Constraint-Based Learning for Non-Parametric Continuous Bayesian Networks

Marvin Lasserre, Régis Lebrun, Pierre-Henri Wuillemin

The Thirty-Third International
FLAIRS Conference (FLAIRS-33)

Constraint-Based Learning for Non-Parametric Continuous Bayesian Networks

Marvin Lasserre,[1] **Régis Lebrun,**[2] **Pierre-Henri Wuillemin**[1]
[1]Laboratoire d'Informatique de Paris 6
4 place Jussieu, 75005 Paris, France
firstname.lastname@lip6.fr
[2]Airbus AI Research
22 rue du Gouverneur Général Eboué
92130 Issy les Moulineaux, France
firstname.lastname@airbus.com

Abstract

Modeling high-dimensional multivariate distributions is a computationally challenging task. Bayesian networks have been successfully used to reduce the complexity and simplify the problem with discrete variables. However, it lacks of a general model for continuous variables. In order to overcome this problem, (Elidan 2010) proposed the model of copula bayesian networks that reparametrizes bayesian networks with conditional copula functions. We propose a new learning algorithm for copula bayesian networks based on a PC algorithm and a conditional independence test proposed by (Bouezmarni, Rombouts, and Taamouti 2009). This test being non-parametric, no model assumptions are made allowing it to be as general as possible. This algorithm is compared on generated data with the score based method proposed by (Elidan 2010). Not only it proves to be faster, but also it generalizes well on data generated from distributions far from the gaussian model.

1 Introduction

Modeling multivariate continuous distributions is an important task in statistics and machine learning with many applications in science and engineering. However, high-dimensional distributions are hard to manipulate and may lead to intractable computations. In addition, apart from simple parametric models such as the gaussian distribution, univariate distributions usually don't have multivariate equivalents leading to difficulties in building multivariate models.

Probabilistic graphical models are used to compactly represent multivariate distributions. In particular, bayesian networks (BN) use a directed acyclic graph (DAG) and a set of conditional probability distributions (CPD) to encode the distribution. This representation reduces the complexity by taking advantage of conditional independencies, allowing efficient inference and learning algorithms. However, BNs lack of a general model for continuous variables: most of the time, discretizations or gaussian models are used despite no theoretical restrictions on CPD models. On the one hand, discretizations need to be determined and are limited in the number of bins that are used. Gaussian models on the other hand allow efficient inference and learning algorithms but lack of expressiveness.

According to Sklar (theorem 1), any multivariate distribution is related to its univariate marginals by means of a copula function. Thus, the copula function allows to model the dependence structure between continuous variables by ruling out the marginal behavior of each variable. From a constructive perspective, this allows to dissociate the choice of the marginals and the dependence structure. In practice however, copulas are limited to a few variables and constructing or manipulating high-dimensional ones is difficult.

In order to take advantage of these two frameworks, many graphical models for copulas have been proposed such as pair-copula construction (Czado 2010), Vine model (Bedford, Cooke, and others 2002) or cumulative distribution networks (Huang 2009). One encouraging model is the Copula Bayesian Network (CBN) (Elidan 2010) which parametrizes a BN with a set of local conditional copula functions giving it the same local properties. Consequently, this allows to use similar methods than in the classical case for inference and learning. In this regard, (Elidan 2010) proposed a learning method based on the well known BIC score, maximized with a TABU search.

The main contribution of this paper is a new learning algorithm for CBNs. This learning algorithm relies on a PC-algorithm coupled with a continuous conditional independence (CI) test proposed by (Bouezmarni, Rombouts, and Taamouti 2009) that uses Bernstein copula estimators. The method is compared to the BIC score method in terms of structural scores and time complexity on generated data sets.

The paper is organized as follows. Section 2 describes copulas and some of their useful properties. Section 3 introduces the CBN framework proposed by (Elidan 2010). Section 4 presents in details the two learning algorithms for CBNs, that is our algorithm and the method proposed in (Elidan 2010). Section 5 compares the algorithms on generated data from known structures in terms of structure learning and time complexity. Section 6 concludes the paper.

2 Copulas

Let $\overline{\mathbb{R}}$ be the extended set of real numbers defined as $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ and $\mathbf{I}$ be the unit segment $[0, 1]$. Let $\mathbf{X} =$

$(X_1, \ldots, X_n)$ be an n-dimensional random vector and $\mathbf{x} = (x_1, \ldots, x_n)$ a vector of $\overline{\mathbb{R}}^n$ denoting a realization of $\mathbf{X}$.

Definition 1 (Distribution Function)**.** *The distribution function $H : \overline{\mathbb{R}}^d \rightarrow \mathbf{I}$ of a random vector $\mathbf{X}$ is given by*

$$H(x_1, \ldots, x_n) = \mathbb{P}\left(X_1 \leqslant x_1, \ldots, X_n \leqslant x_n\right).$$

The distribution function respects the following properties:

1. $H(x_1, \ldots, x_n) = 0$ *if there exists i such that $x_i = -\infty$,*
2. $H(+\infty, \ldots, +\infty) = 1$.

The 1-dimensional marginal distributions[1] F_i, for each individual random variable X_i, are obtained by the formula $F_i(x_i) = H(+\infty, \ldots, x_i, \ldots, +\infty)$.

When variables are independent, the joint distribution can be expressed in terms of its univariate marginals: $H(x_1, \ldots, x_n) = \prod_{i=1}^{n} F_i(x_i)$. Thus, giving any set of arbitrary marginal distributions F_i, a joint distribution can be constructed by taking their product. Copula functions allow to achieve the same objective but with dependent variables.

Definition 2 (Copula)**.** *Let $\mathbf{U} = \{U_1, \ldots, U_n\}$ be a random vector whose components are uniformly distributed on $\mathbf{I}$. A copula function $C : \mathbf{I}^n \rightarrow \mathbf{I}$ is a distribution:*

$$C(u_1, \ldots, u_n) = \mathbb{P}(U_1 \leqslant u_1, \ldots, U_n \leqslant u_n)$$

The relation between the joint distribution and its univariate marginals is a central result of copula theory due to (Sklar 1959):

Theorem 1 (Sklar 1959)**.** *Let H be any multivariate distribution function over a random vector $\mathbf{X}$, there exists a copula function C such that*

$$H\left(x_1, \ldots, x_n\right) = C\left(F_1(x_1), \ldots, F_n(x_n)\right). \qquad (1)$$

Furthermore, if each $F_i(x_i)$ is continuous then C is unique.

As the marginals encode the individual behavior of each variables, the copula function C encodes the dependence between these variables. This is interesting from a constructive perspective since the choice of marginals can be separated from the choice of the dependence structure. Moreover, Sklar's theorem may be used to construct new copulas from known multivariate distributions by inverting[2] (1) :

$$C(u_1, \ldots, u_n) = H(F_1^{-1}(u_1), \ldots, F_n^{-1}(u_n))$$

where $u_i = F(x_i)$. Taking $H = \Phi_R$, the multivariate standard gaussian distribution with correlation matrix R , we obtain the well known gaussian copula (Nelsen 2007) :

$$C_G(u_1, \ldots, u_n) = \Phi_R(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_n))$$

where ϕ is the univariate standard gaussian distribution. Copula functions are invariant under increasing transformations of the random variables. Indeed, let $\{\psi_i\}$ be a family of such transformations and let $U_i = \psi_i(X_i)$, then

$$H'(u_1, \ldots, u_n) = C'(F_1'(u_1), \ldots, F_n'(u_n)).$$

[1]When it is clear from context, the index i will be dropped in order to alleviate notations.

[2]The univariate marginals may be not invertible and in this case the inverse has to be replaced by the generalized inverse F^* defined as $F^*(y) = \inf\{x | F(x) \geqslant y\}$.

By definition of marginal distributions,

$$\begin{aligned} F_i'(u_i) &= \mathbb{P}(U_i \leqslant u_i) = \mathbb{P}(\psi_i(X_i) \leqslant u_i) \\ &= \mathbb{P}(X_i \leqslant \psi_i^{-1}(u_i)) = F(\psi_i^{-1}(u_i)) \end{aligned}$$

and injecting it in the previous equation, it gives that

$$\begin{aligned} H'(u_1, \ldots, u_n) &= \mathbb{P}(U_1 \leqslant u_1, \ldots, U_n \leqslant u_n) \\ &= \mathbb{P}(X_1 \leqslant \psi_1^{-1}(u_1), \ldots, X_n \leqslant \psi_n^{-1}(u_n)) \\ &= H(\psi_1^{-1}(u_1), \ldots, \psi_n^{-1}(u_n)) \\ &= C(F_1(\psi_1^{-1}(u_1)), \ldots, F_n(\psi_n^{-1}(u_n))) \\ &= C(F_1'(u_1), \ldots, F_n'(u_n)) \end{aligned}$$

hence $C' = C$. Using this last property with $\psi_i = F_i$, we have that $H'(u_1, \ldots, u_n) = C(u_1, \ldots, u_n)$ which allows to work directly with the copula function and to look at the dependence structure. However, in many applications the F_i's are usually unknown and rank variables R_i are used instead. Given a database $\mathcal{D}$ of size M, the rank variable $R_i[m]$ is obtained as the rank of $X_i[m]$ among the set of instances.

If a distribution function is continuous, its joint density is obtained by deriving it : $h(\mathbf{x}) = \frac{\partial^n H(x_1, \ldots, x_n)}{\partial x_1 \ldots \partial x_n}$. A copula density function can be equivalently defined by derivation $c(u_1, \ldots, u_n) = \frac{\partial^n C(u_1, \ldots, u_n)}{\partial u_1 \ldots \partial x_n}$. Using Sklar's theorem, the joint density is then related to the copula density by:

$$\begin{aligned} h(x_1, \ldots, x_n) &= \frac{\partial^n H(x_1, \ldots, x_n)}{\partial x_1 \ldots \partial x_n} \\ &= \frac{\partial^n C(F_1(x_1), \ldots, F_n(x_n))}{\partial F_1(x_1) \ldots \partial F(x_n)} \prod_{i=1}^{n} \frac{\partial F_i(x_i)}{\partial x_i} \\ &= c\left(F_1(x_1), \ldots, F_n(x_n)\right) \prod_{i=1}^{n} f_i(x_i). \qquad (2) \end{aligned}$$

This formula will be used extensively in the next section to define CBNs.

3 Copula Bayesian Networks

A BN structure $\mathcal{G}$ is a DAG whose vertices $\mathbf{X} = \{X_1, \ldots, X_n\}$ represent random variables. Let $\mathbf{Pa}_i$ be the parents of X_i in $\mathcal{G}$ and $\mathbf{ND}_i$ be the variables that are non-descendants of X_i in the graph. A multivariate probability distribution P over variables $\mathbf{X}$, is said to factorize according to $\mathcal{G}$, if it can be expressed as the product

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | \mathbf{Pa}_i). \qquad (3)$$

and $\mathcal{G}$ then encodes the set of independencies :

$$\mathcal{I}(\mathcal{G}) = \{(X_i \perp \mathbf{ND}_i | \mathbf{Pa}_i)\}.$$

A BN is a pair $\mathcal{B} = (\mathcal{G}, P)$ where $\mathcal{G}$ is defined as previously and P factorizes over $\mathcal{G}$. To each node X_i of the BN structure is associated its corresponding CPD $P(X_i | \mathbf{Pa}_i)$ that appears in the factorization of the joint distribution P.

In the discrete case, CPDs are most often represented via conditional probability tables (CPT) while in the continuous case, they are represented via linear gaussian model

(Lauritzen and Wermuth 1989) $f(x_i|\mathbf{pa}_i) = \mathcal{N}(\beta_{i0} + \beta_i^T \mathbf{pa}_i; \sigma_i^2)$. Although gaussian distributions allow fast probabilistic computations and estimation, they lack of expressiveness and some distributions, like rare events ones, cannot be well approximated by such models. The CBN model introduced by (Elidan 2010) tries to address this problem by using copula functions to parametrize the BN.

In order to do so, the first step is to use (2) in the Bayes formula for $f(x_i|\mathbf{pa}_i)$:

$$\begin{aligned} f(x_i|\mathbf{pa}_i) &= \frac{f(x_i, \mathbf{pa}_i)}{f(\mathbf{pa}_i)} \\ &= \frac{c(F(x_i), F(\text{pa}_{i1}), \ldots, F(\text{pa}_{ik_i}) f(x_i) \prod_{j=1}^{k_i} f(\text{pa}_{ij})}{\frac{\partial^{k_i} C(1, F(\text{pa}_{i1}), \ldots, F(\text{pa}_{ik_i}))}{\partial F(\text{pa}_{i1}) \ldots \partial F(\text{pa}_{ik_i})} \prod_{j=1}^{k_i} f(\text{pa}_{ij})} \\ &= \frac{c(F(x_i), F(\text{pa}_{i1}), \ldots, F(\text{pa}_{ik_i}) f(x_i)}{\frac{\partial^{k_i} C(1, F(\text{pa}_{i1}), \ldots, F(\text{pa}_{ik_i}))}{\partial F(\text{pa}_{i1}) \ldots \partial F(\text{pa}_{ik_i})}} \\ &= R_{c_i}(F(x_i), F(\text{pa}_{i1}), \ldots, F(\text{pa}_{ik_i})) f(x_i) \end{aligned}$$

where $k_i = |\mathbf{pa}_i|$. Consequently, if $f(\mathbf{x})$ that is supposed to be strictly positive, factorizes on $\mathcal{G}$ as $f(\mathbf{x}) = \prod_{i=1}^{n} f(x_i|\mathbf{pa}_i)$, it is the same for the copula density :

$$\begin{aligned} c(F(x_1), \ldots, F(x_n)) &= \frac{f(\mathbf{x})}{\prod_{i=1}^{n} f(x_i)} = \frac{\prod_{i=1}^{n} f(x_i|\mathbf{pa}_i)}{\prod_{i=1} f(x_i)} \\ &= \frac{\prod_{i=1}^{n} R_{c_i}(F(x_i), F(\text{pa}_1), \ldots, F(\text{pa}_{K_i})) f(x_i)}{\prod_{i=1}^{n} f(x_i)} \\ &= \prod_{i=1}^{n} R_{c_i}(F(x_i), F(\text{pa}_1), \ldots, F(\text{pa}_{K_i})). \end{aligned}$$

Like with BNs, the converse is also true :

Theorem 2 (Elidan 2010)**.** *Let $\mathcal{G}$ be a DAG over $\mathbf{X}$. In addition, let $\{c_i(F(x_i), F(pa_{i1}), \ldots, F(pa_{ik_i}))\}$ be a set of strictly positive copula densities associated with the nodes of $\mathcal{G}$ that have at least one parent. If $\mathcal{I}(\mathcal{G})$ holds then the function*

$$h(F(x_1), \ldots, F(x_n)) = \prod_{i=1}^{n} R_{c_i}(F(x_i), \{F(pa_{ik})\}) f(x_i),$$

is a valid density over $\mathbf{X}$.

This leads to the definition of a CBN as given by (Elidan 2010) :

Definition 3 (Copula Bayesian Network)**.** *A Copula Bayesian Network is a triplet $\mathcal{C} = (\mathcal{G}, \Theta_C, \Theta_f)$ that encodes the joint density $f(\mathbf{x})$. Θ_C is a set of local copula densities functions $c_i(F(x_i), \{F(pa_{ik})\})$ that are associated with the nodes of $\mathcal{G}$ that have at least one parent. Θ_f is the set of parameters representing the marginal densities $f(x_i)$. $f(\mathbf{x})$ is parametrized as*

$$f(\mathbf{x}) = \prod_{i=1}^{n} R_{c_i}(F(x_i), \{F(pa_{ik})\}) f(x_i). \qquad (4)$$

4 Learning

CBNs share the same local properties as the (classical) BNs allowing to use similar algorithms in order to learn the structure of a CBN. Those algorithms can be roughly divided into two classes: score based methods and constrained based methods. For score based method, the learning task is viewed as a model selection and a scoring function is used to measure how good the model fit the dataset. The space of all DAG structures being superexponential, this score is often maximized using local search methods such as hill-climbing, gradient ascent, simulated annealing, TABU list, etc. Constrained-based methods on the other hand look at the graph as a set of (conditional) independences and use CI tests, such as χ^2 in the discrete case, to obtain information about the underlying structure. We present one method of each kind in this section and compare them in the next section.

Score based method (CBIC)

In (Elidan 2010), a score-based method is used to learn the structure of a CBN. The proposed score is the well-known bayesian information criterion (BIC) (Schwarz 1978). Its expression on a CBN structure $\mathcal{G}$ is given by :

$$\mathcal{S}_{BIC}(\mathcal{G} : \mathcal{D}) = \ell(\mathcal{D} : \hat{\theta}, \mathcal{G}) - \frac{1}{2} \log(M) |\Theta_{\mathcal{G}}|,$$

where ℓ is the log-likelihood, $\hat{\theta}$ are the maximum likelihood parameters estimators (MLE) and $|\Theta_{\mathcal{G}}|$ is the number of free parameters associated with the graph structure. Using the factorization of the joint density (4), we have :

$$\ell(\mathcal{D} : \mathcal{G}) = \sum_{m=1}^{M} \sum_{i=1}^{N} \log R_i \left(u_i[m], \pi_{i1}[m]), \ldots, \pi_{ik_i}[m])\right)$$

where $u_i = F(x_i)$ and $\pi_{ij} = F(\text{pa}_{ij})$. (Elidan 2010) uses several copula models to define the R_{c_i}'s but we only retain the most expressive one which is the gaussian copula model parametrized by a full correlation matrix Σ. Finding directly the MLE for Σ may be difficult in high dimension and this is why a proxy is used. It relies on the relation $\Sigma_{ij} = \sin(\frac{\pi}{2}\tau_{ij})$ between Kendall's tau τ_{ij} and correlation matrix Σ_{ij} that holds for every elliptical distribution (Lindskog, McNeil, and Schmock 2003). The τ_{ij} are given by

$$\tau(X_i, X_j) = \mathbb{E}\left[\text{sign}\left((X_i - \tilde{X}_i)(X_j - \tilde{X}_j\right)\right].$$

where $(\tilde{X}_i, \tilde{X}_j)$ is an independent copy of (X_i, X_j). An estimator of Kendall's tau is given by (Genest and Favre 2007)

$$\begin{aligned} \tau(X_i, X_j) = \frac{2}{M(M-1)} \times \\ \sum_{m_1=1}^{M-1} \sum_{m_2 > m_1}^{M} \text{sign}\Big(\quad (X_i[m_1] - X_i[m_2]) \times \\ (X_j[m_1] - X_j[m_2])\Big). \end{aligned}$$

However, the matrix obtained by this process is not necessarily a correlation matrix, that is a positive semi-definite

(PSD) matrix, and regularization techniques may be needed to obtain one (Rousseeuw and Molenberghs 1993). Finally, the BIC score is maximized using a TABU list algorithm with random restarts (Glover and Laguna 1998).

Continuous PC algorithm (CPC)

The PC algorithm introduced by (Spirtes et al. 2000) and on which relies our method can be divided in three main steps : skeleton learning, v-structures search and constraint propagation. The skeleton search consists in removing edges from the complete non-oriented graph on $\mathbf{X}$ by using CI tests between pairs of variables conditioned on subset of common neighbors. Once this first step is completed, the triplets $X - Y - Z$ such that X and Z are not neighbors and Y is not in **Sepset**(X, Z), are oriented as $X \rightarrow Y \leftarrow Z$ which is called a v-structure. Finally, the remaining non-oriented edges are oriented under the constraint that no new v-structures are added into the graph unless it implies adding an oriented cycle. For further details on the PC algorithm, see page 84 of (Spirtes et al. 2000).

The CI test, which is based on Hellinger's distance, is taken from (Bouezmarni, Rombouts, and Taamouti 2009; 2010) and (Su and White 2008). Taking two random variables X, Y and $\mathbf{Z} = \{Z_1, \ldots, Z_d\}$ a set of random variables; and with $C_{X,Y,\mathbf{Z}}$ a copula and $c_{X,Y,\mathbf{Z}}$ its density, the article proposes to test:

$$X \perp\!\!\!\perp Y \,|\, \mathbf{Z} \iff \mathbb{P}\left(c_{XY|\mathbf{Z}} = c_{X|\mathbf{Z}} \cdot c_{Y|\mathbf{Z}}\right) = 1$$

The Hellinger's distance is then used as a measure of the conditional independence[3]:

$$H(Y, Z|\mathbf{X}) = \int_{[0,1]^{d+2}} \left(1 - \sqrt{\frac{c_{X,\mathbf{Z}}(x,\mathbf{z}) \cdot c_{Y,\mathbf{Z}}(y,\mathbf{z})}{c_{X,Y,\mathbf{Z}}(x,y,\mathbf{z}) \cdot c_{\mathbf{Z}}(\mathbf{z})}}\right)^2 \times c_{X,Y,\mathbf{Z}}(x,y,\mathbf{z}) dx dy d\mathbf{z}. \quad (5)$$

From a database, it is possible to derive the non-parametric Bernstein copula $\widehat{C}_{X,Y,\mathbf{Z}}$ (Sancetta and Satchell 2004) as an estimation of the copula $C_{X,Y,\mathbf{Z}}$. We can then estimate the distance of Hellinger by:

$$\widehat{H} = \frac{\sum_{m=1}^{M} \left(1 - \sqrt{\frac{\widehat{c}_{X,\mathbf{Z}}(x[m],\mathbf{z}[m]) \cdot \widehat{c}_{Y,\mathbf{Z}}(y[m],\mathbf{z}[m])}{\widehat{c}_{X,Y,\mathbf{Z}}(x[m],y[m],\mathbf{z}[m]) \cdot \widehat{c}_{\mathbf{Z}}(\mathbf{z}[m])}}\right)^2}{M} \quad (6)$$

where the $(x[m], y[m], \mathbf{z}[m])$ are the realizations of the variables $(X, Y, \mathbf{Z})$ in the database of M samples for the copula C. Based on this estimation of the distance, (Bouezmarni, Rombouts, and Taamouti 2009) proposes a BRT statistic for CI test[4] for any dimension of $\mathbf{Z}$. Under the assumption $H_0 : X \perp\!\!\!\perp Y \mid \mathbf{Z}$, it can be proven that $\mathrm{BRT} \sim \mathcal{N}(0, 1)$.

[3]Some formulas like the equation 5 were a bit wrong in (Bouezmarni, Rombouts, and Taamouti 2009; Wan and Zabaras 2014) and have been fixed here.

[4]For the expression of BRT, we refer to theorem 1 of (Bouezmarni, Rombouts, and Taamouti 2009).

Our contribution is a PC algorithm using a continuous CI test relying on the BRT to learn CBNs. This method follows the same idea from the work of (Wan and Zabaras 2014) which proposes a learning procedure to factorize a joint distribution and then learn a mixture of gaussians for the CPDs. However, in the case of (Wan and Zabaras 2014), the structure learning and parameter learning models being different, this can lead to non-consistent results. In our case, copulas are at the core of the model since they are used to parametrize the CBN and using a copula based CI test makes perfect sense.

5 Experimental Results

This section presents the results of the comparison between CPC and CBIC methods[5]. The experiments have been carried out with the C++ libraries aGrUM (Gonzales, Torti, and Wuillemin 2017), which allows to build graphical models, and OpenTURNS (Baudin et al. 2015) which allows to model continuous multivariate probabilistic distributions.

Simulation setup

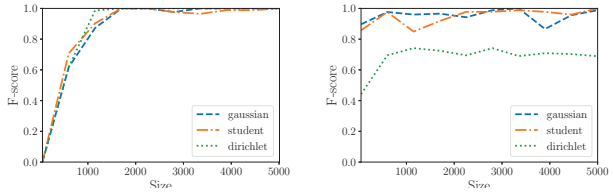
The two algorithms have been tested on simulated data from the Asia (Lauritzen and Spiegelhalter 1988) and Alarm (Beinlich et al. 1989) networks. Asia is a relatively small graph containing 8 nodes and 8 arcs while Alarm is much bigger, containing 37 nodes and 46 arcs. In order to obtain continuous data from these structures, a forward sampling has been used to generate data from Gaussian, Student and Dirichlet copulas. These copulas are used as the local copulas appearing in the R_{c_i} coefficients of the CBN.

Skeleton performances

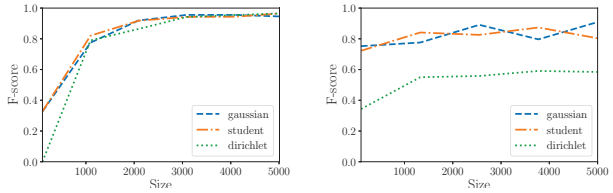
The structural performances of the two learning algorithms have been computed by comparing the skeleton of the learned graph with the skeleton of the true structure that have been used to generate the data. Precision (P) is the proportion of learned edges that are actually in the true structure while recall (R) is the proportion of edges that are in the true structure that have been recovered. The F-score is then defined as $\mathrm{F} = 2\mathrm{PR}/(\mathrm{P} + \mathrm{R})$. If the true skeleton has been perfectly retrieved, the value of the F-score is 1. Figure 1 shows the results in terms of F-score for Asia and Alarm network and both methods.

As can be seen, CBIC performs better on data generated from gaussian and Student copulas since it needs less data to recover the true structure. This is an expected behavior since the gaussian assumption is true, or close to the true model in these cases. However, it performs poorly with data generated from Dirichlet copulas and cannot recover the true structure. Although it needs more data to recover the true structure, continuous PC performs well and equally on each copula model, illustrating the strength of a non-parametric method.

[5]While linear gaussian model is the standard when learning BNs with continuous variables, it has not been compared to our model since it turns out to be less efficient than the CBIC method (Elidan 2010).

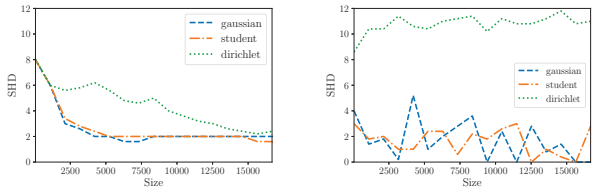


(a) CPC on Asia (b) CBIC on Asia

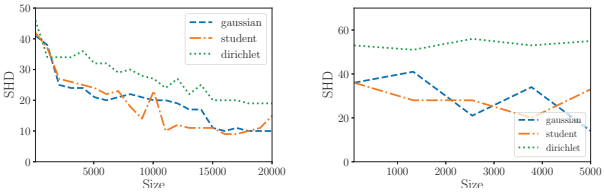


(c) CPC on Alarm (d) CBIC on Alarm

Figure 1: Evolution of the F-score for CPC and CBIC methods with respect to the size of the dataset for gaussian (dashed orange line), Student (dot-dashed orange line) and Dirichlet (doted green line) distributions. The results are averaged over 5 restarts in the case of Asia.



(a) CPC on Asia (b) CBIC method on Asia



(c) CPC on Alarm (d) CBIC on Alarm

Figure 2: Evolution of the structural hamming distance for CPC and CBIC methods with respect to the size of the data set for gaussian (dashed orange line), Student (dot-dashed orange line) and Dirichlet (doted green line) distributions. The results are averaged over 5 restarts for the case of Asia.

CPDAG performances

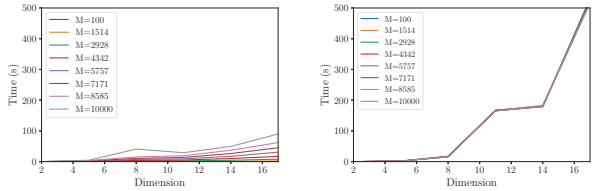
In order to score the oriented structure, structural hamming distance (Colombo and Maathuis 2014) has been used. This metric works on the completed partially directed acyclic graphs (CPDAG) that represents the Markov class equivalences of the DAG (Koller and Friedman 2009) and counts the numbers of elementary operations that are needed to obtain the true structure from the estimated one. Those transformations are edge insertions, deletions and flipping. Figure 2 shows the results for Asia and Alarm network.

These results are similar from the ones of the skeleton. Indeed, as can be seen, the CBIC method recovers almost perfectly the CPDAG in the case of Gaussian and Student copulas but does not in the case of Dirichlet copula. Continuous PC, on the other hand, is less performing on recovering the CPDAG but keeps the property to be indifferent to the distribution model.

Time complexity

The two methods have been compared in terms of time complexity. To do so, random graphs of different sizes have been generated with an MCMC method (Ide and Cozman 2002) and used to generate data. Finally, the learning time on these data has been measured for both methods with respect to the dimension and for several sizes of data set. The results are shown on figure 3.

As can be seen, the time complexity for CBIC mainly depends on the dimension. However, this complexity grows more rapidly than for the CPC algorithm, resulting in intractable computations for high dimensional data set such as the Alarm network. For this reason, figure 2d is restrained to the domain size $[100, 5000]$.



(a) CPC (b) CBIC

Figure 3: Learning time for CPC (left) and CBIC (right) methods with respect to dimension for several sizes of sample.

6 Conclusion and Future Work

CBN is a promising model for dealing with continuous data in the BN context and for dealing with high-dimensional multivariate distributions from the copula perspective. One of the strength of the model is that it allows to use similar techniques used with classical BNs for inference and learning tasks. In this regard, (Elidan 2010) proposed a score based method using a continuous BIC score. In turn, we proposed a constraint based method which uses a PC algorithm and a non-parametric CI test, thus making no assumptions on the model that generated the data on which the structure is learned. Consequently, this method is more general than the CBIC method, since by essence it is restrained to parametric models. The experimental part illustrated this last property since CPC can deal with data far from the gaussian model such as Dirichlet data. Moreover, even if the gaussian copula model could have been changed, the true model is rarely known in applications. In addition, the time complexity of the local search grows exponentially making it difficult to make computations for high dimensional data set such as the

Alarm network. The source files to manage and learn CBNs can be find in a still experimental plugin of OpenTURNS that uses aGrUM (otagrum).

The provided method allowing us to remove the gaussian hypothesis, it would be interesting to test it on application cases. While the local search maximization of the CBIC is quite slow, it could be interesting to try to decompose the score as in the discrete case (Koller and Friedman 2009). This decomposition involves to study entropy and mutual information in the continuous case which are not equivalent to their discrete counterparts. Studying these quantities in the light of copula theory would be interesting in order to use method that are based on information theory such as MIIC (Affeldt, Verny, and Isambert 2016).

Acknowledgments

This work was partially supported by Airbus Research through the AtRandom project (CRT/VPE/XRD).

References

Affeldt, S.; Verny, L.; and Isambert, H. 2016. 3off2: A network reconstruction algorithm based on 2-point and 3-point information statistics. In *BMC bioinformatics*, volume 17, S12. BioMed Central.

Baudin, M.; Dutfoy, A.; Iooss, B.; and Popelin, A.-L. 2015. Openturns: An industrial software for uncertainty quantification in simulation.

Bedford, T.; Cooke, R. M.; et al. 2002. Vines–a new graphical model for dependent random variables. *The Annals of Statistics* 30(4):1031–1068.

Beinlich, I. A.; Suermondt, H. J.; Chavez, R. M.; and Cooper, G. F. 1989. The alarm monitoring system: A case study with two probabilistic inference techniques for belief networks. In *AIME 89*. Springer. 247–256.

Bouezmarni, T.; Rombouts, J.; and Taamouti, A. 2009. A nonparametric copula based test for conditional independence with applications to granger causality. Economics working papers, Universidad Carlos III, Departamento de Economía.

Bouezmarni, T.; Rombouts, J. V.; and Taamouti, A. 2010. Asymptotic properties of the bernstein density copula estimator for α-mixing data. *Journal of Multivariate Analysis* 101(1):1 – 10.

Colombo, D., and Maathuis, M. H. 2014. Order-independent constraint-based causal structure learning. *The Journal of Machine Learning Research* 15(1):3741–3782.

Czado, C. 2010. Pair-copula constructions of multivariate copulas. In *Copula theory and its applications*. Springer. 93–109.

Elidan, G. 2010. Copula bayesian networks. In *Advances in neural information processing systems*, 559–567.

Genest, C., and Favre, A.-C. 2007. Everything you always wanted to know about copula modeling but were afraid to ask. *Journal of hydrologic engineering* 12(4):347–368.

Glover, F., and Laguna, M. 1998. *Tabu Search*. Boston, MA: Springer US. 2093–2229.

Gonzales, C.; Torti, L.; and Wuillemin, P.-H. 2017. aGrUM: a Graphical Universal Model framework. In *International Conference on Industrial Engineering, Other Applications of Applied Intelligent Systems*, Proceedings of the 30th International Conference on Industrial Engineering, Other Applications of Applied Intelligent Systems.

Huang, J. C. 2009. *Cumulative distribution networks: Inference, estimation and applications of graphical models for cumulative distribution functions*. Citeseer.

Ide, J. S., and Cozman, F. G. 2002. Random generation of bayesian networks. In *Brazilian symposium on artificial intelligence*, 366–376. Springer.

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

Lauritzen, S. L., and Spiegelhalter, D. J. 1988. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)* 50(2):157–194.

Lauritzen, S. L., and Wermuth, N. 1989. Graphical models for associations between variables, some of which are qualitative and some quantitative. *The annals of Statistics* 31–57.

Lindskog, F.; McNeil, A.; and Schmock, U. 2003. Kendall's tau for elliptical distributions. In *Credit Risk*. Springer. 149–156.

Nelsen, R. B. 2007. *An introduction to copulas*. Springer Science & Business Media.

Rousseeuw, P. J., and Molenberghs, G. 1993. Transformation of non positive semidefinite correlation matrices. *Communications in Statistics–Theory and Methods* 22(4):965–984.

Sancetta, A., and Satchell, S. 2004. The bernstein copula and its applications to modeling and approximations of multivariate distributions. *Econometric Theory* 20(03):535–562.

Schwarz, G. 1978. Estimating the dimension of a model. *The annals of statistics* 6(2):461–464.

Sklar, A. 1959. Fonctions de répartition à n dimensions et leurs marges. *Publ. Inst. Statist. Univ. Paris* 8:229–231.

Spirtes, P.; Glymour, C. N.; Scheines, R.; Heckerman, D.; Meek, C.; Cooper, G.; and Richardson, T. 2000. *Causation, prediction, and search*. MIT press.

Su, L., and White, H. 2008. A nonparametric hellinger metric test for conditional independence. *Econometric Theory* 24(4):829–864.

Wan, J., and Zabaras, N. 2014. A probabilistic graphical model based stochastic input model construction. *J. Comput. Physics* 272:664–685.