $\mathring{\mathbf{\Omega}}_{|x}$, $||\mathring{y} - \mathring{z}|| < \sqrt{n(\epsilon/\sqrt{n}M)^2} = \frac{\epsilon}{M}$. So, as f is Lipschitz, $|f(\mathring{y}) - f(\mathring{z})| \leq M\frac{\epsilon}{M} = \epsilon$. As a consequence, $f^+ - f(\mathring{x}) \leq \epsilon$ and $f(\mathring{x}) - f^- \leq \epsilon$. Hence, $|f(\mathring{x}) - P(x)\prod_{i=1}^n h(\mathring{x}_i|x_i)| \leq \epsilon$.

To complete the proof, note that any joint distribution $P(X_1, \ldots, X_n)$ can be rewritten as $P(X_1, \ldots, X_n) = P(X_1)\prod_{i=2}^n P(X_i|X_1, \ldots, X_{i-1})$. Using this decomposition, we obtain a ctdBN whose discrete and continuous nodes are $\{X_1, \ldots, X_n\}$ and $\{\mathring{X}_1, \ldots, \mathring{X}_n\}$ respectively, and in which the parents of discretized node X_i are the set $\{X_1, \ldots, X_{i-1}\}$, the conditional probability $P(X_i|X_1, \ldots, X_{i-1})$ resulting from the joint probability $P(X_1, \ldots, X_n)$ are defined in the above paragraphs. Finally, to each X_i is assigned as a child a continuous node $\mathring{X}_i$ whose conditional truncated density is $h(\mathring{X}_i|X_i)$ as defined in the above paragraph. As shown above, this ctdBN approximates f up to ϵ. ■

Proof of Corollary 1 The absolute value of the derivative of the density function of a univariate normal distribution is highest at its inflection point, which corresponds to $x = \mu \pm \sigma$. At that point, the derivative is equal to $M = \exp(-0.5)/(\sqrt{2\pi\sigma^2})$. So the univariate normal distribution is Lipschitz. By Proposition 2, it can be approximated up to ϵ by a ctdBN.

The density function of a multivariate normal distribution is equal to:

$$f(\mathring{x}) = \frac{1}{\sqrt{(2\pi)^n|\boldsymbol{\Sigma}|}} \exp\left[-\frac{1}{2}(\mathring{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathring{x}-\boldsymbol{\mu})\right].$$

$\boldsymbol{\Sigma}^{-1}$ being invertible, it is diagonalizable and its eigenvalues are all different from 0. By expressing $\mathring{x}$ in the basis of the eigenvectors, $f(\mathring{x})$ becomes a product of univariate normal distributions and the preceding paragraph implies that f is Lipschitz and can be approximated up to ϵ by a ctdBN.

The density of the Beta distribution is $f(\mathring{x}) = \mathring{x}^{\alpha-1}(1-\mathring{x})^{\beta-1}/B(\alpha,\beta)$, with $B(\alpha,\beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha+\beta)$. The derivative is therefore equal to $f'(\mathring{x}) = [(\alpha-1)(1-x)-(\beta-1)x](\alpha-1)x^{\alpha-2}(1-x)^{\beta-2}/B(\alpha,\beta)$ If $2 = \alpha < \beta$, then $|f'(\mathring{x})|$ is maximal when $\mathring{x} = 0$ and it is equal to $(\alpha-1)/B(\alpha,\beta)$. If $2 = \beta < \alpha$, then $|f'(\mathring{x})|$ is maximal when $\mathring{x} = 1$ and it is equal to $(\beta-1)/B(\alpha,\beta)$. Finally, if $2 < \alpha$ and $2 < \beta$, it is known that the Beta distribution is bell-shaped with two inflection points at $\mathring{x} = (\alpha-1\pm\sqrt{\frac{(\alpha-1)(\beta-1)}{\alpha+\beta-3}})/(\alpha+\beta-1)$. So the derivative is bounded and f is Lipschitz.

When $\alpha > 2$, the Gamma distribution is bell-shaped and its inflection points are $\beta(\alpha-1\pm\sqrt{\alpha-1})$. Hence, the distribution is Lipschitz.

To complete the proof, consider a set $\mathring{\mathbf{X}}_\mathbf{C}$ of sets of random variables $\mathring{\mathbf{X}}_\mathbf{C} = \{\mathring{\mathbf{X}}_1,\ldots,\mathring{\mathbf{X}}_n\}$ such that all the $\mathring{\mathbf{X}}_i$'s are mutually independent. Let f_i, $i = 1,\ldots,n$, be the respective density functions of the $\mathring{\mathbf{X}}_i$'s and assume that all the f_i's belong to the probability density functions defined in the above paragraphs. Then, every f_i can be approximated up to $\epsilon_0 = \epsilon/2^n$ by some ctdBN $\mathcal{B}_i$ defining a mixed probability distribution g_i, i.e., for any $\mathring{x}_i \in \mathring{\mathbf{X}}_i$, $|f_i(\mathring{x}_i) - g_i(x_i,\mathring{x}_i)| \leq \epsilon_0$, where x_i correspond to the discretized value of $\mathring{x}_i$. Now, the joint density function $f : \mathring{\mathbf{X}}_\mathbf{C} \mapsto \mathbb{R}$ is defined as $f(\mathring{x}) = \prod_{i=1}^n f_i(x_i)$, for all $x = (x_1,\ldots,x_n)$ since the $\mathring{\mathbf{X}}_i$'s are mutually independent. Let $\mathcal{B}$ be the ctdBN resulting from the union of all the $\mathcal{B}_i$'s, i.e., its graphical structure is the union of all the graphical structures of the $\mathcal{B}_i$'s and $\mathcal{B}$ represents mixed probability $g(x,\mathring{x}) = \prod_{i=1}^n g_i(x_i,\mathring{x}_i)$. So, we have that $g(x,\mathring{x}) \leq \prod_{i=1}^n (f_i(\mathring{x}_i)+\epsilon_0)$. Let $\mathcal{S}_k$ be the set of k-subsets of $\{1,\ldots,n\}$. Then:

$$\prod_{i=1}^n (f_i(\mathring{x}_i)+\epsilon_0) = \prod_{i=1}^n f_i(\mathring{x}_i) + \sum_{k=0}^{n-1}\sum_{S\in\mathcal{S}_k}\left(\epsilon_0^{n-k}\prod_{j\in S} f_j(\mathring{x}_j)\right). \tag{9}$$

As the f_j's are probability density functions, $\prod_{j\in S} f_j(\mathring{x}_j) \leq 1$. In addition, for every n,k, we have that $\epsilon_0^{n-k} \leq \epsilon_0$ since $\epsilon_0 < 1$. Finally, $\sum_{k=0}^{n-1}\sum_{S\in\mathcal{S}_k} 1 <$

2^n since this corresponds to the size minus 1 of the power set of $\{1, \ldots, n\}$. So, the right hand side of Equation (9) is less than or equal to $\prod_{i=1}^{n} f_i(\mathring{x}_i) + 2^n\epsilon_0 = f(\mathring{x}) + \epsilon$. We can show similarly, that $g(x, \mathring{x}) \geq f(\mathring{x}) - \epsilon$. So ctdBN $\mathcal{B}$ approximates up to ϵ probability density function f. ■

Proof of Proposition 3 If $\mathbf{X_D} = \emptyset$, then Proposition 3 exactly corresponds to Proposition 2. So, assume that $\mathbf{X_D} \neq \emptyset$.

For every $y = (y_1, \ldots, y_d) \in \mathbf{\Omega_D}$, let $\pi(y) = \int_{\mathring{x} \in \mathring{\mathbf{X}}_\mathbf{C}} f(y, \mathring{x}) d\mathring{x}$. As y corresponds to the discrete part of f, $\pi(y)$ corresponds to the probability of y w.r.t. f. So $k_y : \mathring{\mathbf{\Omega}}_\mathbf{C} \mapsto \mathbb{R}$ defined as $k_y(\mathring{x}) = f(y, \mathring{x})/\pi(y)$ for all $\mathring{x} \in \mathring{\mathbf{\Omega}}_\mathbf{C}$ is a probability density function. In addition, by the hypotheses of Proposition 3, it is Lipschitz. Hence the proof of Proposition 2 can be applied on it. Let us call $\mathring{a}_y$ the vector $\mathring{a}$ of this proof applied on $k_y(\cdot)$. In addition, let $\mathring{a}$ denote a vector of $\{\mathring{a}_y : y \in \mathbf{\Omega_D}\}$ with the highest L2-norm. Then, for every $y \in \mathbf{\Omega_D}$ and any $\mathring{x} \in \mathring{\mathbf{\Omega}}_\mathbf{C}$, we have $k_y(\mathring{x}) \leq \epsilon$ whenever $||\mathring{x}|| \geq ||\mathring{a}||$. Applying the proof of Proposition 2, with this value of $\mathring{a}$, we can therefore perform the same discretization of $\mathring{x}$ into x and construct the same conditional truncated density functions $h(\mathring{x}|x)$ for all the values of y. The proof of Proposition 2 also shows that, by setting $P_y(x) = \int_{\mathring{\mathbf{\Omega}}_{|x}} k_y(\mathring{x}) d\mathring{x}$, then we have that:

$$\left| k_y(\mathring{x}) - P_y(x) \prod_{i=d+1}^{n} h(\mathring{x}_i|x_i) \right| \leq \epsilon, \quad \text{for all } \mathring{x} \in \mathring{\mathbf{\Omega}}_\mathbf{C} \text{ and all } y \in \mathbf{\Omega_D}, \quad (10)$$

and $\sum_x P_y(x) = 1$ for every y. In other words, $P_y(x)$ corresponds to a conditional distribution of x given y. Define $P(y, x) = P_y(x) \times \pi(y)$. Then, $P(y, x)$ corresponds to the joint distribution of x and y (i.e., $\sum_{x,y} P(y, x) = 1$). So, as $k_y(\mathring{x}) = f(y, \mathring{x})/\pi(y)$ and $\pi(y) \leq 1$ (since it is a probability), Equation (10) implies that:

$$\left| f(y, \mathring{x}) - P(y, x) \prod_{i=d+1}^{n} h(\mathring{x}_i|x_i) \right| \leq \epsilon\pi(y) \leq \epsilon, \quad \text{for all } (y, \mathring{x}) \in \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_\mathbf{C}.$$

The completion of the proof is now the same as that of Proposition 2: joint distribution $P(y, x)$ can be decomposed as $P(y_1) \times \prod_{i=2}^{d} P(y_i|y_1, \ldots, y_{i-1}) \times P(x_{d+1}|y_1, \ldots, y_d) \prod_{j=d+2}^{n} \times P(x_j|y_1, \ldots, y_d, x_{d+1}, \ldots, x_{j-1})$ and the resulting ctdBN follows. ■

Proof of Proposition 4 According to the proof of Proposition 3, since f is Lipschitz, the continuous variables $\mathring{X}_i$ of $\mathring{\mathbf{X}}_\mathbf{C}$ can be discretized into discrete

variables X_i of $\mathbf{X_C} = \{X_{d+1}, \ldots, X_n\}$ and, for all $(y, \mathring{x}) \in \mathbf{\Omega_D} \times \mathring{\mathbf{\Omega}}_\mathbf{C}$, $f(y, \mathring{x})$ can be approximated up to ϵ by $P(y, x) \prod_{i=d+1}^{n} h(\mathring{x}_i | x_i)$, where x is the discretized counterpart of $\mathring{x}$ and $P(y, x)$ is the joint probability distribution defined as:

$$P(y, x) = \int_{\mathring{\mathbf{\Omega}}_{|x}} f(y, \mathring{x}) d\mathring{x}. \tag{11}$$

Let us show that this joint distribution can be decomposed w.r.t. sets $\mathcal{C}_i = \mathbf{X}_{D_i} \cup \mathbf{X}_{C_i} \cup \mathring{\mathbf{X}}_{C_i}$, $i = 1, \ldots, k$. Proof by induction on i: let $i = 1$ and let $\mathbf{X}_{E_1} = \mathbf{X_D} \setminus \mathbf{X}_{C_1}$ and $\mathring{\mathbf{X}}_{E_1} = \mathring{\mathbf{X}}_\mathbf{C} \setminus \mathring{\mathbf{X}}_{C_1}$. Function f can be decomposed as $f(y, \mathring{x}) = f_1(y_{C_1}, \mathring{x}_{C_1}) \times h_1(y_{E_1}, \mathring{x}_{E_1})$, with $h_1(y_{E_1}, \mathring{x}_{E_1}) = \prod_{j=2}^{k} f_j(y_{C_j}, \mathring{x}_{C_j})$. Note that f_1 and h_1 share no variable in common. Then Equation (11) is equal to:

$$\begin{aligned} P(y, x) &= \int_{\mathring{\mathbf{\Omega}}_{|x_{C_1}}} \int_{\mathring{\mathbf{\Omega}}_{|x_{E_1}}} f_1(y_{C_1}, \mathring{x}_{C_1}) \times h_1(y_{E_1}, \mathring{x}_{E_1}) d\mathring{x}_{E_1} d\mathring{x}_{C_1} \\ &= \int_{\mathring{\mathbf{\Omega}}_{|x_{C_1}}} f_1(y_{C_1}, \mathring{x}_{C_1}) d\mathring{x}_{C_1} \int_{\mathring{\mathbf{\Omega}}_{|x_{E_1}}} h_1(y_{E_1}, \mathring{x}_{E_1}) d\mathring{x}_{E_1} \end{aligned}$$

Now, we also have that:

$$\begin{aligned} P(y_{C_1}, x_{C_1}) &= \sum_{y_{E_1}} \sum_{x_{E_1}} P(y, x) \\ &= \int_{\mathring{\mathbf{\Omega}}_{|x_{C_1}}} f_1(y_{C_1}, \mathring{x}_{C_1}) d\mathring{x}_{C_1} \sum_{y_{E_1}} \sum_{x_{E_1}} \int_{\mathring{\mathbf{\Omega}}_{|x_{E_1}}} h_1(y_{E_1}, \mathring{x}_{E_1}) d\mathring{x}_{E_1} \\ &= \int_{\mathring{\mathbf{\Omega}}_{|x_{C_1}}} f_1(y_{C_1}, \mathring{x}_{C_1}) d\mathring{x}_{C_1} \sum_{y_{E_1}} \int_{\mathring{\mathbf{\Omega}}_{E_1}} h_1(y_{E_1}, \mathring{x}_{E_1}) d\mathring{x}_{E_1} \end{aligned}$$

Note that, by definition, $h_1(y_{E_1}, \mathring{x}_{E_1}) = \prod_{j=2}^{k} f_j(y_{C_j}, \mathring{x}_{C_j})$ is a mixed probability distribution over $\mathbf{\Omega}_{E_1} \times \mathring{\mathbf{\Omega}}_{E_1}$. Consequently, we have that:

$$\sum_{y_{E_1}} \int_{\mathring{\mathbf{\Omega}}_{E_1}} h_1(y_{E_1}, \mathring{x}_{E_1}) d\mathring{x}_{E_1} = 1,$$

and, therefore, that $P(y_{C_1}, x_{C_1}) = \int_{\mathring{\mathbf{\Omega}}_{|x_{C_1}}} f_1(y_{C_1}, \mathring{x}_{C_1}) d\mathring{x}_{C_1}$. We can prove in a similar way that $P(y_{E_1}, x_{E_1}) = \int_{\mathring{\mathbf{\Omega}}_{|x_{E_1}}} h_1(y_{E_1}, \mathring{x}_{E_1}) d\mathring{x}_{E_1}$. Consequently, $P(y, x) = P(y_{C_1}, x_{C_1}) P(y_{E_1}, x_{E_1})$. So, P can be decomposed similarly to f as concerns clique $\mathcal{C}_1$. By induction, we can repeat the same process with mixed probability h_1 rather than f and the result follows. ■

Proof of Proposition 5 By definition, the product of all the functions stored into junction tree $\mathcal{T}$ is a mixed probability distribution $P(x, \mathring{x}, e)$. So it satisfies the Shafer-Shenoy axioms [24, 2]. As a consequence, the message-passing algorithm is sound and, for each clique, the function resulting from the multiplication of the functions stored in any clique by the messages sent to this clique is the joint (mixed) probability of the variables of the clique and evidence e. So, if the clique contains only discrete variables, after normalizing this resulting function, we necessarily get a joint posterior distribution of the variables of the clique. On the other hand, if the clique contains a continuous variable $\mathring{X}_k$, then the resulting function is necessarily the posterior mixed distribution of X_k and $\mathring{X}_k$ given evidence e and, by marginalizing out X_k, we get the posterior density function of $\mathring{X}_k$. ■

Proof of Proposition 6 There are at most n continuous random variables. Computing each message they send to their neighbor corresponds to perform $|\Omega_{X_i}| \leq k$ integrals. Hence the overall complexity of computing all these messages is in $O(nk\overline{I})$. As there are n random variables, there are at most n cliques containing only discrete variables. The complexity of computing their messages in both directions is therefore in $O(nk^{w+1})$. For the same reason, the complexity of sending messages from the cliques with only discrete variables to the cliques containing continuous variables is also $O(nk^{w+1})$. To compute the posterior of any discrete or discretized variable, it is sufficient to select one clique that contains it, to multiply the tables stored into this clique by all the messages sent to the clique and, then to marginalize out all the other variables. When performing the distribution phase, the tables stored into the clique are already multiplied by all the messages sent to the clique except one. So, to compute the posterior of the discrete variable, we just need to perform the last product required, with a complexity in $O(k^{w+1})$ and the summation (marginalizing-out) has the same complexity. Finally, there are n continuous variables. To compute the posterior density of a continuous variable, we must perform the operations of Equation (7). There are at most k products to perform and each product has an average complexity of $\overline{J}$, hence the overall complexity of computing all the posterior densities is in $O(nk\overline{J})$. Overall, we get the complexity stated in the proposition. ■

References

[1] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kauffmann, 1988.

[2] G. Shafer, Probabilistic expert systems, Society for Industrial and Applied Mathematics, 1996.

[3] R. Dechter, Bucket elimination: A unifying framework for reasoning, Artificial Intelligence 113 (1999) 41–85.

[4] A. Madsen, F. Jensen, LAZY propagation: A junction tree inference algorithm based on lazy inference, Artificial Intelligence 113 (1–2) (1999) 203–245.

[5] F. Bacchus, S. Dalmao, T. Pitassi, Algorithms and complexity results for #sat and Bayesian inference, in: Proceedings of FOCS'03, 2003, pp. 340–351.

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

[7] S. Lauritzen, N. Wermuth, Graphical models for associations between variables, some of which are qualitative and some quantitative, Annals of Statistics 17 (1) (1989) 31–57.

[8] S. Lauritzen, Propagation of probabilities, means and variances in mixed graphical association models, Journal of the American Statistical Association 87 (1992) 1098–1108.

[9] U. Lerner, E. Segal, D. Koller, Exact inference in networks with discrete children of continuous parents, in: Proceedings of UAI'01, 2001, pp. 319–328.

[10] S. Moral, R. Rumí, A. Salmerón, Mixtures of truncated exponentials in hybrid Bayesian networks, in: Proceedings of ECSQARU'01, Vol. 2143 of LNAI, 2001, pp. 156–167.

[11] B. Cobb, P. Shenoy, R. Rumí, Approximating probability density functions in hybrid Bayesian networks with mixtures of truncated exponentials, Statistics and Computing 16 (3) (2006) 293–308.

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

[13] H. Langseth, T. Nielsen, R. Rumí, A. Salmerón, Mixtures of truncated basis functions, International Journal of Approximate Reasoning 53 (2) (2012) 212–227.

[14] H. Langseth, T. Nielsen, R. Rumí, A. Salmerón, Inference in hybrid Bayesian networks with mixtures of truncated basis functions, in: Proceedings of PGM'12, 2012, pp. 171–178.

[15] P. Shenoy, A re-definition of mixtures of polynomials for inference in hybrid Bayesian networks, in: Proceedings of ECSQARU'11, Vol. 6717 of LNCS, 2011, pp. 98–109.

[16] P. Shenoy, J. West, Inference in hybrid Bayesian networks using mixtures of polynomials, International Journal of Approximate Reasoning 52 (5) (2011) 641–657.

[17] P. Shenoy, Two issues in using mixtures of polynomials for inference in hybrid Bayesian networks, International Journal of Approximate Reasoning 53 (5) (2012) 847–866.

[18] A. Mabrouk, C. Gonzales, K. Jabet-Chevalier, E. Chojnaki, Multivariate cluster-based discretization for Bayesian network structure learning, in: Proceedings of SUM'15, 2015.

[19] P. Shenoy, Inference in hybrid Bayesian networks using mixtures of Gaussians, in: Proceedings of UAI'06, 2006, pp. 428–436.

[20] W. Poland, R. Shachter, Three approaches to probability model selection, in: R. L. de Mantaras, D. Poole (Eds.), Proceedings of UAI'94, 1994, pp. 478–483.

[21] B. Cobb, P. Shenoy, Inference in hybrid Bayesian networks with mixtures of truncated exponentials, International Journal of Approximate Reasoning 41 (3) (2006) 257–286.

[22] S. Moral, R. Rumí, A. Salmerón, Estimating mixtures of truncated exponentials from data, in: Proceedings of PGM'02, 2002, pp. 135–143.

[23] R. Romero, R. Rumí, A. Salmerón, Structural learning of Bayesian networks with mixtures of truncated exponentials, in: Proceedings of PGM'04, 2004, pp. 177–184.

[24] P. Shenoy, G. Shafer, Axioms for probability and belief function propagation, in: Proceedings of UAI'90, 1990, pp. 169–198.

[25] P. Shenoy, Binary join trees for computing marginals in the Shenoy-Shafer architecture, International Journal of Approximate Reasoning 17 (1) (1997) 1–25.

[26] J. Baron, Second-order probabilities and belief functions, Theory and Decision 23 (1) (1987) 25–36.

[27] F. van den Eijkhof, H. L. Bodlaender, Safe reduction rules for weighted treewidth, in: Proceedings of WG'02, Vol. 2573 of LNCS, 2002, pp. 176–185.

[28] M. Lichman, UCI machine learning repository (2013).
URL `http://archive.ics.uci.edu/ml`

[29] N. Friedman, M. Goldszmidt, Discretizing continuous attributes while learning Bayesian networks, in: proceedings of ICML'96, 1996, pp. 157–165.

[30] W. K. Hastings, Monte carlo sampling methods using markov chains and their applications, Biometrika 57 (1) (1970) 97–109.