Mixed Cumulative Distribution Networks

Ricardo Silva
ricardo@stats.ucl.ac.uk
Department of Statistical Science, UCL

Charles Blundell
c.blundell@gatsby.ucl.ac.uk
Gatsby Unit, UCL

Yee Whye Teh
ywteh@gatsby.ucl.ac.uk
Gatsby Unit, UCL

Abstract

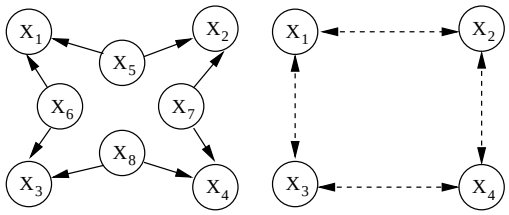
Directed acyclic graphs (DAGs) are a popular framework to express multivariate probability distributions. Acyclic directed mixed graphs (ADMGs) are generalizations of DAGs that can succinctly capture much richer sets of conditional independencies, and are especially useful in modeling the effects of latent variables implicitly. Unfortunately, there are currently no parameterizations of general ADMGs. In this paper, we apply recent work on cumulative distribution networks and copulas to propose one general construction for ADMG models. We consider a simple parameter estimation approach, and report some encouraging experimental results.

1 CONTRIBUTION

Graphical models provide a powerful framework for encoding independence constraints in a multivariate distribution (Pearl, 1988, Lauritzen, 1996). Two of the most common families, the directed acyclic graph (DAG) and the undirected network, have complementary properties. For instance, DAGs are non-monotonic independence models, in the sense that conditioning on extra variables can also destroy independencies (sometimes known as the "explaining away" phenomenon (Pearl, 1988)). Undirected networks allow for flexible "symmetric" parameterizations that do not require a particular ordering of the variables.

More recently, alternative graphical models that allow for both directed and symmetric relationships have been introduced. The *acyclic directed mixed graph* (ADMG) has both directed and bi-directed edges and it is the result of *marginalizing* a DAG: Figure 1 provides an example. Richardson and Spirtes (2002), Richardson (2003) show that DAGs are not closed under marginalization, but ADMGs are. Reading off independence constraints from a ADMG can be done with a procedure essentially identical to d-separation (Pearl, 1988, Richardson and Spirtes, 2002). Given a graphical structure, the challenge is to provide a procedure to parameterize models that correspond to the independence constraints of the graph, as illustrated below.

Example 1: Bi-directed edges correspond to some hidden common parent that has been marginalized. In the Gaussian case, this has an easy interpretation as constraints in the marginal covariance matrix of the remaining variables. Consider the two graphs below.



In the DAG in the left, we marginalize variables $X_5, \ldots, X_8$, obtaining the (fully bi-directed) ADMG on the right. Consider a Gaussian distribution that is Markov with respect to this graph. Its covariance matrix will have the following structure:

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} & 0 \\ \sigma_{12} & \sigma_{22} & 0 & \sigma_{24} \\ \sigma_{13} & 0 & \sigma_{33} & \sigma_{34} \\ 0 & \sigma_{24} & \sigma_{34} & \sigma_{44} \end{bmatrix}$$

That is, the absence of an edge in the fully bi-directed case will correspond to a zero in the implied covariance matrix. This should be contrasted with the undirected Gaussian Markov random field, where zeroes are in the inverse covariance matrix. □

Theoretical properties and practical applications of ADMGs are further discussed in detail by e.g. Bollen (1989), Spirtes et al. (2000), Drton and Richardson (2008), Zhang (2008), Pellet (2008), Silva and Ghahramani (2009), Khare and Rajaratnam (2009), Huang and Jojic (2010). One can also have latent variable ADMG models, where only a subset of the latent variables have been marginalized. In sparse

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15.

models, using bi-directed edges in ADMGs frees us from having to specify exactly which latent variables exist and how they might be connected. In the context of Bayesian inference, Markov chain Monte Carlo in ADMGs might have much better mixing properties compared to models where all latent variables are explicitly included (Silva and Ghahramani, 2009).

However, it is hard in general to parameterize a likelihood function that obeys the independence constraints encoded in an ADMG. Gaussian likelihood functions and their variations (e.g., mixture models and probit models) have been the most common families exploited in the literature (Richardson and Spirtes, 2002, Silva and Ghahramani, 2009, Khare and Rajaratnam, 2009, Rothman et al., 2009). More recently, important progress has been made in constructing binary ADMG models (Drton and Richardson, 2008, Richardson, 2009, Evans and Richardson, 2010), although it is not clear how to extend such models to infinite discrete spaces (such as treating Poisson random variables) – also important, scalability issues arise, as described in the sequel.

This paper provides a flexible construction procedure for probability mass functions and density functions that are Markov with respect to an arbitrary ADMG. In the case where complete parameterizations exist, such as in the multivariate binary case (Richardson, 2009, Evans and Richardson, 2010), our construction has complementary properties: while it provides only a subclass of all binary ADMG models compatible with a given graph (hence less attractive in applications such as joint hypothesis testing of ADMG constraints), it has computational advantages.

Our construction is done by exploiting recent work on *cumulative distribution networks*, CDNs (Huang and Frey, 2008) and *copulas* (Nelsen, 2007, Kirshner, 2007). The usefulness of such parameterizations can then be put to test via some parameter estimation procedure, which in our case will be based on Bayesian learning with Markov chain Monte Carlo (MCMC) We review mixed graphs and cumulative distribution networks in Section 2. The full formalism is given in detail in Section 3. An instantiation of the framework based on copulas is described in Section 4, followed by a short description of a Bayesian parameter learning procedure in Section 5. Experiments are described in Section 6, and we conclude with Section 7.

2 BI-DIRECTED GRAPHS AND CDNS

In this section, we provide a summary of the relevant properties of mixed graph models and cumulative distribution networks, and the relationship between formalisms.

A *bi-directed* graph is a special case of a ADMG without directed edges. The absence of an edge (X_i, X_j) implies that X_i and X_j are *marginally independent*. Hence, bi-directed models are *models of marginal independence* (Drton and Richardson, 2008). Just like in a DAG, conditioning on a vertex that is the endpoint of two arrowheads will make some variables dependent. For instance, for a bi-directed graph $X_1 \leftrightarrow X_2 \leftrightarrow X_3$, we have that $X_1 \perp\!\!\!\perp X_3$ but $X_1 \not\perp\!\!\!\perp X_3 | X_2$. See Drton and Richardson (2003, 2008) for a full discussion[1].

Current parameterizations of bi-directed graphs have many desirable properties but suffer from a number of important practical difficulties. For example, consider binary bi-directed graphs, where a complete parameterization was introduced by Drton and Richardson (2008). Let $\mathcal{G}$ be a bi-directed graph with vertex set X_V. Let $q_A \equiv P(X_A = 0)$, for any vertex set X_A contained in X_V. The joint probability $P(X_A = 0, X_{V \setminus A} = 1)$ is given by

$$P(X_A = 0, X_{V \setminus A} = 1) = \sum_{B : A \subseteq B} (-1)^{|B \setminus A|} q_B \qquad (1)$$

The set $\{q_S : X_S \subset X_V\}$ is known as the Möbius parameterization of $P(X_V)$, since relationship (1) is an instance of the Möbius inversion operation (Lauritzen, 1996). The marginal independence properties of the bi-directed graph imply $P(X_A = 0, X_B = 0) = P(X_A = 0)P(X_B = 0)$ if no element in X_A is adjacent to any element in X_B in $\mathcal{G}$. Therefore, the set of independent parameters in this parameterization is given by $\{q_A\}$, for all X_A that forms a connected set in $\mathcal{G}$. This parameterization is complete, in the sense that *any* binary model that is Markov with respect to $\mathcal{G}$ can be represented by an instance of set $\{q_A\}$. However, this comes at a price: in general, the number of connected sets can grow exponentially in $|X_V|$ even for a sparse, tree-structured, graph. Moreover, the set $\{q_A\}$ is not *variation independent* (Lauritzen, 1996): the parameter space is defined by exponentially many constraints, unlike more standard graphical models (Lauritzen, 1996, Pearl, 1988).

Cumulative distribution networks (CDNs), introduced by Huang and Frey (2008) as a convenient family of cumulative distribution functions (CDFs), provide a alternative construction of bi-directed models by indirectly introducing additional constraints to reduce the total number of parameters. Let X_V be a set of random variables, and let $\mathcal{G}$ be a bi-directed graph[2] with $\mathcal{C}$ being a set of cliques in $\mathcal{G}$. The CDF over X_V is given by

$$P(X_V \leq x_V) \equiv F(x_V) = \prod_{S \in \mathcal{C}} F_S(x_S) \qquad (2)$$

where each F_S is a parametrized CDF over X_S. A sufficient condition for (2) to define a valid CDF is that each F_S is itself a CDF. CDNs satisfy the conditional independence

[1] Notice also the difference with respect to the undirected model $X_1 - X_2 - X_3$, where $X_1 \not\perp\!\!\!\perp X_3$ but $X_1 \perp\!\!\!\perp X_3 | X_2$.

[2] Huang and Frey (2008) describe the model in terms of factor graphs, but for our purposes a bi-directed representation is more appropriate.

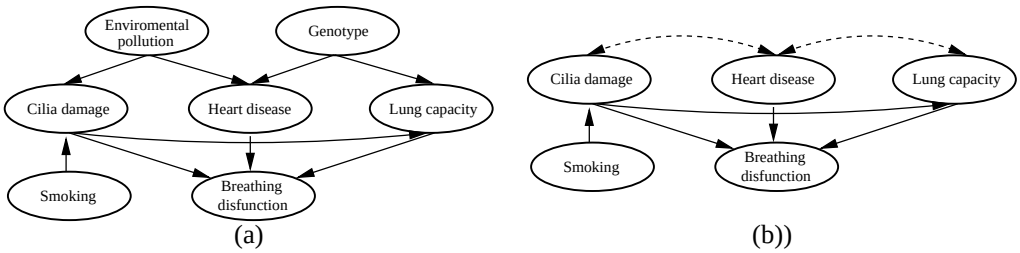


Figure 1: (a) A DAG representing dependencies over a set of variables (adapted from Spirtes et al. (2000), page 137) in a medical domain. (b) The ADMG representing conditional independencies corresponding to (a), but only among the remaining vertices: pollution and genotype factors were marginalized. In general, bi-directed edges emerge from unspecified variables that have been marginalized but still have an effect on the remaining variables. The ADMG is acyclic in the sense that there are no cycles composed of directed edges only. In general, a DAG cannot represent the remaining set of independence constraints after some variables in another DAG have been marginalized.

constraints of bi-directed graphs (Huang and Frey, 2008). For example, consider $X_1 \leftrightarrow X_2 \leftrightarrow X_3$, with cliques $X_{S_1} = \{X_1, X_2\}$ and $X_{S_2} = \{X_2, X_3\}$. The marginal CDF of X_1 and X_3 is $P(X_1 \leq x_1, X_3 \leq x_3) = P(X_1 \leq x_1, X_2 \leq \infty, X_3 \leq x_3) = F_1(x_1, \infty)F_2(\infty, x_3)$. Since this factorizes, it follows that X_1 and X_3 are marginally independent.

The relationship between the complete parameterization of Drton and Richardson and the CDN parameterization can be illustrated in the discrete case. Let each X_i take values in $\{0, 1, 2, ...\}$. Recall that the relationship between a CDF and a probabiliy mass function is given by the following inclusion-exclusion formula (Joe, 1997):

$$P(x_1, \ldots, x_d) = \tag{3}$$

$$\sum_{z_1=0}^{1} \cdots \sum_{z_d=0}^{1} (-1)^{z_1+z_2+\ldots z_d} F(x_1 - z_1, \ldots, x_d - z_d),$$

for $d = |X_V|$. In the binary case, since $q_A = P(X_A = 0) = P(X_A \leq 0, X_{V \setminus A} \leq 1) = F(x_A = 0, x_{V \setminus A} = 1)$, one can check that (3) and (1) are the same expression. The difference between the CDN parameterization (Huang and Frey, 2008) and the complete parameterization (Drton and Richardson, 2008) is that, on top of enforcing $q_{A \cup B} = q_A q_B$ for X_A disconnected from X_B, we have the additional constraints

$$q_A = \prod_{A_C \in \mathcal{C}(A)} q_{A_C} \tag{4}$$

for each connected set X_A, where $\mathcal{C}(A)$ are the maximal cliques in the subgraph obtained by keeping only the vertices X_A and the corresponding edges from $\mathcal{G}$[3].

As a framework for the construction of bi-directed models, CDNs have three major desirable features. First, the number of parameters grows with the size of the largest clique, instead of $|X_V|$. Second, parameters in different cliques are variation independent, since (2) is well-defined if each individual factor is a CDF. Third, this is a general framework that allows not only for binary variables, but continuous, ordinal and unbounded discrete variables as well. Finally, in graphs with low tree-widths, probability densities/masses can be computed efficiently by dynamic programming (Huang and Frey, 2008, Huang et al., 2010).

To summarize, CDNs provide a restricted family of marginal independence models, but one that has computational, statistical and modeling advantages. Depending on the application, the extra constraints may not be harmful in practice, as demonstrated by Huang and Jojic (2010), Huang et al. (2010).

3 MIXED CDN MODELS

In what follows, we will extend the CDN family to general acyclic directed mixed graphs: the *mixed* cumulative distribution network (MCDN) model. In Section 3.1, we describe a higher-level factorization of the *probability* (mass or density) *function* $P(X_V)$ involving subgraphs of $\mathcal{G}$. In Section 3.2, we describe cumulative distribution functions that can be used to parameterize each factor defined in Section 3.1, in the special case where no directed edges exist between members of a same subgraph. Finally, in Section 3.3, we describe the general case.

Some important notation and definitions: there are two kinds of edges in an ADMG; either $X_k \rightarrow X_j$ or $X_k \leftrightarrow X_j$. We use $pa_{\mathcal{G}}(X_A)$ to represent the *parents* of a set of vertices X_A in graph $\mathcal{G}$. For a given $\mathcal{G}$, $(\mathcal{G})_A$ represents the subgraph obtained by removing from $\mathcal{G}$ any vertex *not* in set A and the respective edges; $(\mathcal{G})_{\leftrightarrow}$ is the subgraph obtained by removing all directed edges. We say that a set of nodes A in $\mathcal{G}$ is an *ancestral set* if it is closed under the ancestral relationship: if $X_v \in A$, then all ancestors of X_v in $\mathcal{G}$ are also in A. Finally, define the *districts* of a graph $\mathcal{G}$

[3]This property was called *min-independence* in Huang (2009). To the best of our knowledge, our exposition linking CDNs to the parameterization (1) was never made explicit in Huang (2009) or elsewhere.

as the (maximal) connected components of $(\mathcal{G})_{\leftrightarrow}$. Hence each district is a set of vertices, X_D, such that if X_i and X_j are in X_D then there is a path connecting X_i and X_j composed entirely of bi-directed edges. Because districts are maximal sets, they define a partition of X_V. Note that trivial districts are permitted, where $X_D = \{X_i\}$. Furthermore there can be no directed cyclic paths in the ADMG.

Associated with each district X_{D_i} is a subgraph $\mathcal{G}_i$ consisting of nodes $X_{D_i} \cup pa_{\mathcal{G}}(X_{D_i})$. The edges of $\mathcal{G}_i$ are all of the edges of $(\mathcal{G})_{X_{D_i} \cup pa_{\mathcal{G}}(X_{D_i})}$ excluding all edges among $pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i}$. Two examples are shown in Figure 2.

3.1 District factorization

Given any ADMG $\mathcal{G}$ with vertex set X_V, we parameterize its probability mass/density function as:

$$P(X_V) = \prod_{i=1}^{K} P_i(X_{D_i} \mid pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i}) \qquad (5)$$

where $\{X_{D_1}, X_{D_2}, \ldots, X_{D_K}\}$ is the set of districts of $\mathcal{G}$. That is, each factor is a probability (mass/density) function for X_{D_i} given its set of parents in $\mathcal{G}$ (that are not already in X_{D_i}). We require that

- Each $P_i(X_{D_i} \mid pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i})$ is Markov with respect to $\mathcal{G}_i$,

where a probability (mass or density) function $P(Z \mid Z')$ is *Markov with respect* to a ADMG $\mathcal{G}$ if any conditional independence constraint verifiable in $P(Z \mid Z')$ that is encoded in $\mathcal{G}$ also holds in $P(Z \mid Z')$[4].

The relevance of this factorization is summarized by the following result.

Proposition 1. *A probability (mass or density) function $P(X_V)$ is Markov with respect to $\mathcal{G}$ if it can be factorized according to (5) and each $P_i(X_{D_i} \mid pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i})$ is Markov with respect to the respective $\mathcal{G}_i$.*

The proof of this result is in the Supplementary Material.

Note that (5) is seemingly cyclical: for instance, Figure 2(a) implies the factorization $P_1(X_1, X_2 \mid X_4) P_2(X_3, X_4 \mid X_1)$. This suggests that there are additional constraints tying parameters across different factors. However, there are no such constraints, as guaranteed through the following result:

[4]This is a slight generalization of the Markov condition, as seen in e.g. Spirtes et al. (2000), in the sense that we are excluding independence statements that cannot logically be verified from $P(Z \mid Z')$ alone − such as statements concerning marginal independence of two subsets of Z'.

Proposition 2. *Given an ADMG $\mathcal{G}$ with respective subgraphs $\{\mathcal{G}_i\}$ and districts $\{X_{D_i}\}$, any collection of probability functions $P_i(X_{D_i} \mid pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i})$, Markov with respect to the respective $\mathcal{G}_i$, implies that (5) is a valid probability function (a non-negative function that integrates to 1).*

Proof: There must be some X_v with no children in $\mathcal{G}$, since the graph is acyclic. Those childless vertices can be marginalized in the usual way, as they do not appear on the conditioning side of any factor $P_i(\cdot \mid \cdot)$, and removed from the graph along with all edges adjacent to them. After all such standard marginalizations, suppose that in the current marginalized graph, each childless vertex $X_\emptyset$ appears on the conditioning side of some factor $P_i(X_{S_i} \mid pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i})$, where $X_{S_i} \subset X_{D_i}$. Because $X_\emptyset$ has no children in X_{S_i}, by construction X_{S_i} are $X_\emptyset$ are independent given the remaining elements in $pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i}$. As such, $X_\emptyset$ can be removed from the right-hand side of all remaining factors, and then marginalized. The process is repeated until the last remaining vertex is marginalized, giving 1 as the result. Moreover, it is clear that (5) is non-negative. □.

The implication is that one can independently parameterize each individual $P_i(\cdot \mid \cdot)$ to obtain a valid $P(X_V)$ Markov with respect to any given ADMG $\mathcal{G}$. In the next sections, we show how to parameterize each $P_i(\cdot \mid \cdot)$ by factorizing its corresponding cumulative distribution function.

3.2 Models with barren districts

Consider first the case where district X_{D_i} is *barren*, that is, no $X_v \in X_{D_i}$ has a parent also in X_{D_i} (Richardson, 2009). For a given $\mathcal{G}_i$ with respective district X_{D_i}, consider the following function:

$$F_i(x_{D_i} \mid pa_{\mathcal{G}}(X_{D_i})) \quad \equiv \quad \prod_{X_S \in \mathcal{C}_i} F_S(x_S \mid pa_{\mathcal{G}}(X_{D_i})) \qquad (6)$$

where $\mathcal{C}_i$ is the set of cliques in $(\mathcal{G}_i)_{\leftrightarrow}$. Each term on the right hand side is a conditional cumulative distribution function: for sets of random variables Y and Z, $F(y \mid z) \equiv P(Y \leq y \mid Z = z)$.

Proposition 3. *$F_i(x_{D_i})$ is a CDF for any choice of $\{\{F_S(x_S)\}, \{F_v(x_v \mid pa_{\mathcal{G}}(X_v))\}\}$. If, according to each $F_S(x_S)$, $X_s \in X_S$ is marginally independent of any element in $pa_{\mathcal{G}}(X_{D_i}) \backslash pa_{\mathcal{G}}(X_s)$, the corresponding conditional probability function $F_i(x_{D_i} \mid pa_{\mathcal{G}}(X_{D_i}))$ is Markov with respect to $\mathcal{G}_i$.*

Proof: Each factor in (6) is a CDF with respect to X_{D_i}, with $pa_{\mathcal{G}}(X_{D_i})$ fixed, and hence its product is also a CDF (Huang and Frey, 2008). To show

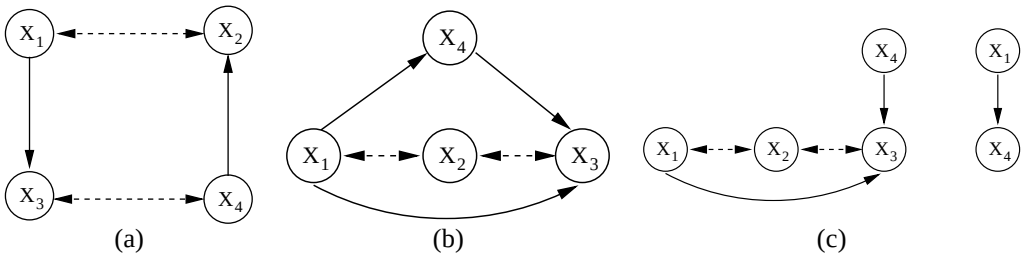


Figure 2: (a) The ADMG has two districts, $X_{D_1} = \{X_1, X_2\}$ with singleton parent X_4, and $X_{D_2} = \{X_3, X_4\}$ with parent X_1. (b) A more complicated example with two districts. Notice that the district given by $X_{D_1} = \{X_1, X_2, X_3\}$ has as external parent X_4, but internally some members of the district might be parents of other members. The other district is a singleton, $X_{D_2} = \{X_4\}$. (c) The two corresponding subgraphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are shown here.

the Markov property, suppose that the graph implies $X_A \perp\!\!\!\perp X_B \mid X_C \cup pa^\star_{\mathcal{G}}(X_{D_i})$ for disjoint sets X_A, X_B, X_C where: $X_A \cup X_C \subseteq X_{D_i}$, $X_B \subseteq X_{D_i} \cup pa_{\mathcal{G}}(X_{D_i})$, and $pa^\star_{\mathcal{G}}(X_{D_i}) \subset pa_{\mathcal{G}}(X_{D_i}) \backslash X_A \cup X_B$. This means there is no (bi-directed) path between any pair of elements $X_a \in X_A$ and $X_b \in X_B$ composed of elements of X_C only (Richardson and Spirtes, 2002, Drton and Richardson, 2008). This fact, plus the given assumption that each $X_s \in X_S$ is marginally independent of any element in $pa_{\mathcal{G}}(X_{D_i}) \backslash pa_{\mathcal{G}}(X_s)$, implies that any factor containing both X_a and X_b when marginalized over $X_{D_i} \backslash \{X_a, X_b\} \cup X_C$, will factorize as $g(X_a, X_{C_a}, pa_{\mathcal{G}}(X_{D_i}) \backslash X_b) h(X_b, X_{C_b}, pa_{\mathcal{G}}(X_{D_i}))$, where no element in X_{C_a} is adjacent to any element in X_{C_b}. Taking the marginal of $F_i(x_{D_i} \mid pa_{\mathcal{G}}(X_{D_i}))$ with respect to $X_A \cup X_B \cup X_C$ (which is equivalent to evaluating (6) at the maximum values of the marginalized variables) and then conditioning of X_C, will result in a function that factorizes over X_A and X_B, as required. □

To obtain the probability function (5), we calculate each $P_i(X_{D_i} \mid pa_{\mathcal{G}}(X_{D_i}) \backslash X_{D_i})$ by differentiating the corresponding (6) with respect to X_{D_i}. Although this operation, in the discrete case, is in the worst-case exponential in $|X_{D_i}|$, it can be performed efficiently for graphs where $(\mathcal{G})_{\leftrightarrow}$ has low tree-width (Huang and Frey, 2008, Huang et al., 2010).

3.3 The general case: reduction to barren case

We reduce graphs with general districts to graphs with only barren districts by introducing artificial vertices. Create a graph $\mathcal{G}^\star$ with the same vertex set as $\mathcal{G}$ and the same bi-directed edges. For each vertex X_v in $\mathcal{G}$, perform the following operation:

- add an artificial vertex $X_v^\star$ to $\mathcal{G}^\star$;
- add the edge $X_v \rightarrow X_v^\star$ to $\mathcal{G}^\star$, and make the children of $X_v^\star$ to be the original children of X_v in $\mathcal{G}$;
- define the model $P(X_V, X_V^\star)$ to have the same factors (5) as $P(X_V)$, but substituting every occurrence of X_v in $pa_{\mathcal{G}}(X_{D_i})$ by the corresponding $pa_{\mathcal{G}^\star}(X_{D_i})$. Moreover, define $P_v^\star(X_v^\star \mid X_v)$ such that

$$P_v^\star(X_v^\star = x \mid X_v = x) = 1 \tag{7}$$

$$\begin{aligned} P(X_V, X_V^\star) &= \prod_{i=1}^K P_i(X_{D_i} \mid pa_{\mathcal{G}^\star}(X_{D_i}) \backslash X_{D_i}) \\ &\times \prod_{X_v \in X_V} P_v^\star(X_v^\star \mid X_v) \end{aligned} \tag{8}$$

Since the last group of factors is identically equal to 1, they can be dropped from the expression.

From (7), it follows that $P(X_V = x_V, X_V^\star = x_V) = P(X_V = x_V)$. Since no two vertices in the same district can now have a parent-child relation, all districts in $\mathcal{G}^\star$ are barren and as such we can parameterize $P(X_V = x_V, X_V^\star = x_V)$ according to the results of the previous section. A similar trick was exploited by Silva and Ghahramani (2009) to reduce a problem of modeling ADMG probit models to Gaussian models.

Figure 3 provides an example, adapted from Richardson (2009). The graph has a single district containing all vertices. The corresponding transformed graph generates several singleton districts composed of one artificial variable either. In Figure 3(c), we rearrange such districts to illustrate the decomposition described in Section 3.1.

The MCDN formalism inherits the same advantages and limitations of the CDN construction. In particular, parameter constraints analogous to (4) are extended to the conditional case (while Richardson (2009) does not require such constraints[5]), at the advantage of having the number of parameters growing exponentially in the size of the largest bi-directed clique (while Richardson (2009) has the number of parameters growing exponentially in $|X_V|$). With the copula construction introduced in the next Section, the MCDN formulation provides easy support to a variety of families of distributions.

[5] See the Supplementary Material for further examples.

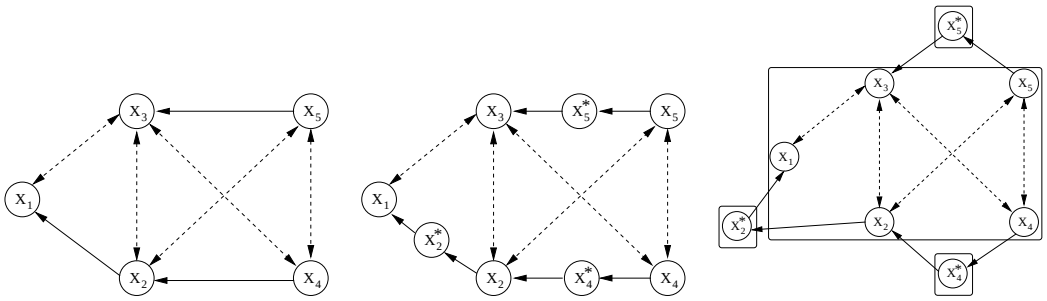


Figure 3: (a) A mixed graph with a single district that includes all five vertices. (b) The modified graph after including artificial vertices (artificial vertices for childless variables are ignored). (c) A display of the four districts of the modified graph in individual boxes. All districts are now barren, i.e., no directed edges can be found within a district.

4 COPULA MCDNS

The main result of Section 3 is that we can parameterize a MCDN model by parameterizing the factors $F_S(x_S|pa_{\mathcal{G}}(X_{D_i})$ in (6) corresponding to each district, which are then put together using the joint model (8). However, we have not yet specified how to construct each factor F_S as introduced in (8). In this section, we describe a particularly convenient way of parameterizing such factors which we call *copula MCDNs*.

Copulas are a flexible approach to defining dependence among a set random variables. This is done by specifying the dependence structure and the marginal distributions separately (Nelsen, 2007) (see also Kirshner (2007) for a machine learning perspective). Simply put, a copula function $C(u_1, \ldots, u_t)$ is just the CDF of a set of dependent random variables, each with a uniform marginal distribution over $[0, 1]$. To define a joint distribution over a set of variables $\{X_v\}$ with arbitrary marginal CDFs $F_v(x_v)$, we simply transform each X_v into a uniform variable u_v over $[0, 1]$ using $u_v \equiv F_v(x_v)$. The resulting joint CDF $F(x_1, \ldots, x_t) = C(F_1(x_1), \ldots, F_t(x_t))$ incorporates both the dependence encoded in C and the marginal distributions F_v.

The motivation for using copulas is two-fold. First, for its flexibility. Second, and arguably the more important advantage in our context, is to be able to easily fulfill the conditions of Proposition 3 that each $X_s \in X_S$ should be independent of any element in $pa_{\mathcal{G}}(X_{D_i}) \backslash pa_{\mathcal{G}}(X_s)$. Before giving the general construction, we first give an example.

Example 2: Let $\mathcal{G}$ be given by

$$\{X_1 \to X_2, X_1 \to X_3, X_2 \to X_4, X_3 \leftrightarrow X_4\}$$

It is necessary to enforce $X_3 \perp\!\!\!\perp X_2 \mid X_1$ while allowing for $X_3 \not\perp\!\!\!\perp X_2 \mid \{X_1, X_4\}$. Fortunately this follows directly from a copula parameterization. Let $F_3(x_3 \mid x_1)$ be a CDF for X_3 conditional on $X_1 = x_1$, analogously for $F_4(x_4 \mid x_2)$. Given a copula $C(u_3, u_4)$, define our joint CDF $F(x_3, x_4 \mid x_1, x_2)$ to be $C(u_3(x_1), u_4(x_2))$ where $u_3 = F_3(x_3 \mid x_1)$ and $u_4 = F_4(x_4 \mid x_2)$. We can see that $X_3 \perp\!\!\!\perp X_2 \mid X_1$, since the marginal CDF of X_3 is $F(x_3, \infty \mid x_1, x_2) = C(u_3, 1) = u_3 = F_3(x_3 \mid x_1)$ which is independent of X_2. The construction also allows for $X_3 \not\perp\!\!\!\perp X_2 \mid \{X_1, X_4\}$, since changing the value of X_2 from x_2 to x_2' might change the value of u_4, and hence allow for $P(x_3 \mid x_1, x_2, x_4) \neq P(x_3 \mid x_1, x_2', x_4)$. □

4.1 Copula construction

Consider the form given by (6) which we wish to parameterize. Since the product of copulas is not necessarily a copula, we cannot simply set each $F_S(x_S \mid pa_{\mathcal{G}}(X_{D_i}))$ to be a copula function. Fortunately, the construction provided by Liebscher (2008) can be adapted to our context. For each clique X_S in $\mathcal{G}_i$ let $C_S(\cdot)$ be a $|S|$-dimensional copula. Let d_v be the number of cliques of $\mathcal{G}_i$ containing variable X_v, and define $a_v \equiv u_v^{1/d_v}$, where $u_v \equiv F_v(x_v \mid pa_{\mathcal{G}}(X_v))$ for some independently parameterized univariate conditional CDF $F_v(x_v \mid pa_{\mathcal{G}}(X_v))$. The modified product of copulas,

$$F_i(x_{D_i} \mid pa_{\mathcal{G}}(X_{D_i})) \equiv \prod_{X_S \in \mathcal{C}_i} C_S(a_S) \qquad (9)$$

where $a_S = \{a_v\}_{v \in S}$ can be shown to be a copula itself (Liebscher, 2008). Moreover, the joint CDF (9) has the form (6) required to be Markov with respect to $\mathcal{G}_i$.

In summary, our parameterization of (6) consists of: a parameterization of each univariate conditional CDF F_v, and a parameterization of a copula C_S for each clique X_S. These parameterizations are variation independent. As a final remark, all required properties still hold if each X_S is a subset of a clique. In our implementation, we define each "clique" to correspond to the pair of vertices linked by a bi-directed edge. This pairwise bi-directed field makes the copula implementation easier, since many copulas are defined for bivariate distributions only (Nelsen, 2007).

5 MODEL DETAILS AND LEARNING

The experiments reported in Section 6 include both discrete and continuous data. In this section we describe the model parameterization used as well as the learning procedure in more detail. For discrete data, the univariate conditional probability function is just a saturated conditional probability table (CPT), as is standard in the Bayesian network literature (Pearl, 1988). For continuous data, we parametrize each univariate conditional density function as a mixture of Gaussian experts (Jacobs et al., 1991):

$$f_v(x_v \mid pa_{\mathcal{G}}(X_v)) = \sum_{z=1}^{K} \pi_{z;v} \mathcal{N}(x_v;\ \mu_{z;v}, \sigma^2_{z;v}) \quad (10)$$

with $\pi_{z;v}$ and $\mu_{z;v}$ depending on $pa_{\mathcal{G}}(X_v)$:

$$\begin{aligned} \mu_{z;v}(pa_{\mathcal{G}}(X_v)) &= \theta_{v0} + \theta_v^{\mathsf{T}} pa_{\mathcal{G}}(X_v) \\ \pi_{z;v}(pa_{\mathcal{G}}(X_v)) &\propto \exp(w_{v0} + w_v^{\mathsf{T}} pa_{\mathcal{G}}(X_v)) \end{aligned} \quad (11)$$

We use the bivariate Frank copula in our implementation:

$$C_F(u_i, u_j; \alpha) = -\frac{1}{\alpha} \ln \left(1 + \frac{(e^{-\alpha u_i} - 1)(e^{-\alpha u_j} - 1)}{e^{-\alpha} - 1} \right)$$

This copula function allows for arbitrarily strong positive or negative associations (Nelsen, 2007).

It is useful to contrast this model against the Gaussian/probit models of Silva and Ghahramani (2009), which is the only Bayesian approach known to us for ADMG parameter learning. Such models can be seen as special cases of the approach described in this paper, using Gaussian copulas only, and Gaussian or probit marginals. Even in the probit case, the bi-directed dependence structure in Silva and Ghahramani (2009) is additive: each X_v is a discretization of an underlying latent variable $X_v^\star = \theta_v^{\mathsf{T}} pa_{\mathcal{G}}(X_v) + \epsilon_v$, where the bi-directed dependency comes from a structured covariance matrix for the error terms ϵ_v, as in the example in the opening Section. Our parameterization does not require such an additive structure.

5.1 Hybrid Bayesian learning

In our experiments we learn the models and make predictions using a framework widely exploited in the copula literature (e.g., Kirshner (2007)): parameters for the marginals are first fit individually and fixed. Given such marginal parameter estimates, copula parameters are then learned. While not as statistically efficient as, say, maximum likelihood estimation, this procedure is still consistent and computationally attractive. An alternative would be a fully Bayesian treatment. Our intention is to validate the usefulness of the parameterization, not to develop a complicated inference method. A simple fully Bayesian approach would be to use Metropolis-Hastings for the univariate parameters jointly. However this is slow computationally and can also be slow to mix as marginal parameters are highly correlated in a way not captured by a naive proposal.

In our context, we will adopt a two-stage Bayesian procedure: first, the univariate conditionals (i.e, the conditional marginals of each district) are individually fit using the posterior expected value estimator. In the continuous case, we calculate the posterior expectations using Gibbs sampling on the mixture of experts. Finally, the estimates of the parametes of the univariate conditionals are treated as if they were the true parameter values. Given such fixed parameters, we then perform MCMC to generate the posterior distribution over copula parameters.

Given the fixed univariate conditionals, we successfully use a standard Metropolis-Hastings algorithm with a random walk proposal to obtain the distribution over copula parameters. Metropolis-Hastings needs the calculation of likelihood ratios: these require transformations of CDFs into probability mass or density functions. While the methods of Huang et al. (2010) could be used, we did a brute-force implementation akin to (3) since, in our experiments, the corresponding districts were no larger than half a dozen variables and brute-force is both simpler and faster. Predictions are performed by using the estimated marginal parameters and by averaging over the samples of copula parameters obtained with the MCMC procedure. For the probit and Gaussian models of Silva and Ghahramani (2009), full Bayesian learning is performed.

Finally we describe the priors used in our experiments. For the discrete CPT parameterization, we use a Dirichlet prior with a pseudo-counts hyperparameter taking the value 0.1. For the mixture of Gaussian experts, each coefficient is given a $\mathcal{N}(0, 5)$ prior, with each conditional variance given a inverse Gamma $(2, 2)$ prior (the data is normalized to have unit variance in the training sets). The number of experts is set at 3 (this worked well; optimizing this number is beyond the scope of this paper). Each Frank copula parameter is given a Gaussian $\mathcal{N}(0, 5)$ prior.

For the Gaussian/probit model of Silva and Ghahramani (2009), each sparse covariance matrix needs a prior, which we set to be $\mathcal{G}$-Inverse Wishart with parameters $(10, \mathbf{I})$, where $\mathbf{I}$ is an identity matrix. We put Gaussian $\mathcal{N}(0, 5)$ priors for the coefficients in the linear model. The probit model also needs thresholds mapping Gaussian variables to discrete variables: thresholds are given factorized Gaussian $\mathcal{N}(0, 5)$ priors constrained to be increasing in value and renormalized.

6 EXPERIMENTS

In this section we evaluate the usefulness of the MCDN parameterization of ADMGs by comparing the predictive performance of copula MCDNs against that of the Gaussian/probit parameterization given in Silva and Ghahra-

Table 1: Characteristics of the data sets used and average log-predictive probabilities per data point of test set data under two different ADMG parameterizations. #V is the number of variables, #D is the total number of data points, $\mathbb{E}[\#\leftrightarrow]$ and $\mathbb{E}[\#\rightarrow]$ are the average number of bidirected and direct edges, respectively, found by MBCS*. The difference between the 10-fold cross validated copula MCDN and Gaussian/probit models' log predictive probabilities and standard errors are given. A star ($\star$) next to results indicates the difference was found to have a median significantly different from zero by the Wilcoxon signed rank test at $p = 0.05$. A more positive difference indicates copula MCDNs predicted the test data better than the Gaussian/probit model.

Data set	Data type	#V	#D	$\mathbb{E}[\#\leftrightarrow]$	$\mathbb{E}[\#\rightarrow]$	Gaussian/probit	Copula MCDN	Difference
SPECT	Binary	23	267	4.1	25.6	-11.32	-11.11	0.21 ± 0.06 $\star$
Breast cancer wisconsin	Ordinal	10	683	5.1	16.3	-12.60	-12.77	-0.17 ± 0.11
Soybean (large)	Ordinal	33	266	9.3	39.8	-20.17	-17.71	2.46 ± 0.20 $\star$
Parkinsons	Continuous	15	5875	8.9	18.2	-11.65	-3.48	8.17 ± 0.28 $\star$
Ionosphere	Continuous	32	351	12.4	32.8	-41.10	-27.45	13.64 ± 0.67 $\star$
Wine quality (red)	Continuous	11	1599	5.7	7.5	-13.72	-11.25	2.47 ± 0.10 $\star$
Wine quality (white)	Continuous	11	4898	7.3	14.5	-13.76	-12.11	1.65 ± 0.09 $\star$

mani (2009). We used seven data sets from the UCI data set repository (Frank and Asuncion, 2010). Three of the data sets have only discrete variables, whilst four have just continuous variables. All discrete variables were removed from the continuous data sets, as was one variable from any pair of variables with a Pearson correlation coefficient greater than 0.95. Statistics are shown in Table 1.

Following preprocessing, we performed 10-fold cross validation on each data set, reporting the test set log predictive probabilities. The training regime is as follows: First, for continuous data, the training and test data were normalized so that the training set has zero mean and unit standard deviation. Then we find a suitable ADMG using the MBCS* algorithm (Pellet, 2008), using the χ^2 test for discrete data, and partial linear correlations for continuous data, both with $p = 0.05$. Finally, parameters for both the copula MCDN and the Gaussian/probit model are estimated as in Section 5. We used the same ADMG in both the copula MCDN and the Gaussian/probit model—our purpose here is to compare parameterizations on real data, not to address the ADMG structure learning problem.

The parameter estimation procedures used are described in Section 5. We used a total of $2,000$ MCMC samples, of which the first 400 formed the burn-in period and were not used for estimating the parameters for prediction. We observed that the MCMC sampler converged within this time by plotting the log likelihood of the training data. We considered increasing the Dirichlet prior hyperparameter to values of 1 and 10, but did not see an improvement to the predictive performance (but the performance was always better than that of the probit model). In future it would be interesting to address the problem of selecting the appropriate amount of smoothing in such discrete models.

Table 1 shows the average log predictive probabilities per test data point, as well as standard errors. As can be seen, the more flexible parameterization afforded by copula MCDNs over the simpler Gaussian and probit models offers significantly better predictions in most cases.

7 CONCLUSION

Acyclic directed mixed graphs are a natural generalization of DAGs. While ADMGs date back at least to Wright (1921), the potential of this framework has only recently being translated into practical applications due to advances into complete parameterizations of Gaussian and discrete networks (Richardson and Spirtes, 2002, Drton and Richardson, 2008, Richardson, 2009). The framework of cumulative distribution networks (Huang and Frey, 2008, Huang and Jojic, 2010) introduced new approaches for flexible parameterizations of bidirected models. In this paper, we extended CDNs to the full ADMG case, introducing the most flexible class of parameterizations of ADMGs to date. We expect that ADMGs will be as readily accessible and as widespread as DAG models in the future.

There are several directions for future work. While classical approaches for learning Markov equivalence classes of ADMGs have been developed by means of multiple hypothesis tests of conditional independencies (Spirtes et al., 2000), a model-based approach based on Bayesian or penalized likelihood functions can deliver more robust learning procedures and a more natural way of combining data with structural prior knowledge. ADMG structures can also play a role in multivariate supervised learning, that is, structured prediction problems. For instance, Silva et al. (2007) introduced some simple models for relational classification inspired by ADMG models and by the link to seemingly unrelated regression (Zellner, 1962). However, efficient ADMG-structured prediction methods and new advanced structural learning procedures will need to be developed.

Acknowledgements

We thank Thomas Richardson from several useful discussions.

References

Bollen, K. (1989). *Structural Equations with Latent Variables*. John Wiley & Sons.

Drton, M. and Richardson, T. (2003). A new algorithm for maximum likelihood estimation in Gaussian models for marginal independence. *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence*.

Drton, M. and Richardson, T. (2008). Binary models for marginal independence. *Journal of the Royal Statistical Society, Series B*, 70:287–309.

Evans, R. and Richardson, T. (2010). Maximum likelihood fitting of acyclic directed mixed graphs to binary data. *Proceedings of the 26th International Conference on Uncertainty in Artificial Intelligence*.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository.

Huang, J. (2009). *Cumulative Distribution Networks: Inference, Estimation and Applications of Graphical Models for Cumulative Distribution Functions*. PhD Thesis, University of Toronto, Department of Computer Science.

Huang, J. and Frey, B. (2008). Cumulative distribution networks and the derivative-sum-product algorithm. *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*.

Huang, J. and Jojic, N. (2010). Maximum-likelihood learning of cumulative distribution functions on graphs. *13th International Conference on Artificial Intelligence and Statistics, AISTATS*.

Huang, J., Jojic, N., and Meek, C. (2010). Exact inference and learning of cumulative distribution functions on loopy graphs. *Neural Information Systems Processing*.

Jacobs, R., Jordan, M., Nowlan, S., and Hinton, G. (1991). Adaptive mixtures of local experts. *Neural Computation*, 3:79–87.

Joe, H. (1997). *Multivariate Models and Dependence Concepts*. Chapman-Hall.

Khare, K. and Rajaratnam, B. (2009). Wishart distributions for covariance graph models. *Technical Report. Department of Statistics, Stanford University*.

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

Lauritzen, S. (1996). *Graphical Models*. Oxford University Press.

Liebscher, E. (2008). Construction of asymmetric multivariate copulas. *Journal of Multivariate Analysis*, 99:2234–2250.

Nelsen, R. (2007). *An Introduction to Copulas*. Springer-Verlag.

Pearl, J. (1988). *Probabilistic Reasoning in Expert Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Pellet, J.-P. (2008). Finding latent causes in causal networks: an efficient approach based on Markov blankets. *Neural Information Processing Systems*.

Richardson, T. (2003). Markov properties for acyclic directed mixed graphs. *Scandinavian Journal of Statistics*, 30:145–157.

Richardson, T. (2009). A factorization criterion for acyclic directed mixed graphs. *Proceedings of the 25th International Conference on Uncertainty in Artificial Intelligence*.

Richardson, T. and Spirtes, P. (2002). Ancestral graph Markov models. *Annals of Statistics*, 30:962–1030.

Rothman, A., Levina, E., and Zhu, J. (2009). Generalized thresholding of large covariance matrices. *Journal of the American Statistical Association*, 104:177–186.

Silva, R., Chu, W., and Ghahramani, Z. (2007). Hidden common cause relations in relational learning. *Neural Information Processing Systems*.

Silva, R. and Ghahramani, Z. (2009). The hidden life of latent variables: Bayesian learning with mixed graph models. *Journal of Machine Learning Research*, 10:1187–1238.

Spirtes, P., Glymour, C., and Scheines, R. (2000). *Causation, Prediction and Search*. Cambridge University Press.

Wright, S. (1921). Correlation and causation. *Journal of Agricultural Research*, pages 557–585.

Zellner, A. (1962). An efficient method of estimating seemingly unrelated regression equations and tests for aggregation bias. *Journal of the American Statistical Association*, 57:348–368.

Zhang, J. (2008). Causal reasoning with ancestral graphs. *Journal of Machine Learning Research*, 9:1437–1474.