Data Mining with Graphical Models

Rudolf Kruse and Christian Borgelt

Dept. of Knowledge Processing and Language Engineering
Otto-von-Guericke-University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, Germany
e-mail: {kruse,borgelt}@iws.cs.uni-magdeburg.de

Abstract. The explosion of data stored in commercial or administrational databases calls for intelligent techniques to discover the patterns hidden in them and thus to exploit all available information. Therefore a new line of research has recently been established, which became known under the names "Data Mining" and "Knowledge Discovery in Databases". In this paper we study a popular technique from its arsenal of methods to do dependency analysis, namely learning inference networks (also called "graphical models") from data. We review the already well-known probabilistic networks and provide an introduction to the recently developed and closely related possibilistic networks.

1 Introduction

Due to the advances in hardware and software technology, large databases (product databases, customer databases, etc.) are nowadays maintained in almost every company and scientific or administrational institution. But often the data is only recorded; evaluation is restricted to simple retrieval and aggregation operations that can be carried out e.g. by SQL queries. It is clear that such operations cannot discover broader structures or general patterns that are present in the data. This, obviously, is a waste of information, since knowing such patterns can give a company a decisive competitive edge. Therefore from recent research a new area called "Data Mining" has emerged, which aims at finding "knowledge nuggets" that are hidden in huge volumes of data. It is the operational core of a process called "Knowledge Discovery in Databases", which (in addition to data mining) comprises data selection, data preprocessing, data transformation, visualization, and result evaluation and documentation [9].

Data mining itself can be characterized best by a set of tasks like classification, clustering (segmentation), prediction, etc. In this paper we focus on dependency analysis, i.e. the task to find dependencies between the attributes that are used to describe a domain of interest. A popular method for this task is the automatic induction of *inference networks*, also called *graphical models* [36, 20], from a set of sample cases.

Graphical models are best known in their probabilistic form, i.e. as Bayesian networks [26] or Markov networks [22]. Efficient implementations of inference systems based on them include HUGIN [1] and PATHFINDER [16]. They are

learned from data by searching for the most appropriate decomposition of the multivariate probability distribution induced by a given dataset [6, 17].

Unfortunately probabilistic graphical models suffer from severe difficulties to deal with imprecise, i.e. set-valued, information in the database to learn from. However, the incorporation of imprecise information is more and more recognized as being indispensable for industrial practice. Therefore graphical models are studied also with respect to other uncertainty calculi, either based on a generalization of the modeling technique to so-called valuation-based networks [30, 31], implemented e.g. in PULCINELLA [28], or based on a specific derivation of possibilistic networks, implemented e.g. in POSSINFER [13, 21]. Recently learning possibilistic networks from data has also been studied [12, 14, 2, 3].

2 Notation and Presuppositions

Notation. Let $V = \{A^{(1)}, \ldots, A^{(m)}\}$ be a finite set of attributes $A^{(k)}$, which are used to describe the section of the world under consideration. We assume the domains $\mathrm{dom}(A^{(k)}) = \{a_1^{(k)}, \ldots, a_{n_k}^{(k)}\}$ of these attributes to be finite sets of categorical values (i.e. we confine to the important case of *discrete graphical models*.) With these presuppositions the reasoning space in which all inferences take place is the joint domain $\Omega = \mathrm{dom}(A^{(1)}) \times \cdots \times \mathrm{dom}(A^{(m)})$, which is sometimes called the *universe of discourse*. Each possible state of the world is described by a tuple $\omega = (a_{i_1}^{(1)}, \ldots, a_{i_m}^{(m)})$ containing the values which the attributes in V assume for this state. For simplicity (and because states with identical describing tuples cannot be distinguished) we identify each $\omega \in \Omega$ with a possible state of the world.

Several times we need to refer to subspaces of Ω and projections of tuples ω to these subspaces. A subspace $\Omega_W \subseteq \Omega$ is the joint domain of a subset $W \subseteq V$ of attributes, i.e. $\Omega_W = \bigtimes_{A \in W} \mathrm{dom}(A)$. A projection of a tuple $\omega \in \Omega$ to this subspace is a tuple $\mathrm{proj}_W^V(\omega) = \omega_W \in \Omega_W$, which contains only the values of the attributes in W.

Presuppositions. Graphical models are concerned with drawing inferences from observations. For a situation in which we are about to draw inferences, we assume that the considered section of the world is in a specific state, whose description $\omega_0 \in \Omega$ we do not know or do not know completely. The inferences to be drawn aim at identifying this state, i.e. at determining the values in ω_0.

To be able to carry out such inferences, *generic knowledge* about dependencies between the values of different attributes must be available. This knowledge is represented as a distribution $\mathcal{D}$ on Ω, which assigns to each tuple $\omega \in \Omega$ a value d_ω, which expresses the probability or (degree of) possibility of the combination of values present in ω. Depending on the values d_ω can have and the interpretation of these values we distinguish between probability and possibility distributions (details are given below). Generic knowledge may be obtained from experts, textbooks, databases etc. Since we are concerned with "data mining", we focus on learning generic knowledge from data.

In addition to generic knowledge we need knowledge to start the inferences from — also called *evidential knowledge* —, which consists in restrictions on the possible values of some of the attributes. This knowledge could be obtained e.g. from observations made about the current state ω_0. From the evidential knowledge about the values of some attributes we infer, using the generic knowledge, restrictions about the values of other attributes, thus narrowing the set of states that have to be considered possible or likely for ω_0.

It is obvious that storing the generic knowledge directly, i.e. the distribution $\mathcal{D}$, would make reasoning very simple, since then we only have to select all $\omega \in \Omega$ compatible with the given evidential knowledge and to combine the corresponding values d_ω appropriately. But, unfortunately, if there are more than only very few attributes, the number of values d_ω to be stored in this case would exceed any reasonable limit. Hence other ways of representing the generic knowledge have to be found. One of them is to use a graphical model, which we discuss in the next section.

3 Graphical Models

In graphical modeling a directed or undirected (hyper)graph is used to represent the generic knowledge about the domain in which the inferences take place. Each vertex corresponds to an attribute, each edge to a dependence between attributes. The edges are the paths along which knowledge about the values of one attribute can be transferred to other attributes, i.e. along which inferences can be drawn. This is understandable, since no information can be transferred from an attribute to another, which is independent of the first.

But even if attributes are dependent, they are sometimes unconnected in a graphical model. The idea underlying this is that an inference need not be direct. If the dependence between two attributes is captured completely by the consecutive dependences on a path connecting the two attributes via other attributes, a direct connection is not necessary. All inferences from one of the attributes to the other can then be carried out along this path.

Conditional independence. Such situations can be characterized by the notion of *conditional independence* [7, 26]. If two attributes get independent, if certain other attributes are fixed, their dependence is not genuine, but only mediated through other attributes. Therefore they need not be connected directly in the graph. Thus the topology of the graph is used to represent an independence model, i.e. a set of conditional independence statements, of the domain under consideration [26].

Of course, not just any notion of conditional independence will do, since, as stated above, the aim is to replace an inference along a direct connection between attributes by an indirect inference. In order to allow such a replacement, the used notion of conditional independence has to satisfy certain axioms, which are known as the *semi-graphoid axioms* [7, 25]. If we denote the independence of a set of attributes X from a set of attributes Y given a set of attributes Z as $X \perp\!\!\!\perp Y \mid Z$, they can be written as

symmetry: $(X \perp\!\!\!\perp Y \mid Z) \Longrightarrow (Y \perp\!\!\!\perp X \mid Z)$

decomposition: $(W \cup X \perp\!\!\!\perp Y \mid Z) \Longrightarrow (W \perp\!\!\!\perp Y \mid Z) \wedge (X \perp\!\!\!\perp Y \mid Z)$

weak union: $(W \cup X \perp\!\!\!\perp Y \mid Z) \Longrightarrow (X \perp\!\!\!\perp Y \mid Z \cup W)$

contraction: $(W \perp\!\!\!\perp Y \mid Z) \wedge (X \perp\!\!\!\perp Y \mid Z \cup W) \Longrightarrow (W \cup X \perp\!\!\!\perp Y \mid Z)$

The *symmetry* axiom states that in any state of knowledge Z, if Y tells us nothing new about X, then X tells us nothing new about Y. The *decomposition* axiom asserts that if two combined items of information are judged irrelevant to X, then each separate item is irrelevant as well. The *weak union* axiom states that learning irrelevant information W cannot help the irrelevant information Y become relevant to X. The *contraction* axiom states that if we judge W irrelevant to X after learning some irrelevant information Y, then W must have been irrelevant before we learned Y. Together the weak union and contraction properties mean that irrelevant information should not alter the relevance of other propositions in the system; what was relevant remains relevant, and what was irrelevant remains irrelevant [26]. It is plausible that any reasonable notion of conditional independence should satisfy these axioms.

Independence graphs. Given an appropriate notion of conditional independence an *independence graph* can be defined. In such a graph the *conditional independence* of two attribute sets given a third is expressed by *separation* of the corresponding node sets by the nodes corresponding to the third set.

What is to be understood by "separation" depends on whether the graph is directed or undirected. If it is undirected, separation is defined as follows: If X, Y, and Z are three disjoint subsets of nodes, then Z separates X from Y, iff after removing the nodes in Z and their associated edges from the graph there is no path, i.e. no sequence of consecutive edges, from a node in X to a node in Y. Or, in other words, Z separates X from Y, iff all paths from a node in X to a node in Y contain a node in Z.

For directed graphs, which have to be acyclic, the so-called *d-separation criterion* is used [26]: If X, Y, and Z are three disjoint subsets of nodes, then Z is said to *d-separate* X from Y, iff there is no path, i.e. no sequence of consecutive edges (of any directionality), from a node in X to a node in Y along which the following two conditions hold:

1. every node with converging edges either is in Z or has a descendant in Z,
2. every other node is not in Z.

With these notions we can define the *Markov properties* of graphs [36]:

pairwise: Attributes, whose nodes are non-adjacent in the graph, are independent conditional on all remaining attributes.

local: Conditional on the attributes corresponding to the adjacent nodes, an attribute is independent of all remaining attributes.

global: Any two subsets of attributes, whose corresponding node sets are separated by a third node set, are independent conditionally only on the attributes corresponding to the nodes in the third set.

Note that the local Markov property is contained in the global, and the pairwise Markov property in the local.

Since the pairwise Markov property refers to the independence of only two attributes, it would be most natural (at least for undirected graphs) to use it to define an independence graph: If two attributes are dependent given all other attributes, there is an edge between their corresponding nodes, otherwise there is no edge [36]. But, unfortunately, the three types of Markov properties are not equivalent in general, and it is obvious that we need the *global* Markov property for inferences from multiple observations. However, the above definition can be used, if — in addition to the semi-graphoid axioms — the following axiom holds:

$$\text{intersection:} \qquad (W \perp\!\!\!\perp Y \mid Z \cup X) \wedge (X \perp\!\!\!\perp Y \mid Z \cup W) \Longrightarrow (W \cup X \perp\!\!\!\perp Y \mid Z)$$

The semi-graphoid axioms together with this one are called the *graphoid axioms* [7, 25]. If they hold for a given notion of conditional independence, an independence graph can be defined via the pairwise Markov condition, since the intersection axiom allows us to infer the global Markov property from the pairwise. If the intersection axiom does not hold, the global Markov property has to be used to define an independence graph.

It is obvious that an independence graph for a given domain is easy to find. For example, the complete undirected graph, i.e. the graph in which every node is connected directly to every other, always is an independence graph. But a complete graph would not reduce the amount of data to be stored (see below). Therefore, in graphical modeling, we have to add the condition that the independence graph has to be *sparse* or even *minimal*, i.e. should contain as few edges as possible.

Note that directed acyclic graphs and undirected graphs represent conditional independence relations in fundamentally different ways. In particular, there are undirected graphs that represent a conditional independence that cannot be represented by a single directed acyclic graph, and vice versa.

The quantitative part of a graphical model. The independence graph is also called the *qualitative* part of a graphical model, since it specifies which attributes are dependent and which are independent, but not the details of the dependences. How the latter information, which is called the *quantitative* part of a graphical model, is described, depends again on the type of the graph. In a directed acyclic graph, it is represented as a set of conditional distributions: one for each attribute conditional on all of its parents in the graph. If an attribute does not have any parents, its associated distribution simplifies to an unconditional distribution.

For an undirected graph, the quantitative part is represented as a set of marginal distributions: one for each maximal clique of the independence graph, where a maximal clique is a fully connected subgraph that is not contained in another fully connected subgraph. Because of this representation an undirected *hyper*graph is often used instead of a normal undirected graph. The nodes of each maximal clique of the normal graph are then connected by one *hyper*edge in the hypergraph. Unfortunately, this approach suffers from the fact that the resulting hypergraph can have cycles. This causes problems, because during an inference process the same information can travel along more than one path and thus may

be used several times to update the knowledge about an attribute. If the inference mechanism is not idempotent, i.e. if a second incorporation of already included information changes the result, this can invalidate the conclusions drawn.

In order to avoid these problems, the discussion is usually restricted to *triangulated* undirected graphs, i.e. to graphs in which each cycle of length four or larger contains a *chord*, where a chord is an edge between two non-consecutive nodes in the cycle. It can be shown that the maximal clique hypergraph of a triangulated undirected graph is always a *hypertree*, i.e. does not contain any cycles. In addition, this type of graphs is important, because it can be shown that a triangulated undirected graph is isomorphic to a directed acyclic graph. Thus, with the restriction to triangulated graphs, the difference between directed and undirected graphs vanishes.

It is worth noting that especially the representation using undirected graphs suggests to view graphical modeling as a decomposition method: The (global) distribution $\mathcal{D}$ is decomposed into a set of (local) distributions $\{\mathcal{D}_{X_1}, \ldots, \mathcal{D}_{X_n}\}$ on subspaces, which are the cross-products of the domains of the attributes in a maximal clique. Because of this decomposition, global reasoning, i.e. drawing inferences using $\mathcal{D}$, can be replaced by local reasoning, which involves only the distributions $\mathcal{D}_{X_k}$.

Reasoning in graphical models. The reasoning process, which we describe here exemplary for an undirected graph, basically is this: Information obtained e.g. by observations about the values of an attribute is extended to the distributions on all hyperedges containing the attribute and then projected to the intersections of these hyperedges with other hyperedges. From there it is extended and projected again etc. until the information is distributed to all attributes.

A general local propagation algorithm for hypertrees has been developed for so-called *valuation-based systems* [30]. The axiomatic framework of a valuation-based system [32] can represent various uncertainty calculi such as probability theory, Dempster-Shafer theory, and possibility theory.

Learning graphical models from data. When we consider learning graphical models from data, problems arise from the fact that various kinds of prior information can be available, expert knowledge as well as a database of sample cases, both of which should be considered in a unified framework. However, since our focus is on "data mining", we restrict ourselves to a purely data-oriented approach, i.e. we assume only a database of observations to be given.

Since constructive methods are rarely available, data oriented learning methods nearly always consist of two parts: a search method and an evaluation measure. The evaluation measure estimates the quality of a given (hyper)graph and the search method governs which (hyper)graphs are inspected. Often the search is guided by the value of the evaluation measure, since it is usually the goal to maximize (or to minimize) its value. Commonly used search methods include optimum weight spanning tree construction [5] (for undirected graphs) and greedy parent selection [6] (for directed graphs). Evaluation measures depend on the underlying uncertainty calculus and are considered below.

4 Probabilistic Graphical Models

In purely probabilistic approaches quantitative knowledge about the dependencies between the attributes in V is described by a probability distribution P on Ω. $P(\omega) = p \in [0,1]$ means that the combination of attribute values in ω has the probability p. A conditional probability distribution is defined in the usual way, i.e. as

$$P(\omega_X \mid \omega_Y) = \frac{P(\omega_{X \cup Y})}{P(\omega_Y)}.$$

Conditional independence is defined in accordance with the usual notion of stochastic independence as follows: Let X, Y, and Z be three disjoint subsets of attributes in V. X is called *conditionally independent* of Y given Z w.r.t. P, abbreviated $X \perp\!\!\!\perp_P Y \mid Z$, iff

$$\forall \omega \in \Omega: \quad P(\omega_{X \cup Y} \mid \omega_Z) = P(\omega_X \mid \omega_Z) \cdot P(\omega_Y \mid \omega_Z)$$

whenever $P(\omega_Z) > 0$.

Bayesian networks. The most popular kind of probabilistic graphical models in artificial intelligence is the *Bayesian network*, also called *belief network* [26]. A Bayesian network consists of a directed acyclic graph and a set of conditional probability distributions $P(\omega_A \mid \omega_{\text{parents}(A)})$, $A \in V$, where parents(A) is the set of attributes corresponding to the parents of the attribute A in the graph.

A Bayesian network describes a decomposition of a joint probability distribution P on Ω into a set of conditional probability distributions: A strictly positive probability distribution P on Ω *factorizes* w.r.t. a directed acyclic graph, if

$$\forall \omega \in \Omega: \quad P(\omega) = \prod_{A \in V} P(\omega_A \mid \omega_{\text{parents}(A)}).$$

In this case P satisfies the *global Markov property* (cf. section 3). It follows, that a Bayesian network can be seen as a graphical representation of a Markov chain.

Since a Bayesian network is a directed graph, it is well-suited to represent direct causal dependencies between variables. In many cases this is quite natural for knowledge representation, e.g. in expert systems designed for diagnostic reasoning (abductive inference) in medical applications.

Markov networks. An alternative type of probabilistic graphical models uses undirected graphs and is called a *Markov network* [22]. Similar to a Bayesian network it describes a decomposition of the joint probability distribution P on Ω, but it uses a *potential representation*: A strictly positive probability distribution P on Ω *factorizes* w.r.t. an undirected graph, if

$$\forall X \in \text{cliques}(G): \exists \phi_X: \forall \omega \in \Omega: \quad P(\omega) = \prod_{X \in \text{cliques}(G)} \phi_X(\omega_X),$$

where cliques(G) is the set of all maximal cliques, each of which is represented by the set of attributes whose corresponding nodes are contained in it. The ϕ_X are strictly positive functions defined on Ω_X, $X \subseteq V$.

Learning probabilistic networks from data. When learning probabilistic networks from data, we have to distinguish between quantitative and qualitative network induction.

Quantitative network induction for a given network structure consists in estimating the joint probability distribution P, where P is selected from a family of parameterized probability distributions. A lot of approaches have been developed in this field, using methods such as maximum likelihood, maximum penalized likelihood, or fully Bayesian approaches, which involve different computational techniques of probabilistic inference such as the expectation maximization (EM) algorithm, Gibbs sampling, Laplace approximation, and Monte Carlo methods. For an overview, see e.g. [34].

Qualitative network induction consists in learning a network structure from a database of sample cases. In principle one could use the factorization property of a probabilistic network to evaluate its quality by comparing for each $\omega \in \Omega$ the probability computed from the network with the relative frequency found in the database to learn from. But this approach is usually much too costly. Other methods include the extensive testing of conditional independences (CI tests) [35] and a Bayesian approach [6]. Unfortunately, CI tests tend to be unreliable unless the volume of data is enormous, and with an increasing number of vertices they soon become computationally intractable. Bayesian learning requires debatable prior assumptions (for example, default uniform priors on distributions, uniform priors on the possible graphs) and also tends to be inefficient unless greedy search methods are used. Nevertheless, several network induction algorithms have successfully been applied. The oldest example is an algorithm to decompose a multi-variate probability distribution into a tree of two-dimensional distributions [5]. It uses mutual information as the evaluation measure and optimum weight spanning tree construction as the search method. Another example is the $K2$ algorithm [6], which uses a greedy parent search and a Bayesian evaluation measure. Its drawback, which consists in the fact that it needs a topological order of the attributes, can be overcome by a hybrid algorithm [33], which combines CI tests (to find a topological order) and $K2$ (to construct the Bayesian network with respect to this topological order). Several evaluation measures, which can be used with optimum weight spanning tree construction and greedy parent search as well as other search methods, are surveyed in [2, 3].

5 Possibilistic Graphical Models

Possibility distributions. A *possibility distribution* π on a universe of discourse Ω is a mapping from Ω into the unit interval, i.e. $\pi : \Omega \to [0, 1]$ [38, 8]. From an intuitive point of view, $\pi(\omega)$ quantifies the degree of possibility that $\omega = \omega_0$ is true, where ω_0 is the actual state of the world (cf. section 2): $\pi(\omega) = 0$ means that $\omega = \omega_0$ is impossible, $\pi(\omega) = 1$ means that $\omega = \omega_0$ is possible without any restrictions, and $\pi(\omega) \in (0, 1)$ means that $\omega = \omega_0$ is possible only with restrictions, i.e. that there is evidence that supports $\omega = \omega_0$ as well as evidence that contradicts $\omega = \omega_0$.

Several suggestions have been made for semantics of a *theory of possibility* as a framework for reasoning with uncertain and imprecise data. The interpretation of a degree of possibility we prefer is based on the context model [11, 21]. In this model possibility distributions are seen as *information-compressed* representations of (not necessarily nested) random sets and a degree of possibility as the one-point coverage of a random set [23].

To be more precise: Let ω_0 be the actual, but unknown state of a domain of interest, which is contained in a set Ω of possible states. Let $(C, 2^C, P)$, $C = \{c_1, c_2, \ldots, c_m\}$, be a finite probability space and $\gamma : C \to 2^\Omega$ a set-valued mapping. C is seen as a set of contexts that have to be distinguished for a set-valued specification of ω_0. The contexts are supposed to describe different physical and observation-related frame conditions. $P(\{c\})$ is the (subjective) probability of the (occurrence or selection of the) context c.

A set $\gamma(c)$ is assumed to be the *most specific correct set-valued specification* of ω_0, which is implied by the frame conditions that characterize the context c. By "most specific set-valued specification" we mean that $\omega_0 \in \gamma(c)$ is guaranteed to be true for $\gamma(c)$, but is not guaranteed for any proper subset of $\gamma(c)$. The resulting *random set* $\Gamma = (\gamma, P)$ is an imperfect (i.e. imprecise *and* uncertain) specification of ω_0. Let π_Γ denote the *one-point coverage of* Γ (the *possibility distribution induced by* Γ*)*, which is defined as

$$\pi_\Gamma : \Omega \to [0, 1], \quad \pi_\Gamma(\omega) = P\left(\{c \in C \mid \omega \in \gamma(c)\}\right).$$

In a complete model the contexts in C must be specified in detail to make the relationships between all contexts c_j and their corresponding specifications $\gamma(c_j)$ explicit. But if the contexts are unknown or ignored, then $\pi_\Gamma(\omega)$ is the total mass of all contexts c that provide a specification $\gamma(c)$ in which ω_0 is contained, and this quantifies the *possibility of truth* of the statement "$\omega = \omega_0$" [11, 13].

That in this interpretation a possibility distribution represents uncertain *and* imprecise knowledge can be understood best by comparing it to a probability distribution and to a relation. A probability distribution covers *uncertain*, but *precise* knowledge. This becomes obvious, if one notices that a possibility distribution in the interpretation described above reduces to a probability distribution, if $\forall c_j \in C : |\gamma(c_j)| = 1$, i.e. if for all contexts the specification of ω_0 is precise. On the other hand, a relation represents *imprecise*, but *certain* knowledge about dependencies between attributes. Thus, not surprisingly, a relation can also be seen as a special case of a possibility distribution, namely if there is only one context. Hence the context-dependent specifications are responsible for the imprecision, the contexts for the uncertainty in the imperfect knowledge expressed by a possibility distribution.

Possibilistic networks. Although well-known for a couple of years [18], a unique concept of possibilistic independence has not been fixed yet. In our opinion, the problem is that possibility theory is a calculus for uncertain *and* imprecise reasoning, the first of which is related to probability theory, the latter to relational theory (see above). But these two theories employ different notions of independence, namely stochastic independence and lossless join decomposability.

Stochastic independence is an *uncertainty-based* type of independence, whereas lossless join decomposability is an *imprecision-based* type of independence. Since possibility theory addresses both kinds of imperfect knowledge, notions of possibilistic independence can be uncertainty-based or imprecision-based.

W.r.t. this consideration two definitions of possibilistic independence have been justified [4], namely uncertainty-based possibilistic independence, which is derived from *Dempster's rule of conditioning* [29] adapted to possibility measures, and imprecision-based possibilistic independence, which coincides with the well-known concept of *possibilistic non-interactivity* [8]. The latter can be seen as a generalization of lossless join decomposability to the possibilistic setting, since it treats each α-cut of a possibility distribution like a relation.

Because of its consistency with the *extension principle* [37], we confine to possibilistic non-interactivity. As a concept of possibilistic independence it can be defined as follows: Let X, Y, and Z be three disjoint subsets of variables in V. Then X is called *conditionally independent* of Y given Z w.r.t. π, abbreviated $X \perp\!\!\!\perp_\pi Y \mid Z$, iff

$$\forall \omega \in \Omega: \quad \pi(\omega_{X \cup Y} \mid \omega_Z) = \min\{\pi(\omega_X \mid \omega_Z), \pi(\omega_Y \mid \omega_Z)\}$$

whenever $\pi(\omega_Z) > 0$, where $\pi(\cdot \mid \cdot)$ is a non-normalized conditional possibility distribution, i.e.

$$\pi(\omega_X \mid \omega_Z) = \max\{\pi(\omega') \mid \omega' \in \Omega \wedge \mathrm{proj}_X^V(\omega) = \omega_X \wedge \mathrm{proj}_Z^V(\omega) = \omega_Z\}.$$

Both mentioned types of possibilistic independence satisfy the *semi-graphoid axioms* (see section 3). Possibilistic independence based on Dempster's rule in addition satisfies the intersection axiom and thus can be used within the framework of the valuation-based systems already mentioned above [30]. However, the intersection axiom is related to uncertainty-based independence. Relational independence does not satisfy this axiom, and therefore it cannot be satisfied by possibilistic non-interactivity as a more general type of imprecision-based independence.

Similar to probabilistic networks, a possibilistic network can be seen as a decomposition of a multi-variate possibility distribution. The factorization formulae can be derived from the corresponding probabilistic factorization formulae (for Markov networks) by replacing the product by the minimum.

Learning possibilistic networks from data. Just as for probabilistic networks, it is possible in principle to estimate the quality of a given possibilistic network by exploiting its factorization property. For each $\omega \in \Omega$ the degree of possibility computed from the network is compared to the degree of possibility derived from the database to learn from. But again this approach can be costly.

Contrary to probabilistic networks, the induction of possibilistic networks from data has been studied much less extensively. A first result, which consists in an algorithm that is closely related to the $K2$ algorithm for the induction of Bayesian networks, was presented in [12]. Instead of the Bayesian evaluation measure used in $K2$, it relies on a measure derived from the *nonspecificity* of a

possibility distribution. Roughly speaking, the notion of nonspecificity plays the same role in possibility theory that the notion of *entropy* plays in probability theory. Based on the connection of the imprecision part of a possibility distribution to relations, the nonspecificity of a possibility distribution can also be seen as a generalization of *Hartley information* [15] to the possibilistic setting.

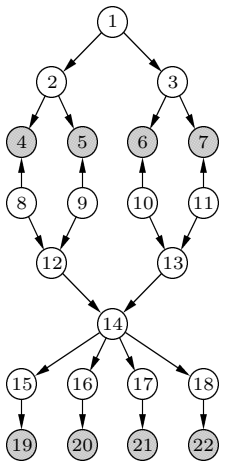
In [14] a rigid foundation of a learning algorithm for possibilistic networks is given. It starts from a comparison of the nonspecificity of a given multi-variate possibility distribution to the distribution represented by a possibilistic network, thus measuring the loss of specificity, if the multi-variate possibility distribution is represented by the network. In order to arrive at an efficient algorithm, an approximation for this loss of specificity is derived, which can be computed locally on the hyperedges of the network. As the search method a generalization of the optimum weight spanning tree algorithm to hypergraphs is used. Several other heuristic local evaluation measures, which can be used with different search methods, are presented in [2, 3].

It should be emphasized, that, as already discussed above, an essential advantage of possibilistic networks over probabilistic ones is their ability to deal with imprecision, i.e. multi-valued, information. When learning possibilistic networks from data, this leads to the convenient situation that missing values in an observation or a set of values for an attribute, all of which have to be considered possible, do not pose any problems.

6 Application

Although a good theory may be the most practical thing to have, all theory must hold its own in a test against reality. As a test case we chose the Danish Jersey cattle blood group determination problem [27], for which a Bayesian network designed by domain experts (cf. figure 1) and a database of 500 real world sample cases exists (an extract of this database is shown in table 1). The problem with this database is that it contains a pretty large number of unknown values — only a little over half of the tuples are complete (This can already be guessed from the extract shown in table 1: the stars denote missing values).

As already indicated above, missing values can make it difficult to learn a Bayesian network, since an unknown value can be seen as representing imprecise information: It states that all values contained in the domain of the corresponding attribute are possible. Nevertheless it is still feasible to learn a Bayesian network — similar to the expert designed one — from the database in this case, since the dependencies are rather strong and thus the remaining small number of tuples is still sufficient to recover the underlying structure. However, learning a possibilistic network from the same dataset is much easier, since possibility theory was especially designed to handle imprecise information. Hence no discarding or special treatment of tuples is necessary. An evaluation of the learned network showed that it was of comparable quality. Thus we can conclude that learning possibilistic networks from data is an important alternative to the established probabilistic methods.



1 – parental error
2 – dam correct?
3 – sire correct?
4 – stated dam ph.gr. 1
5 – stated dam ph.gr. 2
6 – stated sire ph.gr. 1
7 – stated sire ph.gr. 2
8 – true dam ph.gr. 1
9 – true dam ph.gr. 2
10 – true sire ph.gr. 1
11 – true sire ph.gr. 2
12 – offspring ph.gr. 1
13 – offspring ph.gr. 2
14 – offspring genotype
15 – factor 40
16 – factor 41
17 – factor 42
18 – factor 43
19 – lysis 40
20 – lysis 41
21 – lysis 42
22 – lysis 43

The grey nodes correspond to observable attributes. Node 1 can be removed to simplify constructing the clique tree for propagation.

Fig. 1. Domain expert designed network for the Danish Jersey cattle blood type determination example

n	y	y	f1	v2	f1	v2	f1	v2	f1	v2	v2	v2	v2v2	n	y	n	y	0	6	0	6
n	y	y	f1	v2	**	**	f1	v2	**	**	**	**	f1v2	y	y	n	y	7	6	0	7
n	y	y	f1	v2	f1	f1	f1	v2	f1	f1	f1	f1	f1f1	y	y	n	n	7	7	0	0
n	y	y	f1	v2	f1	f1	f1	v2	f1	f1	f1	f1	f1f1	y	y	n	n	7	7	0	0
n	y	y	f1	v2	f1	v1	f1	v2	f1	v1	v2	f1	f1v2	y	y	n	y	7	7	0	7
n	y	y	f1	f1	**	**	f1	f1	**	**	f1	f1	f1f1	y	y	n	n	6	6	0	0
n	y	y	f1	v1	**	**	f1	v1	**	**	v1	v2	v1v2	n	y	y	y	0	5	4	5
n	y	y	f1	v2	f1	v1	f1	v2	f1	v1	f1	v1	f1v1	y	y	y	y	7	7	6	7

Table 1. An extract from the Danish Jersey cattle blood group determination database.

7 Conclusions

In this paper we reviewed, although briefly, the ideas underlying probabilistic networks and provided an equally brief introduction to possibilistic networks. The main advantage of the latter is that they can handle directly imprecise, i.e. set-valued, information. This is especially useful, if an inference network is to be learned from data and the database to learn from contains a considerable amount of missing values. Whereas in order to learn a probabilistic network these tuples have to be discarded or treated in some complicated manner, possibilistic network learning can easily take them into account and can thus, without problem, make use of all available information. These considerations proved to be well-founded in an application on a real-world database.

References

1. S.K. Andersen, K.G. Olesen, F.V. Jensen, F. and Jensen. HUGIN — a shell for building Bayesian belief universes for expert systems. *Proc. 11th International Joint Conference on Artificial Intelligence*, pp. 1080–1085, 1989
2. C. Borgelt and R. Kruse. Evaluation measures for learning probabilistic and possibilistic networks. *Proc. 6th IEEE Int. Conf. on Fuzzy Systems*, pp. 669–676, Barcelona, Spain, 1997
3. C. Borgelt and R. Kruse. Some Experimental Results on Learning Probabilistic and Possibilistic Networks with Different Evaluation Measures. *Proc. 1st Int. J. Conf. on Qualitative and Quantitative Practical Reasoning, ECSQARU-FAPR'97*, 71–85, Bad Honnef, Germany, 1997
4. L.M. de Campos, J. Gebhardt, and R. Kruse. *Syntactic and semantic approaches to possibilistic independence.* Technical report, University of Granada, Spain, and University of Braunschweig, Germany, 1995
5. C.K. Chow and C.N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Trans. on Information Theory* **14**(3) 462–467, 1968
6. G. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning* **9**:309–347, 1992
7. A. Dawid. Conditional independence in statistical theory. *SIAM Journal on Computing* **41**:1–31, 1979
8. D. Dubois and H. Prade. *Possibility Theory.* Plenum Press, New York, NY, 1988
9. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, eds. *Advances in Knowledge Discovery and Data Mining.* AAAI Press / MIT Press, Cambridge, MA, 1996
10. J. Gebhardt and R. Kruse. A new approach to semantic aspects of possibilistic reasoning. In: M. Clarke, R. Kruse, and S. Moral, eds. *Symbolic and Quantitative Approaches to Reasoning and Uncertainty* (Lecture Notes in Computer Science 747), pp. 151–160, Springer, Berlin, Germany, 1993
11. J. Gebhardt and R. Kruse. The context model — an integrating view of vagueness and uncertainty. *Int. Journal of Approximate Reasoning* **9**:283–314, 1993
12. J. Gebhardt and R. Kruse. Learning possibilistic networks from data. *Proc. 5th Int. Workshop on Artificial Intelligence and Statistics*, pp. 233–244, Fort Lauderdale, FL, 1995
13. J. Gebhardt and R. Kruse. POSSINFER — A software tool for possibilistic inference. In: D. Dubois, H. Prade, and R. Yager, eds. *Fuzzy Set Methods in Information Engineering: A Guided Tour of Applications*, pp. 407–418, Wiley, New York, NY, 1996
14. J. Gebhardt and R. Kruse. Tightest hypertree decompositions of multivariate possibility distributions. *Proc. Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU'96)*, pp. 923–927, Granada, Spain, 1996
15. R.V.L. Hartley. Transmission of information. *The Bell Systems Technical Journal* **7**:535–563, 1928
16. D. Heckerman. *Probabilistic Similarity Networks.* MIT Press, Cambridge, MA, 1991
17. D. Heckerman, D. Geiger, and D.M. Chickering. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning* 20:197–243, Kluwer, Dordrecht, Netherlands, 1995

18. E. Hisdal. Conditional possibilities, independence, and noninteraction. *Fuzzy Sets and Systems* **1**:283–297, 1978
19. F.V. Jensen and J. Liang. drHUGIN — a system for value of information in Bayesian networks. *Proc. 5th Int. Conf. on Information Processing and Management of Uncertainty in knowledge-based Systems*, 1994
20. R. Kruse, E. Schwecke, and J. Heinsohn. *Uncertainty and Vagueness in Knowledge-based Systems: Numerical Methods.* Series: Artificial Intelligence, Springer, Berlin, Germany, 1991
21. R. Kruse, J. Gebhardt, and F. Klawonn. *Foundations of Fuzzy Systems.* Wiley, Chichester, England, 1994 Translation of the book: *Fuzzy Systeme (Series: Leitfäden und Monographien der Informatik).* Teubner, Stuttgart, Germany, 1994
22. S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Stat. Soc., Series B* **2**(50):157–224, 1988
23. H.T. Nguyen. Using Random Sets. *Information Science* 34:265–274, 1984
24. J. Pearl. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence* **29**:241–288, 1986
25. J. Pearl and A. Paz. Graphoids: a graph based logic for reasoning about relevance relations. In: B.D. Boulay et. al, eds. *Advances in Artificial Intelligence 2*, pp. 357–363 North Holland, Amsterdam, Netherlands, 1987
26. J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference (2nd edition).* Morgan Kaufmann, San Mateo, CA, 1992
27. L.K. Rasmussen. *Blood Group Determination of Danish Jersey Cattle in the F-blood Group System.* Dina Research Report no. 8, 1992
28. A. Saffiotti and E. Umkehrer. PULCINELLA: a general tool for propagating uncertainty in valuation networks. *Proc. 7th Conf. on Uncertainty in Artificial Intelligence*, pp. 323–331, Morgan Kaufmann, San Mateo, CA, 1991
29. G. Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, NJ, 1976
30. G. Shafer and P.P. Shenoy. *Local computations in hypertrees*, Working paper 201. School of Business, University of Kansas, Lawrence, KS, 1988
31. P.P. Shenoy. *Conditional independence in valuation-based systems*, Working Paper 236, School of Business, University of Kansas, Lawrence, KS, 1991
32. P.P. Shenoy. Valuation-based systems: a framework for managing uncertainty in expert systems. In: L.A. Zadeh and J. Kacprzyk, eds. *Fuzzy Logic for the Management of Uncertainty*, pp. 83–104, Wiley, New York, NY, 1992
33. M. Singh and M. Valtorta. An algorithm for the construction of Bayesian network structures from data. *Proc. 9th Conf. on Uncertainty in Artificial Intelligence* Washington, pp. 259–265, 1993
34. D. Spiegelhalter, A. Dawid, S. Lauritzen, and R. Cowell. Bayesian analysis in expert systems. *Statistical Science* **8**(3):219–283, 1993
35. T.S. Verma and J. Pearl. An algorithm for deciding if a set of observed independencies has a causal explanation. *Proc. 8th Conf. on Uncertainty in Artificial Intelligence* pp. 323–330, 1992
36. J. Whittaker. *Graphical Models in Applied Multivariate Statistics.* Wiley, Chichester, England, 1990
37. L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences* **9**:43–80, 1975
38. L.A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems* **1**:3–28, 1978