Generalized Ideal Parent (GIP): Discovering non-Gaussian Hidden Variables

Yaniv Tenzer **Ilya Soloveytchik** **Ami Wiesel** **Gal Elidan**

Department of Statistics, The Hebrew University

School of Computer Science and Engineering, The Hebrew University

Abstract

A formidable challenge in uncertainty modeling in general, and when learning Bayesian networks in particular, is the discovery of unknown hidden variables. Few works that tackle this task are typically limited to discrete or Gaussian domains, or to tree structures. We propose a novel approach for discovering hidden variables in flexible non-Gaussian domains using the powerful class of Gaussian copula networks. Briefly, we define the concept of a hypothetically optimal predictor of variable, and show how it can be used to discover useful hidden variables in the expressive framework of copula networks. We demonstrate the merit of our approach for learning succinct models that generalize well in several real-life domains.

1 Introduction

Hidden variables are ubiquitous in most scientific domains ranging from psychology and economics to natural speech recognition. For example, a hidden market factor or trend may jointly influence a collection of stocks, a web user's "mood" may influence his entire online behavior, etc. Indeed, the discovery of hidden variables is of fundamental interest in probabilistic modeling and goes back to the work of Charles Spearman on factor analysis [Spearman, 1904].

Aside from the semantic appeal in terms of interpretability, inclusion of such centralized hidden mechanisms can also lead to succinct models, and thereby improved statistical estimation, often resulting in probabilistic graphical models that generalize well. As an example, consider the Naive Bayes model where all observed variables are independent of each other given a single hidden parent. Obviously, this model is usually an approximation of reality, yet it often performs well and is widely used both in academic and industrial applications [Lazarsfeld et al., 1968]. In this work, we focus on the statistical axis and aim to discover hidden variables in non-Gaussian domains that lead to favorable generalization performance.

Bayesian Networks (BNs) [Pearl, 1988] are widely used to model and reason about high-dimensional distributions via a qualitative graph that captures the independencies of the domain and quantitative parameters. Numerous methods exist for parameter and structure learning with complete or partial observations, as well as in the presence of *known* hidden variables [Koller and Friedman, 2009]. However, significantly fewer works address the greater challenge of *discovering* hidden variables. At the high level, this task consists of two steps. The first is the identification of components (sets of variables) that can benefit from the introduction of a latent factor. On the second step such factors are included into the model, together with the required parameter and structural adaptation.

Unfortunately, works aiming at automatic discovery of hidden variables in BNs are typically either limited to discrete domains [Elidan and Friedman, 2005, Chandrasekaran et al., 2010] or to tree structured models [Zhang and Kocka, 2004, Chen et al., 2008, Kirshner, 2012], or are focused on the Gaussian case [Choi et al., 2011]. Real-valued non-Gaussian domains pose formidable difficulties and, to the best of our knowledge, only a very few works discover hidden variables in this setting. Some are limited to relatively simple functional forms [Elidan et al., 2007, Hoyer et al., 2008, Janzing et al., 2009] while the others are applicable only to tree networks [Kirshner, 2012] (see Section 2 for a detailed discussion of related works). Yet, real-life domains are often far from Gaussian, and most

Appearing in Proceedings of the 19^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2016, Cadiz, Spain. JMLR: W&CP volume 41. Copyright 2016 by the authors.

likely do not have simple tree structures. Our goal in this work is to overcome these barriers and take the discovery of hidden variables in non-Gaussian BNs a step further.

We focus on copula networks: a fusion of BNs and copulas [Nelsen, 2007, Joe, 1997]. Briefly, copulas conveniently allow us to separately model the univariate marginals and the dependence function that joins them. Fused with BNs, the result is an expressive high-dimensional model. Indeed, even with the so-called Gaussian copula, the modelled distribution can be far from Gaussian and the representation leads to appealing predictive gains in various domains (e.g., [Kirshner, 2007, Elidan, 2010, Tenzer and Elidan, 2013]).

We start by generalizing the *Ideal Parent* (IP) concept [Elidan et al., 2007] to Gaussian copula networks. An IP of a random variable X is the hypothetical optimal parent predictor that X could have. More formally, an IP *profile* of a random variable X is a hypothetical vector of realizations (one for each training sample) which, given the current predictors of X, perfectly predicts X. This concept is useful in the context of discovering a hidden variable using the following intuition: if several variables have similar ideal parents, then a single hypothetical parent can be used for their *joint* prediction.

In a non-Gaussian setting, the original IP definition is not directly useful. Instead, we introduce a generalized concept of a *Quazi Ideal Parent* (QIP) and develop the machinery needed for efficiently discovering and embedding useful hidden variables in copula networks.

We use our QIP approach to discover hidden variables in a variety of real domains some of which are markedly larger than those considered by previous works. In all cases, we show that the non-Gaussian representation leads to improved generalization performance. Further, we show that our method is superior to the convex approximation approach of [Chandrasekaran et al., 2010], and is competitive with the state-of-the art LTC method [Kirshner, 2012] (where the latter is applicable), while using substantially more succinct copula network models.

2 Related Works

Numerous works involve learning *in the presence* of hidden variables (see [Koller and Friedman, 2009] for references), including some more recent ones in the context of copula-based models [Dauwels et al., 2013, Rey and Roth, 2012].

There are, however, significantly fewer works that aim to *discover* hidden variables. which is the focus of our work. One of the first general purpose approaches for doing so in the context of BNs simply uses crude structural signatures and avoids the issue of the functional form of the network [Elidan et al., 2000]. Subsequent papers tackling this challenge are focused on either the discrete (e.g. [Elidan and Friedman, 2005]) or tree structured scenarios (e.g. [Zhang, 2004, Chen et al., 2008, Daskalakis et al., 2006]).

More recent developments are still restricted to the discrete or Gaussian scenarios, but offer theoretical guarantees of consistency. However, these rely on assumptions that can be quite unrealistic in practice. For example, in [Choi et al., 2011] a correlation decay is assumed, while in [Chandrasekaran et al., 2010] some assumptions are made regarding the algebraic properties of the matrix sets.

Few works address the challenge of discovering hidden variables in *non-Gaussian* domains. Among the most notable ones is the work of [Hoyer et al., 2008] which is restricted to a linear relationships, and the work of [Janzing et al., 2009] which heavily relies on the Confounder with Additive Noise (CAN) model. Unfortunately, these two works are strongly limited in terms of scalability. To quote [Hoyer et al., 2008]: "High dimensionalities are out the question, so good scalability is probably not needed...". In [Janzing et al., 2009], the settings is further restricted to two observed variables and a single confounder.

Two works are most relevant to ours in the context of high-dimensional non-Gaussian domains. [Elidan et al., 2007] introduced the concept of a hypothetically optimal (ideal) parent *profile* in the Gaussian scenario. They also offer an adaptation to a non-linear case which requires case-by-case tailoring and is in practice limited to simple parametric forms (e.g. sigmoid Gaussian). Our work generalizes the ideal parent concept and introduces the machinery needed to make it useful in the more powerful copula based setting.

The work of [Kirshner, 2012] is most similar to ours. It relies on the copula network representation and also aims to discover hidden variables with better generalization rather than focusing on guaranteed identifiability. While the construction is more amenable than ours to additional copula families, it is only applicable to tree structured networks. Such models can be quite powerful, but a large number of hidden variables is required to capture relatively modest domains. In contrast, our work is applicable to general structures. In fact, even when constrained (for demonstration purposes only) to bipartite networks, as shown in Section 6, our method is on par with [Kirshner, 2012], while using models that are substantially simpler. Further, we show that our approach can be applied to significantly larger domains.

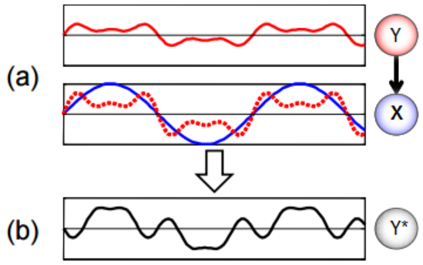


Figure 1: Illustration of the "Ideal Parent" concept for a variable with a single parent Y and a linear link function. The top panel in (a) shows the profile (assignment in all instances) of the parent. The panel below shows the profile of the child node (solid blue) along with the profile predicted for the child based on its parent (dotted red). (b) shows the profile of the ideal hypothetical parent that would lead to zero error in prediction of the child variable if added to the current model.

3 Background

3.1 Bayesian Networks

A Bayesian network (BN) [Pearl, 1988] is used to represent a joint distribution over a finite set of random variables $\mathbf{X} = \{X_1, \dots, X_n\}$ using two components: a directed acyclic graph $\mathcal{G}$ whose vertices correspond to $\mathbf{X}$ encoding the independencies that are assumed to hold, and a set of quantitative parameters of the conditional density of each variable X_i given its parents Par_i in $\mathcal{G}$. The joint density is then defined as $f(\mathbf{X}) = \prod_{i=1}^n f_{\theta_i}(X_i|\mathrm{Par}_i)$.

Given a set of training examples $\mathbf{D} = (\mathbf{x}[1], \dots, \mathbf{x}[M])$ and a structure $\mathcal{G}$, parameter estimation involves finding the parameters $\boldsymbol{\theta}$ maximizing the log-likelihood function $l(\mathbf{D}|\mathcal{G}, \boldsymbol{\theta}) = \log f(\mathbf{D}|\mathcal{G}, \boldsymbol{\theta}) = \sum_m \log(\mathbf{x}[m]|G, \boldsymbol{\theta})$. The common score-based approach for structure learning is to search for $\mathcal{G}$ that maximizes a penalized likelihood measure, such as the Bayesian Information Criterion: $\mathrm{BIC}(\mathbf{D}, \mathcal{G}) = \max_{\boldsymbol{\theta}} l(\mathbf{D}|\mathcal{G}, \boldsymbol{\theta}) - \frac{\log M}{2}|\mathcal{G}|$, where $|\mathcal{G}|$ is the number of parameters in $\mathcal{G}$. Optimization is usually carried out using a greedy search that employs local modifications of the graph structure. See [Koller and Friedman, 2009] for a thorough description of structure learning.

3.2 Copula and Copula Networks

A copula is a flexible general purpose tool for describing real-valued distributions [Joe, 2014, Nelsen, 2007]. Formally, let $U_1, \dots, U_n$ be a set of marginally uniform random variables $U_i \sim U([0,1])$. A copula $C : [0,1]^n \to [0,1]$ is a cumulative distribution function (CDF) over such variables $C(u_1, \dots, u_n) = \mathbb{P}(U_1 \leq u_1, \dots, U_n \leq u_n)$.

Sklar's seminal theorem [Sklar, 1959] shows that for *any* joint distribution $F_{\mathbf{X}}(X_1, \dots, X_n)$, there exists a copula $C(\cdot)$ satisfying $F_{\mathbf{X}}(X_1, \dots, X_n) = C(F_1(X_1), \dots, F_n(X_n))$, where $\{F_i(X_i)\}_{i=1}^n$ are the univariate marginal CDFs. This copula is unique when the marginals are continuous. Conversely, given the marginals $X_1, \dots, X_n$, and any copula, $C(F_1(X_1), \dots, F_n(X_n))$ defines a valid joint distribution whose with marginals $F_i(X_i)$. This provides substantial modeling advantages since the univariate marginals can be estimated independently from the copula that binds them.

The copula density can be derived by differentiating $F_{\mathbf{X}}(X_1, \dots, X_n)$ w.r.t. each X_i, and using $U_i \equiv F_i(X_i)$:

$$f(x_1, \dots, x_n) = \frac{\partial^n C u_1, \dots, u_n}{\partial U_1, \dots \partial U_n} \prod_i f_i(x_i)$$
$$\equiv c(F_1(x_1), \dots, F_n(x_n)) \prod_i f_i(x_i),$$

where $f_i(x_i)$ are the univariate densities and $c(\cdot)$ is the copula density.

Perhaps, the most popular copula family is the *Gaussian copula* defined as [Nelsen, 2007]

$$C_\Sigma(u_1, \dots, u_n) = \Phi_\Sigma\left(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_n)\right),$$

where Φ_Σ is the multivariate normal CDF, Σ is a correlation matrix and Φ is the standard normal CDF. Let ϕ_Σ be the multivariate normal density, let ϕ the univariate standard normal density, and let $z_i \equiv \Phi^{-1}(u_i)$. The copula density is $c_\Sigma(\mathbf{u}) = \phi_\Sigma(\mathbf{z}) / \prod_i \phi(z_i)$. Importantly, this copula can capture distributions that are far from Gaussian and is widely used in numerous domains. See [Embrechts et al., 2003] for more details.

For a vector $\mathbf{v} \in \mathbb{R}^n$, denote $\mathbf{v}_{-i} \equiv (v_1, \dots, v_{i-1}, v_{i+1}, \dots, v_n)$. Given a copula density $c(U_1, \dots, U_n)$, we use $c(U_i|\mathbf{u}_{-i})$ to denote the conditional density of U_i given $\mathbf{u}_{-i}$. For the Gaussian copula, using properties of the Gaussian distribution, this density takes a simple form that will be useful in the sequel: $c(U_i|\mathbf{u}_{-i}) = \phi\left(Z_i|\mathbf{z}_{-i}; \Sigma\right) / \phi(z_i)$, where $\phi\left(Z_i|\mathbf{z}_{-i}; \Sigma\right)$ is the conditional density induced from $\phi_\Sigma(\mathbf{z})$ by conditioning Z_i on $\mathbf{z}_{-i}$.

A *copula network* [Elidan, 2010] is a BN with the local conditional density defined as

$$f_i(X_i|\mathrm{Par}_i) = \frac{c_{\boldsymbol{\theta}}(F_i(X_i), \{F_j(X_j)\}_{j \in \mathrm{Par}_i})}{\int_{X_i} c_{\boldsymbol{\theta}}(F_i(X_i), \{F_j(X_j)\}_{j \in \mathrm{Par}_i})} f_i(X_i).$$

Appealingly, the denominator can be computed from the numerator without integration. Thus, the representation relies solely on the estimation of joint copulas.See [Elidan, 2010] for more details.

4 The Ideal Parent Paradigm

We start this section by an outline on the Ideal Parent (IP) concept [Elidan et al., 2007], we then relax and generalize it for the copula-based setting and, finally, concretize for the powerful case of a Gaussian copula.

4.1 Relaxation of the Ideal Parent

Let $\mathbf{Y} = \{Y_1, \ldots, Y_k\}$ be a set of explanatory variables, and X be a target random variable that depends on $\mathbf{Y}$ through a link function $g : \mathbb{R}^k \times \Omega \to \mathbb{R}$ with parameters $\boldsymbol{\theta} \in \Omega$:

$$X = g(y_1, \ldots, y_k|\boldsymbol{\theta}) + \epsilon, \tag{1}$$

$\epsilon \sim \mathcal{N}(0, \sigma^2)$. For example, if $g(\cdot)$ adds up its arguments, then X follows a Gaussian distribution; if $g(\cdot)$ is the sigmoid of the sum of its arguments, then X follows a log-linear distribution, etc. Note that formally $g(\cdot)$ depends on the number of inputs k, but we omit this dependence to simplify notations.

Given a set of samples of X and $\mathbf{Y}$ and a link function $g(\cdot)$, the Ideal Parent *profile* defined by [Elidan et al., 2007] is a set of realizations $\mathbf{y}^*$, one for each training sample, defining a hypothetical r.v. Y^* such that, X is perfectly predicted by $\mathbf{Y} \cup \{Y^*\}$. Formally,

Definition 4.1. : Given a set of realizations $\{x[m], \mathbf{y}[m]\}_{m=1}^M$ and a link function $g(\cdot)$, $\mathbf{y}^*$ is an Ideal Parent (IP) profile of X if

$$x[m] = g(y_1[m], \ldots, y_k[m], y^*[m] \mid \boldsymbol{\theta}), \ \ \forall m. \tag{2}$$

Figure 1 illustrates the IP profile concept. Note that there could be multiple perfect predictors for X and, thus, an IP does not have to be unique, depending on the specific form of $g(\cdot)$.

Taking the conditional expectation of both sides of Equation (1) and recalling the normality of ϵ, we get $g(\cdot) = \mathbb{E}[X|\mathbf{Y} = y]$. Plug this into Equation (2) to obtain the IP profile as a solution to the system

$$x[m] = \mathbb{E}_{f(X|y_1[m],\ldots,y_k[m],y^*[m])}[X], \ \ \forall m. \tag{3}$$

Next, with the goal of removing the Gaussianiny assumption, we can use Equation (3) as a *relaxed* definition of an IP, and only require that ϵ be centered. We refer to the solution of Equation (3) as the *relaxed IP* of the random variable X. Denoting $\mathbf{F}_{\mathbf{Y}}(\mathbf{y}[m]) \equiv \{F_{Y_1}(y_1[m]), \ldots, F_{Y_k}(y_k[m])\}$, the following lemma provides a rank-based formulation that is equivalent to this relaxed IP definition:

Lemma 4.2.: *Let $c(\cdot)$ be the joint copula density corresponding to the joint distribution of $X, \mathbf{Y}, Y^*$ defined as above and let $U \sim U([0,1])$. The IP profile is given as a solution to the following system of equations:*

$$x[m] = \mathbb{E}_{c(U|\mathbf{F}_{\mathbf{Y}}(\mathbf{y}[m]),F_{Y^*}(y^*[m]))}[F_X^{-1}(U)], \ \ \forall m. \tag{4}$$

The proof is provided in the supplementary material.

The above lemma implies that the IP realizations profile is unique only up to the *CDF transformation* $\{F_{Y^*}(y^*[m])\}_{m=1}^M$, and that, to extract $\mathbf{y}^*$, we will first need to solve Equation (4), and then invert using $F_{Y^*}^{-1}$. This should not come as a surprise since, for the purpose of prediction, we only care about the *dependence* between X and $\mathbf{Y}$ that is independent of the marginal properties of these random variables. Indeed, this is the precise semantics of the copula construction.

4.2 Quasi Ideal Parent

Our goal is now to get an explicit expression for the realizations $\{F_{Y^*}(y^*[m])\}_{m=1}^M$, such that Equation (4) will hold for every m. In Section 5 we will use these realizations as a signature to discover new hidden variables. Unfortunately, this is infeasible even for the simple case of $F_X(X) = \Phi(X)$, where Φ is the standard Gaussian distribution, since we need to solve a set of integral equations that can have a complex form. To overcome this, we adopt the following approximation (see an assessment of its quality below):

Definition 4.3. : Let $X, \mathbf{Y}$ be defined as above and let $U \sim U([0,1])$. Assuming that the dependence structure is modeled by $c(\cdot)$, $\mathbf{y}^*$ is a Quazi Ideal Parent (QIP) realizations profile if

$$x[m] = F_X^{-1}(\mathbb{E}_{c(U|\mathbf{F}_{\mathbf{Y}}(\mathbf{y}[m]),F_{Y^*}(y^*[m]))}[U]), \ \ \forall m. \tag{5}$$

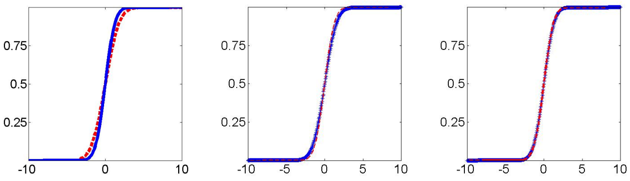
At the technical level, the change from Equation (4) is a simple interchange between the function evaluation and expectation. A similar change has been suggested in the past, albeit in a completely different context (e.g. [Friedman, 1998]). More importantly, similarly to the relaxed IP, the notion of QIP also implies that $\mathbf{y}^*$ realizations can only be identified *up to the CDF transformation* $F_{Y^*}(y^*[m])$, for all m. Applying F_X to both sides of Equation (5), we get

$$F_X(x[m]) = \mathbb{E}_c\left(U|\mathbf{F}_{\mathbf{Y}}(\mathbf{y}[m]), F_{Y^*}(y^*[m])\right), \ \ \forall m. \tag{6}$$

Therefore, QIP is a solution to a system of equations in the *copula domain.*

4.3 QIP for The Gaussian Copula

We are now ready to make the notion of QIP concrete for the case of the Gaussian copula. As we shall see in Section 6, even this "simple" copula leads to clear predictive advantages.



$\sigma = 1$ $\sigma = 0.5$ $\sigma = 0.1$

Figure 2: Evaluation of the quality of the expectation approximation. We compare $\mathbb{E}_{\phi(Z;\mu,\sigma)}[\Phi(Z)]$ (solid blue) and $\Phi\left(\mathbb{E}_{\phi(Z;\mu,\sigma)}[Z]\right)$ (dashed red) as a function of μ (x-axis) for different values of σ.

Recall that a conditional Gaussian $\phi(Z_i|\mathbf{z}_{-i};\Sigma)$ parameterized by a covariance matrix Σ can be equivalently parameterized via a vector of scalars $\boldsymbol{\beta}_{-i} \equiv (\beta_1,\ldots,\beta_{i-1},\beta_{i+1},\ldots,\beta_n)$, corresponding to $\mathbf{z}_{-i}$, and a variance term σ^2 [Bilodeau and Brenner, 1999]. We will use this parametrization and the notation $\phi(Z_i|\mathbf{z}_{-i};\boldsymbol{\beta}_{-i},\sigma)$ to represent the conditional Gaussian. We begin by deriving an explicit form to the RHS of Equation (6) under the Gaussian copula model:

Lemma 4.4.: *Assume* $\mathbf{U} \sim C_\Sigma$ *with* C_Σ *denoting the Gaussian copula with correlation matrix* Σ*, and let* $\mathbf{Z} \equiv (\Phi^{-1}(U_1),\ldots,\Phi^{-1}(U_n))$*, then*

$$\mathbb{E}_{C_\Sigma}[U_i|\mathbf{u}_{-i}] = \mathbb{E}_{\phi(Z_i|\mathbf{z}_{-i};\boldsymbol{\beta}_{-i},\sigma)}[\Phi(Z_i)].$$

The proof can be found in the supplementary material. Applying this lemma to Equation (6), we get that in the case of the Gaussian copula the QIP can be characterizes by a system of equations in $z^*[m] \equiv \Phi^{-1}(F_{Y^*}(y^*[m]))$ (with an additional corresponding coefficient β^*):

$$F_i(x_i[m]) = \mathbb{E}_{\phi(Z_i|\mathbf{z}_{-i}[m],z^*[m];(\boldsymbol{\beta}_{-i},\beta^*),\sigma)}[\Phi(Z_i)], \quad \forall m. \tag{7}$$

We can approximate the expectation in the above using standard tools. Specifically, given a normally distributed variable $Z \sim \mathcal{N}(\mu,\sigma)$ and a smooth function $h(\cdot)$, we use its Taylor's expansion to obtain

$$\mathbb{E}[h(Z)] = \int h(z)\frac{1}{\sqrt{2\pi}}e^{-\frac{(z-\mu)^2}{2\sigma^2}}dz = h(\mu) + h''(\mu')\frac{\sigma^2}{2},$$

where we used $\mathbb{E}[Z-\mu]=0$. In our case $h(z)=\Phi(z)$, and the second derivative is bounded by $h''(z) \le 1/\sqrt{2\pi}$. Moreover, we have $\sigma^2 = 1 - \sum_{j\neq i}\rho_{i,j}^2$, where $\rho_{i,j} = corr(Z_i,Z_j)$ and thus $\sigma^2 \le 1$. This suggests that we approximate $\mathbb{E}[\Phi(X_i)]$ with the zero-order term, $\Phi(\mathbb{E}[X_i])$. Note that when the variables in Equation (7) have greater correlation, σ^2 shrinks and the approximation improves.

Figure 2 shows the quality of approximation: even in the worst case $\sigma^2 = 1$, it is surprisingly accurate. Thus, we can reliably replace Equation (7) with

$$F_i(x_i[m]) = \Phi\left(\mathbb{E}_{\phi(Z_i|\mathbf{z}_{-i}[m],z^*[m];(\boldsymbol{\beta}_{-i},\beta^*),\sigma)}[Z_i]\right), \quad \forall m.$$

From this, using the standard properties of the normal distribution, we obtain

$$F_i(x_i[m]) = \Phi\left(\textstyle\sum_{j\neq i}\beta_j z_j[m] + \beta^* z^*[m]\right), \quad \forall m.$$

The coefficients β are identifiable up to scale and thus (similarly to [Elidan et al., 2007]), we set $\beta^*=1$. Recall that $z_i[m] \equiv \Phi^{-1}(U_i[m]) = \Phi^{-1}(F_i(x_i[m]))$, for all m. Applying Φ^{-1} to both sides and solving w.r.t. z^*, we get

$$z^*[m] = z_i[m] - \textstyle\sum_{j\neq i}\beta_j z_j[m], \quad \forall m. \tag{8}$$

With a little abuse of notation, we refer to $z^*[m]$ as the *Quasi Ideal Parent (QIP)* realizations profile. (formally to get the QIP realizations as defined in 4.3, we need to apply $\mathbf{F}_{Y^*}^{-1} \circ \Phi$ to $z[m]^*$). Note that on the one hand the QIP Equation (8) closely resembles the IP Equation (2). On the other hand, it captures the residuals in the $\{\mathbf{Z}_i\}$ space. As we shall see, this reformulation turns out to be beneficial in practice.

5 Discovering Hidden Variables

We are now ready for our central goal of discovering hidden variables in non-Gaussian domains. We start by describing how QIP can be used to approximate structural modifications in general, and then how this can be used to discover and embed hidden variables.

5.1 Efficient Approximate Scoring

Consider a set $\mathbf{D}$ of M training examples and a copula network model $\mathcal{M}$ where Par_i are parents of X_i. In score-based structure learning, we are interested in evaluating the change of the score of the model as result of structural modification, specifically the addition of a new parent variable W to X_i. Using the BIC score, this involves trivial computation of a penalty term and more difficult evaluation of the log-likelihood function $l(\cdot)$, that requires a computation of the maximum likelihood parameters.

Let $\hat{\boldsymbol{\theta}}$ denote the (already estimated) maximum likelihood parameters before W is added as a new parent of X_i, and let $\boldsymbol{\theta}^+$ denote the parameters (to be estimated) after this structural operation. For brevity, we use the shorthands $F_i[m] \equiv F_i(x_i[m])$, $\mathbf{z}_{\mathrm{Par}_i} \equiv \{\Phi^{-1}(F_j(x_j))\}_{j\in\mathrm{Par}_i}$ and $z_w = \Phi^{-1}(F_W(w))$.

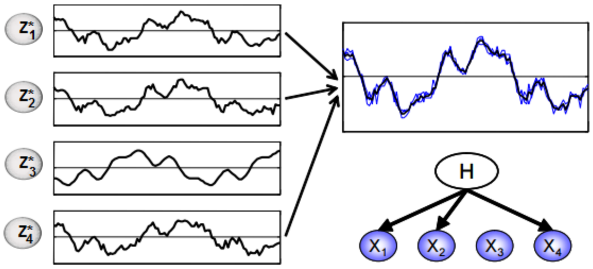


Figure 3: Illustration of how the QIP is used to suggest new hidden variables. Shown on the left are the QIP profiles $Z_1^*, ..., Z_4^*$ of four variables. Recall that these correspond to the residual information of the variables that is not explained by the current model, in the copula domain. Note that the first, second and fourth variables have similar ideal profiles. These profiles are averaged (top right), resulting in a candidate *joint* hidden parent profile for X_1, X_2, X_4 (top right). This variable will be added to the network (bottom right) and its profile will serve a starting point for the parameteric and structural EM that follows.

The change in $l(\cdot)$ that is due to the addition of the edge $W \to X$ can then be calculated as

$$\begin{aligned}
&\Delta_{X_i|\mathrm{Par}_i}(W) \\
&\quad\equiv \max_{\theta^+} l_X(\mathbf{D}:\mathrm{Par}_i \cup \{W\}, \theta^+) - l_X(\mathbf{D}:\mathrm{Par}_i, \hat{\boldsymbol{\theta}}) \\
&\quad= \max_{\boldsymbol{\theta}^+} \textstyle\sum_m \log f(x_i[m]|\mathrm{Par}_i[m], w[m]; \boldsymbol{\theta}^+) \\
&\qquad\quad - \textstyle\sum_m \log f(x_i[m]|\mathrm{Par}_i[m]; \hat{\boldsymbol{\theta}}) \\
&\quad= \max_{\boldsymbol{\theta}^+} \textstyle\sum_m \log c(F_i[m]|\{F_j[m]\}_{j\in\mathrm{Par}_i}, F_W[m]; \boldsymbol{\theta}^+) \\
&\qquad\quad - \textstyle\sum_m \log c(F_i[m]|\{F_j[m]\}_{j\in\mathrm{Par}_i}; \hat{\boldsymbol{\theta}}) \\
&\quad= \max_{\boldsymbol{\beta}^+, \sigma^+} \textstyle\sum_m \phi\left(z_i[m]|\mathbf{z}_{\mathrm{Par}_i}[m], z_w[m]; \boldsymbol{\beta}^+, \sigma^+\right) \\
&\qquad\quad - \textstyle\sum_m \phi\left(z_i[m]|\mathbf{z}_{\mathrm{Par}_i}[m]; \hat{\boldsymbol{\beta}}, \hat{\sigma}\right), \qquad (9)
\end{aligned}$$

where the last line follows from the reparametrization discussed in the previous section. The terms not depending on β^+ and σ are intentionally separated and will cancel out with other terms later.

Ideally, we would like to evaluate $\Delta_{X_i|\mathrm{Par}_i}(W)$ exactly but this can be prohibitive since we need to estimate β^+ and σ^+ for each candidate parent W. Instead, similarly to the original IP, we only estimate the scale parameter associated with Z_w, while keeping all the other parameters fixed to their values before W was added. Using the QIP profile developed in Section 4.2, this approximation can be computed efficiently:

Lemma 5.1. : *Let X_i, Par_i and W be as above, and $\mathbf{z}^* = (z^*[1], \ldots, z^*[m])$ be a QIP profile of X. Denote the scaling parameter associated with $\mathbf{z}_w = (z_w[1], \ldots, z_w[m])$ by β_w^+ and let $\widetilde{\Delta}_{X_i|\mathrm{Par}_i(W)}$ be similar to $\Delta_{X_i|\mathrm{Par}_i}(W)$ but maximized only over β_w^+. Then*

$$\widetilde{\Delta}_{X_i|\mathrm{Par}_i}(W) = \frac{1}{2\hat{\sigma}^2}\frac{(\mathbf{z}^* \cdot \mathbf{z}_w)^2}{\|\mathbf{z}_w\|^2} = \frac{M}{2}\cos^2(\angle(\mathbf{z}^*, \mathbf{z}_w)), \qquad (10)$$

where $(\mathbf{z}^* \cdot \mathbf{z}_w)$ is the standard inner product, and $\angle(\mathbf{z}^*, \mathbf{z}_w)$ is the angle between $\mathbf{z}^*$ and $\mathbf{z}_w$.

Thus, we can use the QIP to efficiently approximate the merit of adding a new parent to the model. Note that this approximation, $\widetilde{\Delta}_{X_i|\mathrm{Par}_i}(W)$, is an asymptotic lower-bound of the objective $\Delta_{X_i|\mathrm{Par}_i}(W)$:

Corollary 5.2.:

$$\widetilde{\Delta}_{X_i|\mathrm{Par}_i}(W) \leq \Delta_{X_i|\mathrm{Par}_i}(W), \quad M \to \infty, \quad a.s. \qquad (11)$$

Proofs of the above lemma and corollary can be found in the supplementary material.

5.2 Adding New Hidden Variables

We are now ready for our core goal: discovering non-Gaussian hidden variables. Recall that the QIP approximates the ideal predictor of a variable. Intuitively, similarly to the original IP, if several variables have a similar QIP, they could benefit from a joint predictor. The high level idea is illustrated in Figure 3 and the technical details are discussed next.

We start by approximating the change to the log-likelihood when a latent H is added as a *joint* parent to $X_1, \ldots, X_L$. Denote by $\mathbf{z}_i^*$ the QIP profile of X_i, let $\mathbf{h}$ be the *unobserved* realizations of H and $\mathbf{z_H} \equiv \Phi^{-1}(\mathbf{h})$. Using Lemma 5.1, the gain is approximated by

$$\widetilde{\Delta}_{X_{1,\ldots,L}}(H) \equiv \textstyle\sum_{i=1}^{L} \widetilde{\Delta}_{X_i|\mathrm{Par}_i}(H) = \sum_{i=1}^{L} \frac{1}{2\hat{\sigma_i}^2}\frac{(\mathbf{z}_i^* \cdot \mathbf{z}_H)^2}{\|\mathbf{z}_H\|^2},$$

where $\hat{\sigma_i}$ is the variance estimate *before* the addition of H (note that the change to the likelihood also includes a term associated with H as a root and a complexity term but these are easy to compute).

Our goal is to find H that is most beneficial as a predictor to all of its children $X_1, \ldots, X_L$. Thus, we need to compute the following:

$$\mathbf{z}_H = \mathrm{argmax}_{\mathbf{z}} \widetilde{\Delta}_{X_1,\ldots,X_L}(H)$$

While a seemingly complex optimization problem, similarly to the IP, this can actually be solved using the eigen vector problem $\left(\gamma^T\gamma\right)\mathbf{z}_H = \lambda\mathbf{z}_H$, where γ is a matrix whose columns are $\mathbf{z}_i^*/\hat{\sigma}_i$ and λ is the largest eigenvalue associated with $\gamma^T\gamma$ (see [Elidan et al., 2007] for details).

Now that we can approximate the gain of adding a new hidden parent to a cluster of variables, we still find the most beneficial cluster. Since the number of clusters is exponential, we follow an agglomerative clustering approach to explore different clusters [Duda et al., 1973].

Figure 4: Comparison of the log-likelihood and complexity of the models learned from synthetic data using our copula-based **QIP** approach, the original Gaussian **IP**, the **Full** copula network with a hidden parent for all variables, and a **Tree** copula network over the observables. Shown are mean and deviation over 10 random train/test partitions. Univariate marginals are Gaussian (left) and exponential (right).

	Edges	Vars	Train	Test
Tree	69 (0)	70 (0)	-93.15 (2.25)	-94.72 (2.1)
IP	182.37 (4.71)	78.72 (2.47)	-85.6 (2.4)	-86.34 (2.25)
Full	328.19 (5.73)	74.4 (2.57)	-98.25 (3.1)	-98.6 (3.79)
QIP	186.15 (5.35)	76.86 (2.64)	-85.24 (2.21)	-85.73 (1.84)

	Edges	Vars	Train	Test
Tree	69 (0)	70 (0)	-70.53 (3.81)	-71.41 (3.63)
IP	206.74 (4.91)	84.52 (2.36)	-86.24 (3.86)	-87.16 (2.94)
Full	323.8 (5.65)	75.42 (2.44)	-94.85 (2.78)	-96.38 (3.62)
QIP	184.77 (5.31)	78.49 (2.52)	-64.72 (4.21)	-66.59 (3.85)

Figure 5: Comparision of our QIP method to LTC [Kirshner, 2012] for the S&P100 datasets (left), and CVO [Chandrasekaran et al., 2010] for the Dow dataset (right). Shown is the mean test log-probability/instance over 10 folds, and the number of hidden variables learned.

	Test LL	Hiddens
QIP	107.63	12
LTC	108.38	83

SP100

	Test LL	Hiddens
QIP	-14.08	5
CVO	-22.02	29

Dow

Starting with each variable as an individual cluster and repeatedly merging two clusters that lead to the best *expected* improvement in the BIC score. A new hidden variable H is then introduced as a parent of each of the resulting cluster variables.

Specifically, let $\mathbf{D}_H$ denote the unobserved part of the data over the hidden variables, $\mathbf{D}_O$ be the observed data, and Q be the distribution represented by the current network. We use the posterior distribution $Q(\mathbf{D}_H|\mathbf{D}_O)$ to estimate the expected BIC score

$$E_Q(BIC(\mathbf{D};\mathcal{G})|\mathbf{D}_O) = \int Q(\mathbf{D}_H|\mathbf{D}_O)BIC(\mathbf{D};\mathcal{G})d\mathbf{D}_H.$$

where the network parameters are estimated using a standard Expectation Maximization (EM) approach [Dempster et al., 1977]. As a starting point of the EM procedure, we use our *pseudo* observations $\mathbf{z}_H$.

6 Experiments

We now evaluate the merit of our QIP approach for discovering hidden variables using Gaussian copula Bayesian networks, both in synthetic and real-life scenarios. Although our approach is applicable to general structures, for concreteness, we learn bipartite networks where hidden variables in the top layer are parents of observed children in the lower level. These networks allow each observed variable to have many parents and contain many loops and can thus be quite expressive. Indeed, some popular large-scale networks use this representation, among them are the QMR system and BN2O [Koller and Friedman, 2009].

In all experiments we use the standard greedy score-based structure learning. At each iteration a new hidden variable is introduced as a parent of a subset of variables. The difference between the methods is in how this subset is chosen, and how the hidden variable is constructed. The structure is then adapted using a standard structural EM procedure that allows for addition/removal of edges [Friedman, 1998].

6.1 Synthetic Evaluation

We construct random two-layer Gaussian copula networks with 7 hidden and 70 observed variables as follows. Edges connect observed and hidden nodes randomly, allowing up to 3 parents. Correlation parameters of the local densities are sampled in the interval [0.4 0.8], allowing for a wide range of dependence strengths. We generate 2000 samples from the network and split them into train/test sets. All results are reported over 10 random splits.

We compare our Gaussian copula **QIP** approach to a Gaussian network using the original **IP** as well as two additional copula baselines: a **Full** model, where at each iteration a hidden variable is added as a parent of all observables (also followed by structural adaptation), and a **Tree** network over the observed variables.

Figure 4 (left panel) shows the log-probability and complexity of the learned models when the univariate marginals are Gaussian. In this case the Gaussian copula is just a standard multivariate Gaussian and, as expected, **QIP** and **IP** are very similar. The **Full** approach is substantially inferior, demonstrating the need for *informed* discovery of hidden variables. Also clear is the advantage over **Tree**, highlighting the power coming from the discovery of hidden variables. Note that the scale of log-probability is in bit/instance so that the advantage of k translates to each instance being, on average, 2^k times more likely.

Figure 4 (right panel) compares the different models when the univariate marginals are exponential with

Figure 6: Performance of our copula-based **QIP** approach, the original Gaussian **IP**, and the maximum likelihood copula **Tree** over the observed variables for several real datasets. Shown is the mean and standard deviation log-probability/instance over 10 random train-test splits.

	Dow (29 vars)		Crime (100)		Music (68 vars)		SP (500 vars)	
	Train	Test	Train	Test	Train	Test	Train	Test
Tree	-15.0	-15.4	73.6	74.5	-68.3	-69.3	-426.2	-427.8
	(0.32)	(0.45)	(1.78)	(2.12)	(0.82)	(1.05)	(2.73)	(2.25)
IP	-18.0	-18.7	52.7	54.3	-81.4	-82.7	-526.7	-529.2
	(0.26)	(0.34)	(1.72)	(2.16)	(1.52)	(1.83)	(2.78)	(2.97)
QIP	-13.7	-14.0	77.4	78.4	-61.7	-62.4	-414.8	-416.9
	(0.24)	(0.19)	(2.63)	(2.31)	(1.33)	(1.42)	(2.35)	(2.46)

$\lambda = 1$. Appealingly, even with this simple marginal representation, the gap between **QIP** and **IP** is substantial, emphasizing the importance of allowing for non-Gaussian representations.

6.2 Real-World Domains

We now evaluate the ability of our QIP to discover effective hidden variables in real-life domains, starting with a comparison to current state-of-the-art baselines.

We compare to the **LTC** approach [Kirshner, 2012] using the reported 8-fold cross-validation protocol on the S&P100 data set: 85 *monthly* stock returns over the years years 1990 - 2007. Results are shown in Figure 5 (LTC numbers are from [Kirshner, 2012]). The difference between the two approaches is within a standard deviation and is, thus, not significant. However, while **LTC** uses 83 hidden variables on average, our **QIP** approach is able to compete with an average of just 12 hidden variables. This difference has scalability ramification when bigger network are considered e.g. such as S&P 500 that is considered below.

Next, we compare to the convex optimization (**CVO**) approach introduced in [Chandrasekaran et al., 2010]. Due to the prohibitive computational demands of this method, we consider the more modest Dow domain with 29 variables. We use the standard CVX tool [Grant and Boyd, 2013, 2008] for the computations. Figure 5 (right) shows the results using 8-fold cross-validation. To give **CVO** a fighting chance, we search for the best fit for its parameters $\lambda \in \sqrt{\frac{p}{M}} \times [1, 100]$ and $\gamma \in [1, 20]$. The superiority of **QIP** is evident. This should not come as a surprise since **CVO** assumes a Gaussian model, and we expect its performance to be close to that of the the original **IP** (see Table 6).

Finally, we compare **QIP** to the standard **IP** as well as a Gaussian copula **Tree** baseline on several real-life domains, some of which are significantly bigger than those explored in the literature: **Dow**. End of day changes of the 29 Dow-Jones stocks for 2000 trading days; **Music** (UCI repository). 68 audio features correspond to 1059 tracks; **Crime** (UCI repository). 100 variables relating to crime in the U.S. for 1994 samples; **SP500**. *Daily* returns of 500 Standard and Poor's index stocks for 2000 trading days.

Results are summarized in Figure 6, where the improvement over **IP** demonstrates the importance of using a non-Gaussian representation. Similarly, the substantial advantage over **Tree** highlights the advantage of discovering hidden variables.

Interestingly, despite of the fact that the notion of consistency does not hold for **QIP**, the models learned are semantically appealing. For example, when running **QIP** over the **SP500** data set, in all random folds, a new parent variable governing a cluster of gas and oil producers was created. Additionally, another cluster of health care providers was created. Similar appealing qualitative clusters are also evident for the other datasets (not shown for lack of space).

7 Discussion and Future Work

In this work we tackled the challenge of *discovering* novel hidden variables in non-Gaussian domains based on the Gaussian copula representation. We introduced QIP, a generalization of the concept of an ideal predictor, and provided the machinery needed to make this concept useful for effective discovery of hidden variables. We demonstrated the advantages of our approach in synthetic and real-life settings.

To the best of our knowledge, only two previous works consider the task of discovering hidden variables in real-valued non-Gaussian high dimensional domains. [Elidan et al., 2007], which we generalize, is limited to simple parametric forms and requires specific tailoring for each one. [Kirshner, 2012] is potentially more flexible in terms of the local copula representation but is limited to a tree structure. As we showed, this results in overly complex models and is inherently less scalable than our approach. Indeed, some of the domains we consider are substantially large than those previously considered in the literature.

An important future derection is the extension of QIP formulation to additional copula families. The goal is to discover effective hidden variables in domains where the dependence structure follow elaborate patterns, such as heavy tail dependence.

References

M. Bilodeau and D. Brenner. Theory of multivariate statistics. *Springer Science & Business Media*, 1999.

V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky. Latent variable graphical model selection via convex optimization. *Annual Allerton Conference on Communication, Control and Computing*, 2010.

T. Chen, N. L. Zhang, and Y. Wang. Efficient model evaluation in the search-based approach to latent structure discovery. *European Workshop on Probabilistic Graphical Models*, 2008.

M. J. Choi, V. Y. Tan, A. Anandkumar, and A. S. Willsky. Learning latent tree graphical models. *The Journal of Machine Learning Research*, 12:1771–1812, 2011.

C. Daskalakis, E. Mossel, and S. Roch. Optimal phylogenetic reconstruction. *Annual ACM Symposium on Theory of Computing*, 2006.

J. Dauwels, H. Yu, S. Xu, and X. Wang. Copula Gaussian graphical model for discrete data. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. 1977.

R. O. Duda, P. E. Hart, et al. Pattern classification and scene analysis. *Wiley New York*, 3, 1973.

G. Elidan. Copula Bayesian networks. *Advances in Neural Information Processing Systems*, pages 559–567, 2010.

G. Elidan and N. Friedman. Learning hidden variable networks: The information bottleneck approach. *Journal of Machine Learning Research*, pages 81–127, 2005.

G. Elidan, N. Lotner, N. Friedman, D. Koller, et al. Discovering hidden variables: A structure-based approach. *Advances in Neural Information Processing Systems*, 2000.

G. Elidan, I. Nachman, and N. Friedman. Ideal Parent structure learning for continuous variable Bayesian networks. *Journal of Machine Learning Research*, 8:1799–1833, 2007.

P. Embrechts, F. Lindskog, and A. McNeil. Modelling dependence with copulas and applications to risk management. *Handbook of heavy tailed distributions in finance*, pages 329–384, 2003.

N. Friedman. The Bayesian structural EM algorithm. *Conference on Uncertainty in Artificial Intelligence*, 1998.

M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.0 beta. http://cvxr.com/cvx, September 2013.

P. O. Hoyer, S. Shimizu, A. J. Kerminen, and M. Palviainen. Estimation of causal effects using linear non-Gaussian causal models with hidden variables. *International Journal of Approximate Reasoning*, 2008.

D. Janzing, J. Peters, J. Mooij, and B. Schölkopf. Identifying confounders using additive noise models. *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 249–257, 2009.

H. Joe. Multivariate models and multivariate dependence concepts. *CRC Press*, 1997.

H. Joe. Dependence modeling with copulas. *CRC Press*, 2014.

S. Kirshner. Learning with tree-averaged densities and distributions. *Advances in Neural Information Processing Systems*, 2007.

S. Kirshner. Latent tree copulas. *European Workshop on Probabilistic Graphical Models*, 2012.

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

P. F. Lazarsfeld, N. W. Henry, and T. W. Anderson. Latent structure analysis. *Houghton Mifflin Boston*, 1968.

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

J. Pearl. Probabilistic reasoning in intelligent systems: networks of plausible inference. *Morgan Kaufmann*, 1988.

M. Rey and V. Roth. Copula mixture model for dependency-seeking clustering. *International Conference on Machine Learning*, 2012.

M. Sklar. Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut de Statistique de L'Universite de Paris*, 8:229–231, 1959.

C. Spearman. "General intelligence," objectively determined and measured. *The American Journal of Psychology*, 15:201–292, 1904.

Y. Tenzer and G. Elidan. Speedy model selection (SMS) for copula models. *Conference on Uncertainty in Artificial Intelligence*, 2013.

N. L. Zhang. Hierarchical latent class models for cluster analysis. *Journal of Machine Learning Research*, 5:697–723, 2004.

N. L. Zhang and T. Kocka. Efficient learning of hierarchical latent class models. *IEEE International Conference on Tools with Artificial Intelligence*, 2004.