Copulas in Machine Learning

Gal Elidan

Abstract Despite overlapping goals of multivariate modeling and dependence identification, until recently the fields of machine learning in general and probabilistic graphical models in particular have been ignorant of the framework of copulas. At the same time, complementing strengths of the two fields suggest the great fruitfulness of a synergy. The purpose of this paper is to survey recent copula-based constructions in the field of machine learning, so as to provide a stepping stone for those interested in further exploring this emerging symbiotic research.

1 Introduction

Multivariate modeling is of fundamental interest in diverse complex domains ranging from computational biology to computer vision to astronomy. Unfortunately, high-dimensional modeling in the context of finite data and limited computational resources can be quite challenging, and susceptible to the curse of dimensionality. Probabilistic graphical models [33], a marriage between probability and graph theory, is a general purpose framework aimed at coping with this task. These models are used to represent multivariate densities via a combination of a qualitative graph structure that encodes independencies and local quantitative parameters. The joint density has a decomposable form that corresponds to the intuitive graph structure. This, in turn, allows for relatively efficient methods for marginal and posterior computations (a task called *inference* in the field), estimation (parameter learning), and model selection (structure learning). Probabilistic graphical models have become a central axis of the field of machine learning, have made substantial impact in related fields such as machine vision, natural language processing and bioinformatics, and have become prevalent in uncountable applications.

Gal Elidan
Department of Statistics, Hebrew University, Jerusalem, Israel e-mail: galel@huji.ac.il

It is somewhat remarkable that, until recently, researchers in the field of probabilistic graphical models were largely unaware of the multivariate modeling framework of copulas. This ignorance is even more perplexing when considering the limitations of graphical models in the context of real-valued measurements: while probabilistic graphical models are conceptually general, practical considerations almost always force the local quantitative part of the model to be of a simple form. In fact, when faced with data that cannot be captured well with multivariate Gaussians or mixtures thereof, the vast majority of works first discretize the data, and then take advantage of the impressive progress that has been made in the discrete case.

Much of the copula community has also been ignorant of the potential of a symbiosis with the field of machine learning. A decade ago, Kurowicka and Cooke [23] identified a relationship between vine models and Bayesian networks (a directed graphical model), and this was later generalized [24, 16] to yield high-dimensional copula constructions. However, no algorithmic innovation was borrowed from or inspired by machine learning, with the goal of, for example, automatically inferring the structure of such models from partially observed data.

There are fundamental reasons as to why a symbiosis between the two fields should be pursued. Graphical models are inherently aimed at high-dimensional domains, and substantial advances have been made in learning such models from data. Unfortunately, in real-valued scenarios the field is still largely handicapped. In contrast, copulas offer a flexible mechanism for modeling real-valued distributions. Yet, much of the field is still focused on the bivariate case or is limited in practice to few variables (exceptions are discussed later). The two frameworks thus complement each other in a way that offers opportunities for fruitful synergic innovations.

The need for a synergy between the copula framework and the field of machine learning goes further than probabilistic graphical models. Dependence measures, most notably Shannon's mutual information, are fundamental to numerous machine learning algorithms such as clustering, features selection, structure learning, causality detection and more. As is well known, copulas are closely tied to such dependence concepts and the meeting of the two fields can give rise to new techniques for measuring dependance in high dimension.

It was only recently that the ignorance barrier between the two fields was broken by Kirshner's work [21] that generalizes Darsow's Markovian operator [7] for tree structured models. Since then, interest in copulas has been steadily growing and the last years have seen a range of innovative copula-based constructions in machine learning. The purpose of this paper is to survey these works. Rather than aiming at a complete coverage, the focus is on multivariate constructions as well as information estimation. For lack of space, additional works that, generally speaking, use copulas in a more plug-in manner, are not discussed. For the interested reader, these include copula-based independent component analysis [35], component analysis [27, 2], mixture models (e.g., [14, 51]), dependency seeking clustering [40]. Also of great interest but not presented here is the use of copulas as a particular instance within the cumulative distribution network model [17, 45]. Finally, this survey does not cover application papers or works that appeared in the computational statistics community, and that are more likely to be familiar to copula researchers.

2 Background

To allow for reasonable accessibility to both copula and machine learning researchers, in this section we briefly review the necessary background material from both fields and set a common notation. We use capital letters X,Y to denote random variables, lower case letters x,y to denote realisations of these variables, bold-faced letters to refer to set of variables $\mathbf{X}$ and their assignments $\mathbf{x}$.

2.1 Copulas

A copula function [48] links univariate marginal distributions to form a joint multivariate one. Formally,

Definition 2.1. Let $U_1,\dots,U_n$ be real random variables marginally uniformly distributed on $[0,1]$. A copula function $C:[0,1]^n \to [0,1]$ is a joint distribution

$$C(u_1,\dots,u_n) = P(U_1 \le u_1,\dots,U_n \le u_n).$$

We will use $C_\theta(\cdot)$ to denote a parameterized copula function where needed.

Sklar's seminal theorem [48] states that *any* joint distribution $F_{\mathbf{X}}(\mathbf{x})$ can be represented as a copula function $C(\cdot)$ of its univariate marginal distributions

$$F_{\mathbf{X}}(x_1,\dots,x_n) = C(F_1(x_1),\dots,F_n(x_n)).$$

When the marginals are continuous, $C(\cdot)$ is uniquely defined. The constructive converse, which is of interest from a modeling perspective, is also true: *any* copula function taking *any* univariate marginal distributions $\{F_i(x_i)\}$ as its arguments, defines a valid joint distribution with marginals $\{F_i(x_i)\}$. Thus, copulas are "distribution generating" functions that allow us to separate the choice of the univariate marginals and that of the dependence structure, encoded in the copula function $C(\cdot)$. Importantly, this flexibility often results in a construction that is beneficial in practice.

Assuming $C(\cdot)$ has n'th order partial derivatives, the joint density can be derived from the copula function using the derivative chain rule

$$f_{\mathbf{X}}(\mathbf{x}) = \frac{\partial^n C(F_1(x_1),\dots,F_n(x_n))}{\partial F_1(x_1)\dots\partial F_n(x_n)} \prod_i f_i(x_i) \equiv c(F_1(x_1),\dots,F_n(x_n)) \prod_i f_i(x_i), \quad (1)$$

where $c_\theta(\cdot)$ is called the *copula density*.

Example 2.1. Perhaps the most commonly used is the Gaussian copula [11]:

$$C_\Sigma(\{F_i(x_i)\}) = \Phi_\Sigma\big(\Phi^{-1}(F_1(x_1)),\dots,\Phi^{-1}(F_n(x_n))\big)\,, \quad (2)$$

where Φ is the standard normal distribution, and Φ_Σ is a zero mean normal distribution with correlation matrix Σ.

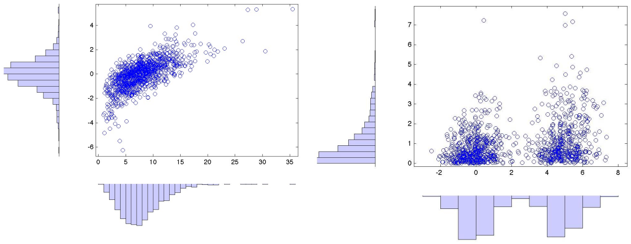

Fig. 1 Samples from the bivariate Gaussian copula with correlation $\theta = 0.25$. (left) with unit variance Gaussian and Gamma marginals; (right) with a mixture of Gaussian and exponential marginals.

Figure 1 shows samples from the bivariate Gaussian copula using two different marginal settings. As can be seen, even in this simple case, markedly different and multi-modal distributions can be constructed. More generally, and without any added computational difficulty, we can use different marginals for each variable, and can also mix and match marginals of different forms with *any* copula function.

2.2 Probabilistic Graphical Models

In this section we briefly review probabilistic graphical models [33], a widely popular framework for representing and reasoning about high-dimensional densities.

A *directed graph* is a set of nodes connected by directed edges. A *directed acyclic graph* (DAG) $\mathcal{G}$ is a directed graph with no directed cycle. The *parents* of a node V in a directed graph is the set of all nodes U such that there exists a direct edge from U to V. A node U is an ancestor V in the graph if there is a directed path from U to V. Children and descendant are similarly defined.

Directed graphical models or *Bayesian networks* (BNs), use a DAG $\mathcal{G}$ whose nodes correspond to the random variables of interest $X_1, \ldots, X_n$ to encode the independencies $I(\mathcal{G}) = \{(X_i \perp \mathbf{ND}_i \mid \mathbf{Pa}_i)\}$, where $\perp$ denotes the independence relationship, and $\mathbf{ND}_i$ are nodes that are not descendants of X_i in $\mathcal{G}$ (independencies that follow from $I(\mathcal{G})$ are easily identifiable via an efficient algorithm). If the independencies encoded by $\mathcal{G}$ hold in $f_{\mathbf{X}}$, then it is easy to show that the joint density decomposes into a product of local conditional densities

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{i=1}^{n} f_{X_i|\mathbf{Pa}_i}(x_i \mid \mathbf{pa}_i),$$

where $\mathbf{Pa}_i$ are the parents of node X_i in $\mathcal{G}$. The converse composition theorem states that a product of *any* local conditional densities defines a valid joint density, and that the independencies encoded by $\mathcal{G}$ hold in this density.

As an example, Figure 2 shows a plausible model that involves relocation of human population into Mars. Human pollution is unfortunately assumed independent of Solar activity. Yet, these two factors are dependent given evidence of livability conditions on earth. Similar deductions all follows from the independencies encoded

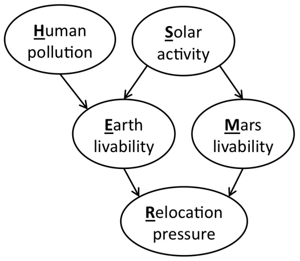

Fig. 2 An toy Bayesian network of a Mars relocation scenario where $f(\cdot) = f(H)f(S)f(E|S,H)f(M|S)f(R|E,M)$.

in the graph. Note that inferences can be made in any direction, regardless of the direction of the edges, hence the name *Bayesian* networks.

Undirected graphical models, or *Markov Networks* (MNs), use an undirected graph $\mathscr{H}$ that encodes the independencies $I(\mathscr{H}) = \{(X_i \perp \mathbf{X} \setminus \{X_i\} \cup \mathbf{Ne}_i \mid \mathbf{Ne}_i)\}$, where $\mathbf{Ne}_i$ are the neighbors of X_i in $\mathscr{H}$. That is, each node is independent of all others given its neighbors in $\mathscr{H}$. Let C be the set of cliques in $\mathscr{H}$ (a clique is set of nodes such that each node is connected to all others in the set). As for directed models, the Hammersley-Clifford theorem [15] states that, for positive densities, if the independence statements encoded by $\mathscr{H}$ hold in $f_{\mathbf{X}}(\mathbf{x})$, then the joint density decomposes according to the graph structure:

$$f_{\mathbf{X}}(\mathbf{x}) = \tfrac{1}{Z} \prod_{c \in \mathscr{C}} \phi_c(\mathbf{x}_c), \tag{3}$$

where $\mathbf{X}_c$ are the set of nodes in the clique c, and $\phi_c : \mathbb{R}^{|c|} \to \mathbb{R}^+$ is any positive function over the values of these nodes. Z is a normalizing constant called the partition function. The converse composition theorem also holds.

There are various generalization of the Bayesian and Markov network representations (which overlap only for tree structured models) including temporal, relational and mixed directionality models (chain graphs). The common theme is that of decomposition into local terms which, in additional to facilitating compact representation, gives rises to relatively efficient marginal and conditional computations (a task called *inference* in the ML community), estimation (parameter learning), and high-dimensional model selection (structure learning). See [33, 22] for a comprehensive presentation of probabilistic graphical models.

3 Multivariate Copula-based Construction

In this section we present several high-dimensional copula-based models recently developed in the machine learning community. As is common in the copula community [20], these works generally start with univariate estimation, and then plug in the "given" marginals into the copula function. Thus, except where essential, our exposition below does not cover the relatively straightforward and standard univariate estimation step. Instead, we focus on the multivariate construction. We end with a comparative summary in Section 3.5, which can also be read first.

3.1 Tree Structured Models

The first work in the machine learning community to combine ideas from the graphical models framework and copulas is that of Kirshner [21] (the earlier work of [24] independently developed in the copula community is discussed in Section 3.3). We start by describing the basic tree-structured copula construction, and then present the tree-averaged density model. We conclude this section with a flexible Bayesian approach to a mixture of copula trees suggested by Silva and Gramacy [47].

Tree-structured Copulas

Let T be an undirected tree structured graph (i.e., a graph with no cycles) and let $\mathscr{E}$ denote the set of edges in T that connect two vertices. From the Hammersley-Clifford decomposition of (3), it easily follows that, if the independencies $I(T)$ hold in $f_{\mathbf{X}}(\mathbf{x})$, then it can be written as

$$f_{\mathbf{X}}(\mathbf{x}) = \left[\prod_i f_i(x_i)\right] \prod_{(i,j)\in\mathscr{E}} \frac{f_{ij}(x_i,x_j)}{f_i(x_i)f_j(x_j)}.$$

Using (1), a decomposition of the joint copula also follows

$$c_T(\cdot) = \frac{f_{\mathbf{X}}(\mathbf{x})}{\prod_i f_i(x_i)} = \prod_{(i,j)\in\mathscr{E}} \frac{f_{ij}(x_i,x_j)}{f_i(x_i)f_j(x_j)} = \prod_{(i,j)\in\mathscr{E}} c_{ij}(F_i(x_i),F_j(x_j)), \tag{4}$$

where $c_T(\cdot)$ is used to denote a copula density that corresponds to the structure T, and $c_{ij}(\cdot)$ is used to denote the bivariate copula corresponding to the edge (i,j). The converse composition also holds: a product of local bivariate copula densities, each associated with an edge of T, defines a valid copula density. This result generalizes Darsow's operator [7] to the case of Markov trees. Indeed, it can be proved directly or by an inductive application of Darsow's product operator starting from the leaves of the trees and progressing inwards.

The main appeal of the above decomposition, as is the case for graphical models in general, is that estimation or learning also benefits from the compact representation. Given univariate marginals, (4) leads to a decomposition of the log-likelihood into independent terms, and estimation can be carried out by only considering bivariate statistics. This is in contrast to vine copula models [3] that also involve bivariate copulas but where (conditional) statistics over large sets of variables are required (see Section 3.5 for further discussion).

Tree-averaged Copulas

As noted, the main appeal of the tree-structured copula is that it relies solely on bivariate estimation. However, this comes at the cost of firm independence assumptions. To relax these, Kirshner suggests the construction of a mixture of all copula trees model. On the surface, such a model may appear to be computationally prohibitive as the number of possible trees with n variables is n^{n-2}.

This difficulty is overcome by defining an appropriate decomposable prior over all spanning trees suggested by Meila and Jaakkola [28]. Let β be a symmetric $n \times n$ matrix with non-negative entries and zero on the diagonal. Let $\mathscr{T}$ be the set of all spanning trees over $X_1, \ldots, X_n$. The probability of a spanning tree T is defined as

$$P(T \in \mathscr{T} \mid \beta) = \tfrac{1}{Z} \textstyle\prod_{(u,v) \in \mathscr{E}_T} \beta_{uv},$$

where Z is a normalization constant. Using a generalization of the Laplacian matrix:

$$L_{uv}(\beta) = \begin{cases} -\beta_{uv} & u \neq v \\ \sum_w \beta_{uw} & u = v, \end{cases}$$

it can be shown that the normalization constant Z is equal to the determinant $|L^*(\beta)|$, where $L^*(\beta)$ represents the first $(n-1)$ rows and columns of $L(\beta)$. This result can then be used to efficiently compute the density of the average of *all* copula spanning trees, which itself is also a copula density:

$$\sum_{T \in \mathscr{T}} P(T \mid \beta) c_T(\cdot) = \frac{1}{Z} \sum_{T \in \mathscr{T}} \left[\prod_{(u,v) \in \mathscr{E}_T} \beta_{uv} c_{uv}(F_u(x_u), F_v(x_v)) \right] = \frac{|L^*(\beta \circ c_T(\cdot))|}{|L^*(\beta)|},$$

where $\circ$ denotes an element-wise product. The reader is referred to Kirshner [21] for additional details on the efficient EM method used for parameter estimation of the model, and for appealing results of modeling multi-site precipitation data using an HMM-based construction.

Bayesian Mixtures of Copula Trees

The all tree mixture model described in the previous section overcomes some of the limitations imposed by a single tree model. However, to facilitate computational efficiency, the prior used involves heavy parameter sharing. Specifically, the set of all n^{n-2} trees is parameterized by only $n(n-1)$ parameters. Further, the approach relies on the assumption that there are no missing observations.

To offer more flexibility, Silva and Gramacy [46] suggest a Bayesian approach that allows for a mixture of *some* trees with flexible priors on all components of the model. The construction is based on the Bayesian nonparametric Dirichlet process infinite mixture model. This model, first formalized by Ferguson [12], is a distribution over discrete mixtures such that for every finite set of mixtures, its parameters have a Dirichlet prior. Following Silva and Gramacy, we present the model here as the limit as $K \rightarrow \infty$ of a finite mixture model with K components.

Let $\mathbf{X}$ be a set of random variables, z be an index of the set of all trees $\mathscr{T}$ over these variables, and Θ be the set of copula parameters, one for each pair of variables. The following model is a standard Bayesian mixture model, with the novelty that the parameters of the univariate marginals Λ are shared by all mixture components:

$$\begin{aligned} \Lambda &\sim f_\Lambda & T_z &\sim T_0(z) \\ \pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) & \Theta_z &\sim f_\Theta \\ z \mid \pi &\sim \text{Discrete}(\pi_1, \ldots, \pi_K) & \mathbf{X} | z, \mathscr{T}, \Theta, \Lambda &\sim f(X \mid T_z, \Theta_z, \Lambda). \end{aligned}$$

The first two lines on the left correspond to any general generating mechanisms for the univariate marginal parameters Λ and the mixture prior parameters π. Given these, a specific tree is selected by sampling z from any discrete distribution of the appropriate dimension parameterized by π. The parameters corresponding to the tree edges Θ_z are then sampled from a prior on the copula parameters. Finally, given a specific tree and previously sampled parameters, the density for a sample $f(X \mid T_z, \Theta_z, \Lambda)$ is constructed using a copula tree, as described in previous section.

Obviously, the above model offers great flexibility and using a Dirichlet process formulation where $K \rightarrow \infty$, allows for a variable number of components. The flexibility comes with a computational burden which is the central challenge addressed by Silva and Gramacy using a Markov chain Monte Carlo approach. The central difficulty is in the sampling of trees since, given a specific tree, most parameters are redundant and sampling these naively will lead to useless computations in later iterations. The solution is a novel proposal distribution from which trees *and* parameters are sampled in a sensible way. The reader is referred to Silva and Gramacy [46] for the precise details. Experiments are carried out on several datasets from the UCI repository [31], as well as missing data scenarios using financial data.

3.2 Undirected Structure Learning

The *lasso* method of Tibshirani [52] extends linear regression to the high-dimensional case by including in the objective function an L1 norm sparsity constraint on the feature coefficients, and proposing an efficient method for optimizing this objective. A nonparametric extension, called sparse additive models was recently developed by Ravikumar et al. [38]. Orthogonally, the *graphical lasso* (glasso) [13] employs similar sparsity constraints to facilitate high-dimensional estimation of undirected Gaussian graphical models. In this section we present the work of Liu et al. [26] that fills the void of high-dimensional nonparametric structure estimation. Specifically, a theoretically founded structure estimator is developed based on the combination of the Gaussian copula and a specific form of nonparametric univariate marginals.

Parametric Undirected Graph Estimation

Let $\mathscr{H}$ be an undirected graph whose nodes correspond to real-valued random variables $X_1, \ldots, X_n$. For multivariate Gaussian distributions, the independencies between the random variables as encoded by the graph's structure are characterized by the inverse covariance matrix $\Omega = \Sigma^{-1}$. Specifically, X_i is independent of X_j given all other variables, denoted by $X_i \perp X_j \mid \mathbf{X}_{\setminus\{i,j\}}$ if and only if $\Sigma^{-1}_{ij} = 0$. Given m samples of the random vector $\mathbf{X}$, estimation of Σ when $n > m$ cannot be carried out using a maximum likelihood estimator since the empirical covariance matrix is not full rank. Inspired by the success of L1 sparsity regularization for linear models, several authors suggested that Σ be estimated by finding the solution to the following regularized likelihood objective:

$$\widehat{\Omega} = \min_{\Omega} -\frac{1}{2}\left(\log|\Omega| - tr(\Omega\hat{S})\right) + \lambda \sum_{j\neq k} |\Omega_{jk}|, \tag{5}$$

where $\hat{S}$ is the sample covariance matrix. The estimator $\widehat{\Omega}$ can be computed efficiently by the glasso algorithm, which is simply a block coordinate descent that applies the standard lasso to a single row and column of Ω at each iteration. The resulting estimator has been shown to have appealing theoretical properties [41, 39].

Nonparanormal Estimation

A real-valued random vector $\mathbf{X}$ is said to have a nonparanormal distribution, $\mathbf{X} \sim NPN(\mu,\Sigma,g)$, if there exist functions $\{g_i\}_{i=1}^n$ such that $(g_1(X_1),\ldots,g_n(X_n)) \sim N(\mu,\Sigma)$. When g_i are monotone and differentiable, this is simply the Gaussian copula. Now, define

$$h_i(x) = \Phi^{-1}(F_i(x_i)),$$

and let Λ be the covariance matrix of $h(X)$. The independence properties discussed above for the multivariate Gaussian hold so that $X_i \perp X_j \mid \mathbf{X}_{\setminus\{i,j\}}$ if and only if $\Lambda_{ij}^{-1} = 0$. Thus, to estimate the graph's structure, it is sufficient to identify Λ^{-1}.

Consider the obvious rank based estimator for Λ that relies on the empirical marginal distribution function $\hat{F}_i(t) \equiv \frac{1}{m}\sum_{l=1}^m \mathbf{1}_{\{x_i[l]\leq t\}}$, where $x_i[l]$ is used to denote the assignment to X_i in the l'th sample. Unfortunately, using this estimator as a plug-in to covariance estimation does not work well in high dimension since the variance of $\hat{F}_i$ can be large. Instead, the following Winsorized estimator is suggested

$$\tilde{F}_i(x) = \begin{cases} \delta_m & \text{if } \hat{F}_i(x) < \delta_m \\ \hat{F}_i(x) & \text{if } \delta_m \leq \hat{F}_i(x) \leq 1-\delta_m \\ (1-\delta_m) & \text{if } \hat{F}_i(x) > 1-\delta_m, \end{cases}$$

where δ_m is a truncation parameter. Using $\delta_m \equiv \frac{1}{4m^{1/4}\sqrt{\pi\log m}}$ strikes the right bias-variance tradeoff that leads to the desirable theoretical properties discussed below. Given this estimate for the distribution of X_i, and using $\tilde{h}_i(x) = \Phi^{-1}\left(\tilde{F}_i(x)\right)$, define the transformation functions by

$$\tilde{g}_i(x) \equiv \hat{\mu}_i + \hat{\sigma}_i\tilde{h}_i(x), \tag{6}$$

where $\hat{\mu}_i$ and $\hat{\sigma}_i$ are sample mean and standard deviation of X_i, respectively. The sample covariance matrix $S_m(\tilde{g})$ can now be plugged in (5) in place of $\hat{S}$, defining a two-step estimation procedure for the estimator $\widehat{\Omega}_m$:

1. Replace the observations with Winsorized normalized scores as defined in (6).
2. Use the graphical lasso to estimate the undirected graph.

Appealingly, the procedure is both easy to compute and makes little assumptions regarding the distribution of $\mathbf{X}$. The only tuning parameter is the regularization parameter λ that defines the objective minimized by the glasso algorithm. Next, we summarize the theoretical and empirical merits of this estimator.

Properties of the Estimator

Building on the analysis of Rothman et al. [41] and Ravikumar et al. [39], Liu et al. are able to show that their estimator has favorable persistency, norm consistency and model selection consistency properties. The main technical result is an analysis of the covariance of the Winsorized estimator. Specifically, under appropriate conditions,

$$\max_{i,j} \left| S_m(\tilde{g})_{ij} - S_m(g)_{ij} \right| = O_P\left(m^{-1/4} \right).$$

Using this result, norm consistency of $\hat{\Omega}$ with respect to the Frobenius and L2 norm follows, with a similar dependence on m. Using additional technical assumptions, a model selection consistency result (so that the true structure is recovered) is also provided. Further, Liu et al. also show that their estimator is consistent in risk, that is when the true distribution is not assumed to be nonparanormal.

Liu et al. demonstrate the ability of their method to accurately recover known structure in simulation experiments under different transformations that are applied to the univariate marginals, and various training sample sizes. The also apply their method to biological and financial data, leading to structures that are different than those learned with a purely Gaussian model, potentially revealing novel insights. The interested reader is referred to Liu et al. [26] for details.

3.3 Copula Bayesian Networks

Elidan [8] tackles the task of flexibly representing a multivariate real-valued distribution based on a directed graph representation.

The CBN Model

As discussed in Section 2, a joint distribution that relies on a directed acyclic graph to encode independencies is quantified by local conditional densities. Accordingly, the construction starts with the following building block:

Lemma 3.1. *Let $f(x \mid \mathbf{y})$, with $\mathbf{y} = \{y_1, \ldots, y_k\}$, be a conditional density function. There exists a copula density function $c(F(x), F_1(y_1), \ldots, F_K(y_K))$ such that*

$$f(x \mid \mathbf{y}) = R_c(F(x), F_1(y_1), \ldots, F_K(y_K)) f_X(x),$$

where R_c is the copula ratio

$$R_c(F(x), F_1(y_1), \ldots, F_K(y_K)) \equiv \frac{c(F(x), F_1(y_1), \ldots, F_K(y_K))}{\frac{\partial^K C(1, F_1(y_1), \ldots, F_K(y_K))}{\partial F_1(y_1) \ldots \partial F_K(y_K)}},$$

and R_c is defined to be 1 *when $\mathbf{Y} = \emptyset$. The converse is also true: for* any *copula, $R_c(F(x), F_1(y_1), \ldots, F_K(y_K)) f_X(x)$ defines a valid conditional density.*

Note that the denominator of R_c is only seeming complex and is in fact a derivative of a lower order than the numerator copula density. Thus, whenever the copula density has a convenient form, so does R_c, and the conditional normalization does not involve any costly integration. With this building block in hand, the multivariate density model can be defined:

Definition 3.1. A Copula Bayesian Network (CBN) is a triplet $\mathscr{C} = (\mathscr{G}, \Theta_C, \Theta_f)$ that defines $f_{\mathbf{X}}(\mathbf{x})$. $\mathscr{G}$ encodes the independencies $\{(X_i \perp \mathbf{ND}_i \mid \mathbf{Pa}_i)\}$, assumed to hold in $f_{\mathbf{X}}(\mathbf{x})$. Θ_C is a set of local copula functions $\{C_i(F(x_i), F(\mathbf{pa}_{i1}), \ldots, F(\mathbf{pa}_{ik_i}))\}$ that are associated with the nodes of $\mathscr{G}$ that have at least one parent. In addition, Θ_f is the set of parameters representing the marginal densities $f_i(x_i)$ (and distributions $F_i(x_i)$). The joint density $f_{\mathbf{X}}(\mathbf{x})$ then takes the form

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{i=1}^{n} R_{c_i}\left(F(x_i), F(\mathbf{pa}_{i1}), \ldots, F(\mathbf{pa}_{ik_i})\right) f_i(x_i).$$

Elidan showed that if the independencies encoded in $\mathscr{G}$ hold in $f_{\mathbf{X}}(\mathbf{x})$, then the joint copula decomposes into a product of local copula ratio terms R_{c_i}. However, the converse is only partially true. The above product $\prod_i R_{c_i}(\cdot) f_i(x_i)$ *always* defines a valid joint density. However, the product $\prod_i R_{c_i}$, when each copula ratio is constructed independently, does not always define a valid copula. In this case, the marginals of the *valid* joint distribution do not necessarily equal to $F_i(x_i)$.

While this may seem unacceptable from a copula perspective, the model offers greater flexibility at the cost of marginal skewness, which in practice is not substantial. Moreover, when the structure of the graph $\mathscr{G}$ is a tree, the model collapses to the tree model described in Section 3.1, and the univariate marginals are preserved. Further, when using the Gaussian copula, the correct marginals can be maintained using an appropriate specification scheme, in which case the model is equivalent to a nonparametric BN model [24]. See Section 3.5 for further discussion.

Importantly, the above flexibility allows for the use of efficient algorithmic tools. Straightforwardly, assuming the marginals are estimated first, estimation of the entire CBN model decomposes into independent estimation of local copulas. Building on the same decomposability, standard greedy methods for structure learning can also be employed. More interestingly, the representation gives rise to approximate inference and structure learning innovations that are specifically tailored to the model. The latter is briefly described next while the interested reader is referred to Elidan [9] for details of the former.

Lightning-speed Structure Learning

Elidan [10] tackles the challenge of automated structure learning of CBNs in a high-dimensional settings. When the graph $\mathscr{G}$ is constrained to be a tree, the optimal structure can be learned using a maximum spanning tree procedure [6]. More generally, as the number of possible graphs is super-exponential in the number of variables, the common approach for structure learning is a greedy procedure that involves local structure modifications (e.g., single edge addition, delete and reversal) and is guided by a model selection score. Typical scores, such as the Bayesian Information Criterion (BIC) [43], balance the likelihood of the model and its complexity. See, for example, Koller and Friedman [22] for details and variants.

The building block of essentially all score-based structure learning methods for graphical models is the evaluation of the merit of an edge in the network. This involves computing the likelihood gain that would result from adding an edge to the network, which in turn involves estimation of the bivariate maximum likelihood parameters. In the case of the CBN model, this involves computation of

$$\textstyle\sum_{l=1}^{m} \log c_{\hat{\theta}}(F_X(x[l]), F_Y(y[l])),$$

where $\hat{\theta}$ are the estimated parameters, $x[l]$ is the value of X in the l'th instance, and the sum is over samples. Unfortunately, estimating $\hat{\theta}$, as well as the actual computation of the log-likelihood function can be difficult. In fact, for non-Gaussian real-valued models, even the learning of a tree structure can be prohibitive. Elidan [10] proposes an alternative that builds on the fact that as m grows, the above expression approaches the negative (differential) entropy:

$$-H(C_\theta(U,V)) = \int c_\theta(u,v) \log c_\theta(u,v) du dv, \tag{7}$$

with $U \equiv F_X, V \equiv F_Y$. However, computation of the copula entropy can also be difficult since for most copula families the above integral does not have a closed form. Instead, an efficient to compute proxy is proposed.

The relationship between Spearman's rho rank correlation measure of association $\rho_s(X,Y) \equiv \frac{cov(U,V)}{\sigma(U)\sigma(V)}$ and the copula function is well known: it can be easily shown (e.g., [30]) that for a distribution $f_{X,Y}(x,y)$ and its corresponding copula

$$\rho_s(X,Y) = \rho_s(C_\theta) \equiv 12 \iint C_\theta(U,V) du dv - 3. \tag{8}$$

Further, the vast majority of copula families define a concordance ordering where $\theta_2 > \theta_1$ implies $C_{\theta_2}(u,v) > C_{\theta_1}(u,v)$ for all u,v. Thus, for most copula families, Spearman's rho is monotonic in the dependence parameter θ.

Elidan identifies a further intriguing relationship: it is conjectured that Spearman's rho is monotonic *in the copula entropy*, possibly given some weak necessary conditions. The result is proved for elliptical copulas and for the Farlie-Gumbel-Morgenstern family. In addition, the conjecture is demonstrated via simulation for varied families whose only known commonality is concordance ordering.

Thus, in many cases, the easy to compute Spearman's rho can be used as a proxy to the expected log-likelihood, and asymptotically consistent model selection can be carried out for tree models. For several real-life datasets, where the underlying distribution in unknown, a near monotonic relationship is demonstrated in practice between the log-likelihood function and the empirical Spearman's rho. For more complex structures, Spearman's rho can be used to heuristically guide the learning procedure. The result is a lightning-speed procedure that learns structures that are as effective in terms of generalization to unseen test data as those learned by a costly exact procedure, with orders of magnitude improvement in running time. Appealingly, the running time improvement grows with the domain's complexity. A 100 variable structure, for example, is learned in essentially the same time that it takes to learn the structure of a naive Gaussian BN (less than a minute on a single CPU).

3.4 Copula Processes

Consider the problem of measuring the dependencies between real-valued measurements of a continuous process. For example, the dependence between a rocket's velocity at different times as it leaves earth, and how it relates to the dependence between the rocket's distances. As Wilson and Ghahramani [53] observe, these quantities are naturally on different scales and have different marginal distributions. Thus, it is desirable to separate the univariate effect from the dependence structure. Toward this goal, they define a copula process which can describe the dependence between *arbitrarily* many random variables.

Definition 3.2. Let $\{X_t\}$ be a collection of random variables indexed by t with marginal distributions $U_t \equiv F_t(X_t)$. Let G_t be the marginal distributions of a base process, and let H be the base joint distribution. X_t is a *copula process* with G_t, H, denoted $X_t \sim \mathrm{CP}(G_t, H)$, if for every finite set of indices $\mathscr{I} = \{t_1, \ldots, t_n\}$

$$P\left(\cap_{i=1}^{n}\{G_{t_i}^{-1}(U_{t_i}) \le a_i\}\right) = H_{t_1,\ldots,t_n}(a_1,\ldots,a_n),$$

where G_t^{-1} is the quasi-inverse of G_t. That is, for all $t_i \in \mathscr{I}$, H defines the joint distribution over $\{G_{t_i}^{-1}\}_{t_i \in \mathscr{I}}$.

As an example, consider the case where the base measure is a Gaussian process (GP). X_t is a GP if for every finite subset of indices $\mathscr{I}$, the set $\{X_{t_i}\}_{t_i \in \mathscr{I}}$ has a multivariate Gaussian distribution. To allow for a variable size set $\mathscr{I}$, a GP is parameterized by a mean function $m(t)$ that determines the expectation of the random variable X_t, and a kernel function $k(t,t')$ that determines the covariance of X_t and X_t'. GPs are widely used in machine learning to define distributions over an arbitrary number of random variables or functions (see Rasmussen [37]). When the base measure is chosen to be a GP, we say that X_t has a Gaussian copula process (GCP) distribution. This is equivalent to the existence of a mapping Ψ such that $\Psi(X_t)$ is a GP. We denote this by $X_t \sim \mathrm{GCP}(\Psi, m(t), k(t,t'))$.

In principle, given complete samples and a known mapping, one can estimate a GCP by simply transforming the data and using black box procedures for GP estimation, such as that of Snelson et al. [49]. Wilson and Ghahramani, however, consider a more challenging application setting that requires further algorithmic innovation. Concretely, they introduce a volatility model where the unobserved standard deviations of the data follow a GCP distribution

$$\sigma_t \sim \mathrm{GCP}(g^{-1}, 0, k(t,t')).$$

Model	References	variables	Structure	Copula	Comments
Vines	[3, 1, 25]	< 10 in practice	conditional dependence	any bivariate	well understood general purpose framework
Nonparametric BBN	[24, 16]	100s	BN + vines	Gaussian in practice	mature application
Tree-averaged	[21, 46] Section 3.1	10s	Mixture of trees	any bivariate	requires only bivariate estimation
Nonparanormal	[26] Section 3.2	100-1000s	MN	Gaussian	high-dimensional estimation with theoretical guarantees
Copula networks	[8, 10] Section 3.3	100s	BN	any	flexible at the cost of partial control over marginals
Copula processes	[53, 18] Section 3.4	∞ (replications)	-	multivariate	nonparametric generalization of Gaussian processes

Table 1 Summary of the different copula-based multivariate models

The observations $X_t \sim N(0, \sigma_t^2)$ are assumed to follow a normal distribution, though this assumption can easily be relaxed. The difficulty is rooted in the fact that the σ_t's are never observed, and that the so called *warping function g* is unknown.

Let θ be the parameters that define both the GP covariance function and the warping function. Further, using a different notation from Wilson and Ghahramani to maintain consistency, let $z_t = g^{-1}(\sigma_t)$ be the latent function values that have a GP distribution. The central components involved in estimating θ from samples x_t and making prediction at some unrealized time $t^\star$ are:

- A Laplace approximation for the posterior $f(f_Z(z_{t^\star}) \mid \mathbf{y}, \theta)$.
- A Markov Chain Monte Carlo technique to sample from this posterior, specifically the elliptical slice sampling method [29].
- A flexible parametric as well as nonparametric warping functions to transform the samples into standard deviation space.

We refer the interested reader to Wilson and Ghahramani [53] for the details, as well as favorable results relative to a GARCH model when applied to financial data.

3.5 Comparative Summary

In this section we summarize the relative merits of the different multivariate approaches presented in the previous sections. Also discussed is the relationships to vine models and a related BN-based construction. Table 3.5 summarizes the properties of each of the models discussed.

Vine models [19, 3] have become the dominant tool in the copula community for the construction of flexible multivariate copulas. The widely studied formalism builds on successive conditioning and the use of bivariate copulas to construct multivariate distributions. While the framework is quite general, the seemingly bivariate estimation relies on conditional terms of greater dimension that can be hard to estimate. In practice, most applications are computationally limited to less then 10 variables, with recent innovations (e.g., [5]) somewhat pushing this boundary.

The tree-average distribution model of Kirshner [21] described in Section 3.1 generalizes Darsow's Markovian operator and allows for the construction of high-dimensional copulas via a composition of (unconditional) bivariate copulas. Appealingly this requires only bivariate estimation but is hampered by the independence assumptions implied by the tree structure. These assumptions are relaxed by allowing for a mixture of all trees construction which is efficiently learned using a compactly represented prior. A Bayesian refinement of the work was later suggested by Silva and Gramacy [46]. The construction is practical for 10s of variables.

Distribution-free or nonparametric belief Bayesian networks (NPBBNs) [24, 16], are aimed at overcoming the limitations of simple vines by using a BN structure to encode a decomposition of the joint distribution, and employing local vines to encode $f_{X_i|\mathbf{Pa}_i}$. In principle, the construction can be used with any copula for which the specified conditional rank correlations can be realized. In practice, this can be carried out easily only when using an elliptical copula. That said, NPBBNs have led to the most mature and large-scale copula constructions to date.

Copula Bayesian networks (CBNs) [8], developed in the machine learning community also use on a BN structure to encode independencies that are assumed to hold in the distribution. The local conditional density, however, is parameterized differently via a proper normalization of a joint local copula over a variable and its parents in the graph. For tree structured models, a CBN reduces to the tree construction suggested by Kirshner [21]. When using a Gaussian copula, as discussed, it is also possible to estimate the parameters of the entire model so to ensure preservation of the univariate marginals. Thus specified, the model is equivalent to NPBBNs using local Gaussian copulas. However, CBNs also allow for greater flexibility at the cost of "skewed" marginals. Intuitively, this results from overlapping influences of multiple parents of a variable. Practically, since each local density is parameterized via an estimated joint copula with the same marginals, the overall univariate marginals are quite accurate. From a *given marginals* viewpoint this may be unacceptable. However, from a broader modeling perspective, in the face of finite data and an unknown joint distribution, the goals of maximum likelihood and full control over the univariate marginals are competing ones. In this light, a balance between flexible modeling and univariate control may be beneficial. Importantly, if one is willing to strike this balance, then the CBN construction opens the door for algorithmic advances from the field of probabilistic graphical models. Indeed, the experiments presented in Elidan [8] are the largest where the structure of the model was automatically learned. The construction also subsequently led to specifically tailored efficient inference [9] and structure learning methods [10].

The nonparanormal method of Liu et al. [26] tackles the problem of structure learning in the complementing representation of undirected graphs. While it is specifically focused on an Gaussian copula, it provides appealing theoretical guarantees of consistency when the data is generated from the model, as well as risk consistency guarantees when samples arise from a different distribution. Importantly, the method applies to the previously unstudied regime of nonparametric estimation in high-dimensions when the number of parameters exceeds that of the samples.

Finally, the copula process model of Wilson and Ghahramani [53] defines a distribution over an infinite number of random variables while allowing for the explicit control over the marginals, thus generalizing Gaussian processes. We note, that "infinite" here may be misleading since a "variable" is a replication, and Gaussian processes can also suffer from computational limitations. An obvious but challenging future prospect is the combination of this construction with local decomposability.

4 Information Estimation

Estimation of the mutual information of a set of variables is a fundamental challenge in machine learning that underlies numerous tasks ranging from learning the structure of graphical models to independent component analysis to image registration. However, for real-valued non-Gaussian random variables, estimation of different information measures can be difficult. In particular, the plug-in approach of computing the information based on an estimated density is often ineffective due to the difficulty of constructing complex joint distributions. Fortunately, just as copulas are opening new frontiers for modeling high-dimensional complex densities, so do they offer new opportunities for estimation of information measures. In this section we describe a series of recent works that build on such opportunities.

For all works discussed below, let $X[1:m] = \mathbf{X}[1],\ldots,\mathbf{X}[m]$ be m i.i.d. samples of $\mathbf{X}$. The first (obvious in the context of copulas) step of all works is a rank based transform $Z_i[l] = \frac{1}{m}\sum_{k=1}^{m} \mathbf{1}_{X_i[l]\leq X_i[k]}$. Asymptotically, Z_i will be uniformly distributed on $[0,1]$. However, the random samples $\mathbf{Z}[1],\ldots,\mathbf{Z}[m]$ are no longer independent. The works below take advantage of the former property and overcome the limitations of the latter consequence to produce appealing information estimators.

4.1 Information Estimation Based on Graph Optimization

The goal of both Póczos et al. [36] and Pal et al. [32] is to effectively estimate the Rényi information defined as

$$I_\alpha(\mathbf{X}) = \frac{1}{\alpha}\log\int f_{\mathbf{X}}^{\alpha}(\mathbf{x})\left(\prod_i f_i(x_i)\right)^{1-\alpha} d\mathbf{x}.$$

Note that when $\alpha \to 1$, Rényi information converges to the well known Shannon's mutual information measure. Rather then attempt to estimate $f_{\mathbf{X}}(\mathbf{x})$ which is a nuisance parameter, both works perform direct nonparametric estimation of $I_\alpha(\mathbf{X})$ by combining copula-based tools and graph-based estimators for the Rényi entropy

$$H_\alpha(\mathbf{X}) = \frac{1}{\alpha}\log\int f_{\mathbf{X}}^{\alpha}(\mathbf{x})d\mathbf{x}.$$

Although both works contain interesting contributions, for clarity of exposition we focus on the former and encourage the interested reader to explore the latter.

Let G be a graph with m nodes. Note that this is *not* a probabilistic graphical model over $\mathbf{X}$ but rather a graph whose nodes will index the training samples. Let $E(G)$ be the set of edges in G and let $\mathscr{G}$ be a family of such graphs. For example, $\mathscr{G}_{ST}$ will correspond to the family of all spanning trees over m nodes. Now define

$$L_m(\mathbf{X}[1:m]) = \min_{G \in \mathscr{G}} \sum_{l,k \in E(G)} \|\mathbf{X}[l] - \mathbf{X}[k]\|^p.$$

In words, $L_m(\cdot)$ is the minimum p-power weighted edge length of graphs in $\mathscr{G}$. For example, for $\mathscr{G}_{ST}$ and $p = 1$, $L_m(\cdot)$ is simply the length of the minimal spanning tree, a quantity readily found using efficient graph optimization. Remarkably, $L_m(\cdot)$ is also useful for entropy estimation:

Theorem 4.1. *(Steele [50]) Let* $n \geq 2, 0 < \alpha < 1$*, and let* $\mathbf{X}[1:m]$ *be i.i.d. random vectors supported on* $[0,1]^n$ *with density* $f_{\mathbf{X}}$*. Define the estimator*

$$H_m(\mathbf{X}[1:m]) = \frac{1}{1-\alpha} \log \frac{L_m(\mathbf{X}[1:m])}{\gamma_{n,\alpha} m^{\alpha}},$$

where $\gamma_{n,\alpha}$ *is a constant that does not depend on* $f_{\mathbf{X}}$*. Then,* $H_m(\mathbf{X}[1:m]) \to H_\alpha(\mathbf{X})$ *almost surely as* $m \to \infty$ *(similar theorems exist for other graph families* $\mathscr{G}$*, see Póczos et al. [36] for details and references).*

The first obstacle in using the above theorem is that it applies to variables that are supported on $[0,1]^n$. This is easily overcome by the rank-based transform that results in $\mathbf{Z}[1], \ldots, \mathbf{Z}[m]$. Now, since Z_i is defined via a measurable invertible mapping, $I_\alpha(\mathbf{Z}) = I_\alpha(\mathbf{X})$. Further, since the marginals of $\mathbf{Z}$ are uniform, we have $I_\alpha(\mathbf{Z}) = -H_\alpha(\mathbf{Z})$ so that an entropy estimator can used to estimate information (this generalizes the known fact that Shannon's information is equal to the negative copula entropy). The transform, however, introduces a new difficulty since the samples $Z[m]$ are now dependent. Poczos et al [36] shows that despite this the estimator has favorable strong consistency and robustness properties. They also demonstrate the advantage of their rank based approach in practice, for an image registration task.

4.2 Kernel-based Dependency Measures

Like the above works, Póczos et al. [34] also start with an empirical rank transformation of the data followed by the application of an existing distance measure between distributions. The combination, however, is quite different than the graph optimization based approaches described above. Omitting most of the technical details, we briefly present the high level idea and the merits of the resulting estimator. We start with the definition of the maximum mean discrepancy (MMD) measure of distributions similarity, which can be efficiently estimated from i.i.d. samples:

Definition 4.1. Let $\mathscr{F}$ be a class of functions, P and Q be probability distributions. The MMD between P and Q on the function class $\mathscr{F}$ is defined as follows:

$$\mathscr{M}[\mathscr{F},P,Q] \equiv \sup_{f\in\mathscr{F}} \left(\mathbb{E}_{\mathbf{X}\sim P}[f(\mathbf{x})] - \mathbb{E}_{\mathbf{Y}\sim Q}[f(\mathbf{y})]\right).$$

We will focus on functional spaces that are a reproducing kernel Hilbert Space (RKHS), a fundamental tool in machine learning [42]. Without going into the technical definition, the notion of RKHS is important since, assuming that $\mathscr{F}$ is a unit ball of RKHS $\mathscr{H}$, measures such as $\mathscr{M}(\mathscr{F},P,Q)$ and related quantities can be estimated efficiently [4]. Building on this fact, consider the following dependence measure

$$I_k(X_1,\ldots,X_n) \equiv \mathscr{M}(\mathscr{F},F_{\mathbf{X}},F_{\mathbf{U}})$$

where we use $F_{\mathbf{U}}$ to denote the n-dimensional uniform distribution. Póczos et al. [34] show that if ones chooses $\mathscr{F}$ properly (a RKHS with an additional denseness requirement), then I_k is a proper dependence measure that follows Schweizer and Wolffs's intuitive axioms [44]. They suggest an empirical estimator for I_k that is based on an empirical MMD estimation of the rank transformed samples $\mathbf{Z}[m]$, and prove that their easy to compute estimator is almost surely consistent. Further, they provide upper bound convergence rates. Finally, they demonstrate the merit of the estimator in practice in the context of a feature selection task.

5 Summary

In the introduction it was argued that, in the context of multivariate modeling and information estimation, the complementing strengths and weaknesses of the fields of machine learning and that of copulas offer opportunities for symbiotic constructions. This paper surveyed the main such synergic works that recently emerged in the machine learning community.

While discrete high-dimensional modeling has been studied extensively, real-valued modeling for more than a few dimensions is still in its infancy. There exists no framework that is as general and as flexible as copulas for multivariate modeling. Thus, it is inevitable that machine learning researchers who aim to stop discretizing data, will have to pay serious attention to the power of copulas. Conversely, if researchers in the copula community aim to cope with truly high-dimensional challenges, algorithmic prowess, a focus of the machine learning community, will have to be used. True, impressive large-scale models have been built using NPBBNs. However, the multi-year endeavor supported by human expertise cannot be scaled up or easily applied to a broad range of problems. Automated learning of models that takes into account the difficulties presented by the high-dimensional and partially observed setting is clearly needed. The goal of this survey is to provide an entry point for those aiming to tackle this far from realized challenge.

Acknowledgements G. Elidan was partially supported by a Google research grant and by an Israeli Science Foundation Center of Research grant.

References

1. Aas, K., Czado, C., Frigessi, A., Bakken, H.: Pair-copula constructions of multiple dependencies. Insurance: Mathematics and Economics **44**, 182–198 (2009)
2. Abayomi, K.: Diagnostics for Multivariate Imputation, Copula Based Independent Component Analysis and a Motivating Example. Columbia University (2008)
3. Bedford, T., Cooke, R.: Vines - a new graphical model for dependent random variables. Annals of Statistics (2002)
4. Borgwardt, K.M., Gretton, A.A., Rasch, B.M.J., peter Kriegel, H., Schlkopf, A.B., D, B.A.J.S.: Integrating structured biological data by kernel maximum mean discrepancy. In: Proccedings of Intelligent Systems for Molecular Biology (ISMB) (2006)
5. Brechmann, E.C., Czado, C., Aas, K.: Truncated regular vines in high dimensions with applications to financial data. Canadian Journal of Statistics **40**(1), 68–85 (2012)
6. Chow, C.K., Liu, C.N.: Approximating discrete probability distributions with dependence trees. IEEE Trans. on Info. Theory **14**, 462–467 (1968)
7. Darsow, W., Nguyen, B., Olsen, E.: Copulas and Markov processes. Illinois Journal of Mathematics **36**, 600–642 (1992)
8. Elidan, G.: Copula Bayesian networks. In: Neural Info. Processing Systems (NIPS) (2010)
9. Elidan, G.: Inference-less density estimation using copula bayesian networks. In: Uncertainty in Artificial Intelligence (UAI) (2010)
10. Elidan, G.: Lightning-speed structure learning of nonlinear continuous networks. In: Proceedings of the AI and Statistics Conference (AISTATS) (2012)
11. Embrechts, P., Lindskog, F., McNeil, A.: Modeling dependence with copulas and applications to risk management. Handbook of Heavy Tailed Distributions in Finance (2003)
12. Ferguson, T.S.: A Bayesian analysis of some nonparametric problems. The Annals of Statistics **1**(2), 209–230 (1973)
13. Friedman, J., Hastie, T., Tibshirani, R.: Sparse inverse covariance estimation with the graphical lasso. Biostatistics **9**(3), 432–441 (2007)
14. Fujimaki, R., Sogawa, Y., Morinaga, S.: Online heterogeneous mixture modeling with marginal and copula selection. In: Proceedings of the ACM SIGKDD international conference on Knowledge discovery and data mining (2011)
15. Hammersley, J.M., Clifford, P.E.: Markov fields on finite graphs and lattices (1971). Unpublished manuscript.
16. Hanea, A., Kurowicka, D., Cooke, R.M., Ababei, D.: Mining and visualising ordinal data with non-parametric continuous bbns. Comp Statistics and Data Analysis **54**(3), 668–687 (2010)
17. Huang, J., Frey, B.: Cumulative distribution networks and the derivative-sum-product algorithm. In: Uncertainty in Artificial Intelligence (UAI) (2008)
18. Jaimungal, S., Ng, E.: Kernel-based copula processes. In: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part I (2009)
19. Joe, H.: Families of m-variate distributions with given margins and m(m-1)/2 bivariate dependence parameters. In: Distributions with Fixed Marginals and Related Topics (1996)
20. Joe, H., Xu, J.: The Estimation Method of Inference Functions for Margins for Multivariate Models. Technical report no. 166, Dept. of Statistics, University of British Columbia (1966).
21. Kirshner, S.: Learning with tree-averaged densities and distributions. In: Neural Information Processing Systems (NIPS) (2007)
22. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. The MIT Press (2009)
23. Kurowicka, D., Cooke, R.: The vine copula method for representing high dimensional dependent distributions: Applications to cont. belief nets. In: Proc. of the Simulation Conf. (2002)
24. Kurowicka, D., Cooke, R.M.: Distribution-free continuous bayesian belief nets. In: Modern statistical and mathematical methods in reliability. Selected papers based on the presentation at the international conference on mathematical methods in reliability (MMR) (2005)

25. Kurowicka, D., Joe, H.: Dependence Modeling: Vine Copula Handbook. World Scientific (2011)
26. Liu, H., Lafferty, J., Wasserman, L.: The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. Journal of Machine Learning Research (2010)
27. Ma, J., Sun, Z.: Copula component analysis. In: Proceedings of the International conference on Independent component analysis and signal separation (2007)
28. Meilă, M., Jaakkola, T.: Tractable bayesian learning of tree belief networks. Statistics and Computing **16**(1), 77–92 (2006)
29. Murray, I., Adams, R., MacKay, D.: Elliptical slice sampling. In: Proceedings of the AI and Statistics Conference (AISTATS) (2010)
30. Nelsen, R.: An Introduction to Copulas. Springer (2007)
31. Newman, D., Hettich, S., Blake, C., Merz, C.: UCI repository of machine learning databases (1998). URL `http://www.ics.uci.edu/~mlearn/MLRepository.html`
32. Pal, D., Póczos, B., Szepesvári, C.: Estimation of Rényi entropy and mutual information based on generalized nearest-neighbor graphs. In: Neural Info. Processing Systems (NIPS) (2010)
33. Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann (1988)
34. Póczos, B., Ghahramani, Z., Schneider, J.: Copula-based kernel dependency measures. In: Proceedings of the International Conference on Machine Learning (ICML) (2012)
35. Póczos, B., Kirshner, S.: Ica and isa using schweizer-wolff measure of dependence. In: Proceedings of the International Conference on Machine Learning (ICML) (2008)
36. Póczos, B., Kirshner, S., Szepesvári, C.: REGO: Rank-based estimation of Rényi info. using Euclidean graph optimization. In: Proc. of the AI and Statistics Conference (AISTATS) (2010)
37. Rasmussen, C.E.: Gaussian processes for machine learning. MIT Press (2006)
38. Ravikumar, P., Lafferty, J., Liu, H., Wasserman, L.: Sparse additive models. Journal of the Royal Statistical Society Series B **71**(5), 1009–1030 (2009)
39. Ravikumar, P., Wainwright, M., Raskutti, G., Yu, B.: Model selection in gaussian graphical models: High-dimensional consistency of l1-regularized mle. In: Neural Information Processing Systems (NIPS). MIT Press, Cambridge, Massachusetts (2009)
40. Rey, M., Roth, V.: Copula mixture model for dependency-seeking clustering. In: International Conference on Machine Learning (ICML) (2012)
41. Rothman, A., Bickel, P., Levina, E., Zhu, J.: Sparse permutation invariant covariance estimation. Electronic Journal of Statistics **2**, 494–515 (2008)
42. Schölkopf, B., Smola, A.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press (2001)
43. Schwarz, G.: Estimating the dimension of a model. Annals of Statistics **6**, 461–464 (1978)
44. Schweizer, B., Wolff, E.: On nonparameteric measures of dependence for random variables. The Annals of Statistics **9** (1981)
45. Silva, R., Blundell, C., Teh, Y.: Mixed cumulative distribution networks. Proceedings of the AI and Statistics Conference (AISTATS) (2011)
46. Silva, R., Gramacy, R.: Mcmc methods for bayesian mixtures of copulas. Proceedings of the AI and Statistics Conference (AISTATS) (2009)
47. Silva, R., Gramacy, R.B.: Mcmc methods for bayesian mixtures of copulas. In: Proceedings of the AI and Statistics Conference (AISTATS) (2009)
48. Sklar, A.: Fonctions de repartition a n dimensions et leurs marges. Publications de l'Institut de Statistique de L'Universite de Paris **8**, 229–231 (1959)
49. Snelson, E., Rasmussen, C.E., Ghahramani, Z.: Warped gaussian processes. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS (NIPS). MIT Press (2003)
50. Steele, J.: Probability Theory and Combinatorial Optimization. Society for Industrial and Applied Mathematics (1987)
51. Tewari, A., Giering, M.J., Raghunathan, A.: Parametric characterization of multimodal distributions with non-gaussian modes. In: IEEE Conf on Data Mining Workshops (2011)
52. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal fo the Royal Statistical Society B **58**(1), 267–288 (1996)
53. Wilson, A., Ghahramani, Z.: Copula processes. In: Neural Information Processing Systems (NIPS) (2010)