

Evaluating product-based possibilistic networks learning algorithms

Nahla Ben Amor, Maroua Haddad, Philippe Leray

▶ **To cite this version:**

Nahla Ben Amor, Maroua Haddad, Philippe Leray. Evaluating product-based possibilistic networks learning algorithms. 13th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2015), 2015, Compiègne, France. 10.1007/978-3-319-20807-7_28 . hal-01150813

HAL Id: hal-01150813
https://hal.science/hal-01150813

Submitted on 15 Apr 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Evaluating product-based possibilistic networks learning algorithms

Maroua Haddad[1,2], Philippe Leray[2], and Nahla Ben Amor[1]

[1] LARODEC Laboratory ISG, Université de Tunis, Tunisia.
maroua.haddad@gmail.com, nahla.benamor@gmx.fr
[2] LINA-UMR CNRS 6241, University of Nantes, France.
philippe.leray@univ-nantes.fr

Abstract. This paper proposes a new evaluation strategy for *product-based* possibilistic networks learning algorithms. The proposed strategy is mainly based on sampling a possibilistic networks in order to construct an imprecise data set representative of their underlying joint distribution. Experimental results showing the efficiency of the proposed method in comparing existing possibilistic networks learning algorithms is also presented.

1 Introduction

Researches devoted to graphical models handle a classical form of data which consists in precise information and at most handle missing data. This is due to the fact that most of these works are defined in the probabilistic framework which represents a well-founded normative framework for knowledge representation and reasoning with uncertain but *precise* data. However, in real world applications, we are often faced to more sophisticated imperfect data. In such situation, probability theory does not remain the adequate framework, hence, the birth of several other uncertainty theories such as the case of possibility theory [1]. Consequently, alternative graphical models have been proposed to reason with this form of imperfect data such as possibilistic networks. Despite the multitude of research endeavors concerning propagation in possibilistic networks, e.g. [2, 3], the problem of learning such networks from data remains very limited. Moreover, existing methods [4, 5, 6] do not propose a convincing evaluation process since most of them has been limited by the lack of an accurate and standard validation procedure. This paper proposes a new evaluation strategy for *product-based* possibilistic networks learning algorithms based on sampling. Such an approach is commonly used for probabilistic graphical models and especially in the evaluation of Bayesian networks learning algorithms, but, it raises several difficulties when applied to possibilistic networks as it will be detailed in this paper. This paper is organized as follows: Section 2 gives a brief introduction to possibility theory. Section 3 defines possibilistic networks and discusses their learning from data. Section 4 details our proposed evaluation strategy to possibilistic networks learning algorithms. Section 5 is dedicated to the experimental results.

2 Brief recall on possibility theory

This section recalls elementary notions of possibilisty theory [7] and points out the notion of possibility distribution estimation. Let $V = \{X_1, ..., X_n\}$ be a set of variables such that $D_1, ..., D_n$ are their respective domains and let x_{ik} be an instance of X_i, i.e. each $x_{ik} \in D_i$ corresponds to a state (a possible value) of X_i. The agents knowledge (state set) of X_i can be encoded by a possibility distribution $\pi(X_i)$ corresponding to a mapping from the universe of discourse D_i to the unit interval [0,1]. For any state $x_{ik} \in D_i$, $\pi(x_{ik}) = 1$ means that x_{ik} realization is totally possible $\pi(x_{ik}) = 0$ means that x_{ik} is an impossible state. It is generally assumed that at least one state x_{ik} is totally possible and π is then said to be normalized. Extreme cases of knowledge are presented by *complete knowledge*, i.e. $\exists x_{ik} \in D_i$ s.t. $\pi(x_{ik}) = 1$ and $\forall x_{ij} \in D_i$ s.t. $x_{ij} \neq x_{ik}, \pi(x_{ij}) = 0$ and *total ignorance*, i.e. $\forall x_{ik} \in D_i, \pi(x_{ik}) = 1$ (all values in D_i are possible). The definition of a possibility distribution could be generalized to a set of variables V defined on the universe of discourse $\Omega = D_1 \times ... \times D_n$ encoded by π. π corresponds to a mapping from Ω to the unit interval [0,1]. ω is called interpretation or event and is denoted by a tuple $(x_{1k}, ..., x_{nl})$. $\omega[X_i]$ is the value of X_i in ω.

Possibility theory is based on minimum non-specificity principle. More precisely, let π and π' be two possibility distributions, π is said to be more specific (more informative) than π' iff $\forall x_{ik} \in D_i, \pi(x_{ik}) \leq \pi'(x_{ik})$. Given a possibility distribution π, we can define for any subset $A \subseteq D_i$ two dual measures: possibility measure $\Pi(A) = \max_{x_{ik} \in A} \pi(x_{ik})$ and necessity measure $N(A) = 1 - \Pi(\bar{A})$ where Π assesses at what level A is consistent with our knowledge represented by π whereas N evaluates at what level $\bar{A}$ is impossible.

The particularity of the possibilistic scale is that it can be interpreted in two-fold. First, it can be interpreted in an ordinal manner which means that possibility degrees reflect only a specific order between possible values. Second, the possibilistic scale can be interpreted in a numerical way meaning that possibility degrees make sense in the ranking scale. These two interpretations induce two definitions of possibilistic conditioning which consists in reviewing a possibility distribution by a new certain information ϕ, an interpretation of $\Phi \subseteq V$. The *product-based* conditioning is defined by:

$$\pi(\omega|_*\phi) = \begin{cases} \frac{\pi(\omega)}{\Pi(\phi)} \; if \; \omega[\Phi] = \phi \\ 0 \quad otherwise. \end{cases} \tag{1}$$

While the *min-based* conditioning is defined by:

$$\pi(\omega \,|_{min}\, \phi) = \begin{cases} 1 & if \; \pi(\omega) = \Pi(\phi) \text{ and } \omega[\Phi] = \phi \\ \pi(\omega) & if \; \pi(\omega) < \Pi(\phi) \text{ and } \omega[\Phi] = \phi \\ 0 & otherwise. \end{cases} \tag{2}$$

One crucial notion when sampling networks, is the estimation of possibility distribution from generated data sets. In the numerical interpretation, Joslyn

[8] has proposed a possibility distribution estimation method from imprecise data using possibilistic histograms. Moreover, he discusses the non-specificity of obtained possibility distributions in some particular cases such as certain and consistent data sets (for more details see [8]). Let $\mathcal{D}_i = \{d_i^{(l)}\}$ be a dataset relative to a variable X_i, $d_i^{(l)} \in D_i$ (resp. $d_i^{(l)} \subseteq D_i$) if data are precise (resp. imprecise). The number of occurrences of each $x_{ik} \in D_i$, denoted by N_{ik}, is the number of times x_{ik} appears in $\mathcal{D}_i$: $N_{ik} = N(\{x_{ik}\} \in \mathcal{D}_i)$. The non-normalized estimation $\hat{\pi}^{nn}(x_{ik})$ is expressed as follows:

$$\hat{\pi}^{nn}(x_{ik}) = \frac{N_{ik}}{N} \tag{3}$$

where N is the number of observations in $\mathcal{D}_i$. N is equal (resp. lower or equal) to the sum of N_{ik} if data are precise (resp. imprecise). Equation 3 could be defined as a set of variables $X_i, X_j, ...X_w$. In this case, N_{ik} becomes $N_{ik,jl,...,wp} = N(\{x_{ik}x_{jl}...x_{wp}\} \subseteq \mathcal{D}_{ijw})$.

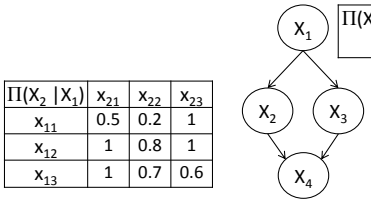
3 Learning possibilistic networks

3.1 Definition of possibilistic networks

Possibilistic networks [9] represent the possibilistic counterpart of Bayesian networks [10] having similarly two components: a *graphical component* composed of a DAG which encodes a set of independence relations (i.e. each variable $X_i \in V$ is conditionally independent of its non-descendent given its parents) and a *numerical component* corresponding to the set of conditional possibility distributions relative to each node $X_i \in V$ in the context of its parents, denoted by $Pa(X_i)$, i.e. $\pi(X_i|Pa(X_i))$. The two definitions of the possibilistic conditioning lead naturally to two different ways to define possibilistic networks: product-based possibilistic networks based on the product-based conditioning expressed by Equation 1. These models are theoretically and algorithmically close to Bayesian networks. In fact, these two models share the graphical component, i.e. the DAG and the product operator in the computational process. This is not the case of qualitative based on min-based conditioning defined by Equation 2 that represents a different semantic. In both cases, possibilistic networks are a compact representation of possibility distributions. More precisely, the joint possibility distribution could be computed by the possibilistic chain rule expressed as follows:

$$\pi_{\otimes}(X_1, ..., X_n) = \otimes_{i=1..n} \pi(X_i \mid_{\otimes} Pa(X_i)) \tag{4}$$

where $\otimes$ corresponds to the product operator (*) for quantitative possibilistic networks and to the minimum operator (min) for qualitative possibilistic networks. In the remaining, we focus on *product-based* possibilistic networks. Figure 1 represents an example of a *product-based* possibilistic network with four ternary variables.



$\Pi(X_1)$	x_{11}	x_{12}	x_{13}
	0.4	1	0.1

$\Pi(X_2 \mid X_1)$	x_{21}	x_{22}	x_{23}
x_{11}	0.5	0.2	1
x_{12}	1	0.8	1
x_{13}	1	0.7	0.6

$\Pi(X_3 \mid X_1)$	x_{31}	x_{32}	x_{33}
x_{11}	0.4	1	0.1
x_{12}	1	0.3	0.5
x_{13}	0.8	1	0.6

$\Pi(X_4 \mid X_2X_3)$	x_{41}	x_{42}	x_{43}	$\Pi(X_4 \mid X_2X_3)$	x_{41}	x_{42}	x_{43}	$\Pi(X_4 \mid X_2X_3)$	x_{41}	x_{42}	x_{43}
$x_{21}x_{31}$	1	0.3	0.4	$x_{22}x_{31}$	0.2	0.1	1	$x_{23}x_{31}$	0.6	1	0.5
$x_{21}x_{32}$	0.5	1	0.1	$x_{22}x_{32}$	1	0.5	0.5	$x_{23}x_{32}$	1	0.8	0.4
$x_{21}x_{33}$	1	0.2	0.8	$x_{22}x_{33}$	0.1	0.2	1	$x_{23}x_{33}$	1	0.7	0.3

Fig. 1. Example of a product-based possibilistic network

3.2 Possibilistic networks structure learning

By analogy to Bayesian networks, structure learning methods could be categorized into three families: constraint-based, score-based and hybrid methods. In the possibilistic case, Gebhardt and Kruse have proposed a score-based method handling imprecise data [5]. Borgelt et al. [4] have proposed possibilistic versions of two learning methods initially proposed to Bayesian networks: K2 and maximum weight spanning tree handling, also, imprecise data. Sangüesa et al. [6] have proposed two hybrid learning methods from precise data: the first one learns trees and the second one learns the more general structure of DAGs. Most of attempts to learn possibilistic networks are direct adaptations of learning methods initially proposed for Bayesian networks ignoring also parameters learning problem. Moreover, all these works have been proposed before advances made concerning possibilistic networks as independence models leading to use contrary hypothesis (* in the conditioning and min in the conditional independence).

3.3 Possibilistic networks parameters learning

Parameters learning of Bayesian networks is performed satisfying *maximum entropy* principle [11]. The possibilistic analog of the latter corresponds to minimum non-specificity. By analogy to Bayesian networks, learning possibilistic networks parameters consists in estimating possibility distributions according to minimum non-specificity principle [12], i.e. estimating the most specific possibility distributions. As far as we know, parameters learning has not been studied yet and existing learning methods compute possibility distributions using either Equation 3 as done in [4] and [5] or probability possibility transformations [13] as done in [6].

4 Evaluating learning algorithms

Probabilistic graphical models learning methods, in particular Bayesian networks, are tested using randomly generated networks (synthetic) or networks that have been used in real systems, so that the structure of the network is

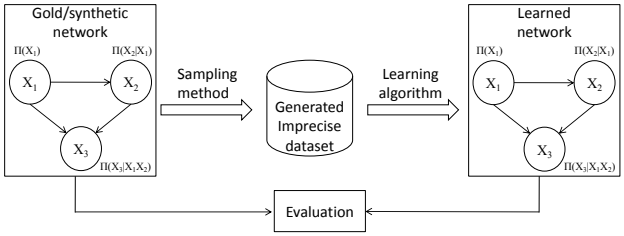


Fig. 2. Evaluation process of possibilistic networks learning algorithms

known and can serve as a rigorous gold standard e.g. Asia and Insurance networks. In the probabilistic case, evaluating learning algorithm is ensured using the following process: we select an arbitrary Bayesian network either a synthetic one or a gold standard from which we generate a data set using Forward Sampling algorithm [14]. Then, we try to recover the initial network using a learning algorithm and we compare the initial network with the learned one.

None of the existing possibilistic networks learning methods has proposed a formal evaluation strategy and each existing work has proposed its evaluation measure whose values are difficult to interpret. In the following, we propose to transpose the evaluation strategy proposed in the probabilistic case to the possibilistic one in order to evaluate a possibilistic networks learning algorithm as shown by Figure 2.

4.1 Generating possibilistic networks

In the possibilisic case, there are currently no publicly available possibilistic networks used in real systems and could be used as gold standard. We propose either to generate randomly a possibilistic network or to transform a gold Bayesian network to a possibilistic one. Generating a random possibilistic network consists in generating its two components. Concerning the graphical component, we could use any method proposed in the context of Bayesian networks such as [15]. For the numerical component, we propose to generate random values from [0,1] for each distribution satisfying normalization property, i.e. at least one of states degrees is equal to 1. We can, also, transform a Bayesian network to a possibilistic one retaining the same structure and performing a probability possibility transformation on its distributions, e.g. [13], on its probability distributions.

4.2 Sampling possibilistic networks

Once the possibilistic network is generated, we want to generate an imprecise dataset representative of its possibility distributions. To the best of our knowledge, there is no possibilistic networks sampling method. However, two approaches have been proposed [16, 17] to sample one variable and are based on α-*cut* notion: $\alpha\text{-}cut_{X_i} = \{x_{ik} \in D_i \text{ s.t. } \pi(x_{ik}) \geq \alpha\}$ where α is randomly generated from [0,1]. The epistemic sampling method proposed by Guyonnet et al. in [17] focuses on the generation of imprecise data by returning all values

of α-cut_{X_i} for any variable X_i. In fact, it returns a nested random set which represents the state of knowledge about the sampled variable X_i. Chanas and Nowakowski proposed another method in [16] which is dedicated to the generation of precise data from the pignistic probability distribution by returning a single value uniformly chosen from α-cut_{X_i}.

In this paper, we propose to generalize the variable sampling method proposed in [17] to possibilistic networks. This choice is justified by the fact that this method generates a more generic form of imperfect data i.e. imprecise data. The sampling process constructs a database of N (predefined) observations by instantiating all variables in V w.r.t. their possibility distributions. Obviously, variables are most easily processed w.r.t. a topological order, since this ensures that all parents are instantiated. Instantiating a parentless variable corresponds to computing its α-*cut*. Instantiating a conditioned variable corresponds to computing also its α-*cut* but given its sampled parents values. This could not be directly applied to conditional possibility distribution since it is composed of more than one distribution depending on the number of the values of its sampled parents. For conditional possibility distributions, we propose to use the *product-based* possibilistic counterpart of Jeffrey's rule which satisfies possibilistic Jeffrey's rule kinematics properties [18] to revise it before the sampling operation. The revision based on the possibilistic counterpart of Jeffrey's rule in the *product-based* framework has been formalized as follows [18]:

Definition 1. *Let $\pi(X_i)$ be a possibility distribution and (λ_1, σ_1),...,(λ_h, σ_h) be a set of exhaustive and mutually exclusive events where the uncertainty is of the form $\pi'(\lambda_i) = \sigma_i$ (meaning that after the revision operation, the possibility of each event λ_i is equal to σ_i). The revised possibility degree of any arbitrary event $\psi \subseteq D_i$ is computed as follows:*

$$\forall \psi \subseteq D_i, \pi'(\psi) = \max_{\lambda_i}(\sigma_i * \frac{\pi(\psi, \lambda_i)}{\Pi(\lambda_i)}) \tag{5}$$

Example 1. Let us consider the possibilistic network in Figure 1. The topological order is X_1, X_2, X_3, X_4. Applying the described sampling process we obtain:

1. X_1: $\alpha = 0.3$, α-cut$_{X_1} = \{x_{11}, x_{12}\}$.
2. X_2: $\alpha = 0.9$
   (a) $\pi'(x_{21}) = max(0.4 * 0.5, 1 * 1) = 1$, $\pi'(x_{22}) = max(0.4 * 0.2, 1 * 0.8) = 0.8$, $\pi'(x_{23}) = max(0.4 * 1, 1 * 1) = 1$.
   (b) α-cut$_{X_2} = \{x_{21}, x_{23}\}$.
3. X_3: $\alpha = 0.7$
   (a) $\pi'(x_{31}) = max(0.4 * 0.4, 1 * 1) = 1$, $\pi'(x_{32}) = max(0.4 * 1, 1 * 0.3) = 0.4$, $\pi'(x_{33}) = max(0.4 * 0.1, 1 * 0.5) = 0.5$.
   (b) α-cut$_{X_3} = \{x_{31}\}$.
4. X_4: $\alpha = 0.2$
   (a) $\pi'(x_{41}) = max(1 * 1 * 1, 1 * 1 * 0.6) = 1$, $\pi'(x_{32}) = max(1 * 1 * 0.3, 1 * 1 * 1) = 1$, $\pi'(x_{33}) = max(1 * 1 * 0.4, 1 * 1 * 0.5) = 0.5$.
   (b) α-cut$_{X_4} = \{x_{41}, x_{42}, x_{43}\}$.

The obtained observation is then $(\{x_{11}, x_{12}\}, \{x_{21}, x_{23}\}, \{x_{31}\}, \{x_{41}, x_{42}, x_{43}\})$. We repeat the process to obtain N samples.

4.3 Evaluation measures

An evaluation measure assesses learned possibilistic networks quality and quantify the efficiency of the learning method graphically or numerically. We could evaluate learning algorithms graphically by comparing the initial and the learned possibilistic networks structures using graphical evaluation measures proposed in the context of Bayesian networks, e.g. sensitivity (ratio of edges correctly identified in the learned network), specificity (ratio of edges correctly identified as not belonging in the learned network) and editing distance (number of operations required to transform a learned possibilistic network structure into the initial one. For more details, see [19, 20]. Note that, it is necessary to take into account Markov equivalence properties when computing these measures. In fact, we should compute editing distance between equivalence class representatives and sensitivity and specificity of DAGs skeletons i.e. without edges orientation or DAGs v-structure (in the form $X_i \longrightarrow X_j \longleftarrow X_k$).

Learning algorithms could be evaluated numerically by comparing the initial network and the learned one using a possibilistic dissimilarity measure between their joint possibility distribution as done by KL divergence in the probabilistic case. Such a measure has been proposed to compare two possibility distributions π and π' defined in D_i s.t. $\pi(x_{ik}) \geq \pi'(x_{ik}) \forall x_{ik} \in D_i$ [4]. This hypothesis is restrictive for comparing two possibilistic networks. However, we can use the possibilistic similarity measure proposed in [21] which is expressed by:

$$Aff(\pi_0, \pi_l) = 1 - \frac{\kappa * d(\pi_0, \pi_l) + \lambda * Inc(\pi_0, \pi_l)}{\kappa + \lambda} \tag{6}$$

Information affinity is based on two quantities: inconsistency degree $Inc(\pi_0, \pi_l) = 1 - \max_{\omega_i \in \Omega} \{\pi_0(\omega_i) \wedge \pi_l(\omega_i)\}$ ($\wedge$ can be taken as min or product operator[3]) and Manhattan distance i.e. $d(\pi_0, \pi_l) = \frac{\sum_{l=1}^{m} |\pi_0(\omega_i) - \pi_l(\omega_i)|}{m}$, where $\kappa > 0$ and $\lambda > 0$.

KL divergence and information affinity involve heavy computing if the number of variables increases. This can be explained by the fact that they involve all $\omega \in \Omega$. For KL divergence, we can compute an approximation to it, but, for information affinity, such approximation has not been studied yet.

5 Experimental study

This section proposes an experimental study having two main purposes. The first set of experiments evaluates the efficiency of the proposed sampling method to generate an imprecise data set representative of a given possibilistic network. The second set of experiments illustrates the whole proposed evaluation strategy on main existing possibilistic learning algorithms in literature. These experiments were ran on the following platform: 2.30 GHz Intel(R) Core (TM) i5-2410M with 8 Go of memory.

[3] using the min operator instead of the product means that we give less importance to the inconsistency degree.

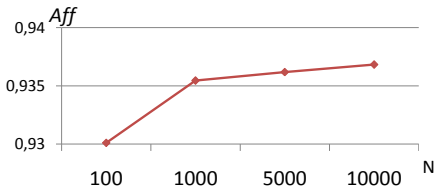


Fig. 3. Information affinity between π_0 and π_l w.r.t the number of data (average over 100 experiments)

5.1 Evaluation of the proposed sampling method

The first set of experiments evaluates the efficiency of our sampling method. We study the convergence of the joint possibility distribution computed from generated data using Equation 3, denoted by π_0, to the theoretical one, i.e. computed using Equation 4, denoted by π_0. Specifically, we generate synthetic data sets containing 100, 1000, 5000 and 10000 observations from 100 randomly generated possibilistic networks composed of nb nodes where nb is randomly generated in [5,10]. In order to compare π_0 and π_l, we measure the similarity between the two possibility distributions using Information Affinity (Equation 6) and we take $\lambda = \kappa = 1$ and $\wedge$ is the min operator. Figure 3 presents information affinity values between π_0 and π_l. Each value is the average of results of the 100 experiments carried out with a standard deviation around 0.04. Figure 3 shows that the information affinity grows relatively smoothly with the number of observations, as expected. This is an obvious result because when we increase the number of observations, the data set becomes more informative and representative of the joint possibility distribution, i.e. most possible ω_i appears more frequently, less possible appears less frequently and so on until reaching the least possible ω_i or impossible ω_i. Consequently, we deflate considerably the gap between the initial possibility distribution and the learned one. Note that in all experiments if $\pi_0(\omega_i) = 1$, then, $\pi_l(\omega_i) = 1$, i.e. the proposed sampling method conserves the most possible interpretation.

5.2 Illustration of the evaluation strategy

In the second set of experiments, we generate 100 data sets of 100, 1000, 5000 and 10000 observations from the famous Asia network [22] (8 nodes and 8 edges). This network is a probabilistuc one, so in order to adapt it to our possibilistic context, we apply *optimal probability possibility transformation* [13] on its conditional possibility distributions. Then, we apply existing possibilistic learning structure algorithms which handle imprecise data, i.e. the possibilistic adaptation of k2 (πK2), maximum weight spanning tree (πMWST) [4] and greedy search (πGS) [5]. In the current work, πK2 [4] and πMWST are tested using two scores, namely, *possibilistic mutual information* (d_{mi}) and *possibilistic* χ^2 *measure* (d_{χ^2}) and πK2 treats variables in a predefined order (we generate 5 orders in each experiment and we retain the best structure). πGS uses expected non specificity as score and begins with the class of all directed graphs w.r.t V that

	Editing distance			
N	100	1000	5000	10000
πGS	17,23+/-1,2	17,3+/-1,3	17,22+/-1,4	17,28+/-1,2
πMWST+$d_{\chi 2}$	12,65 +/- 0,8	11,81 +/- 1,2	11,97 +/- 1	11,98 +/- 1
πMWST+ d_{mi}	12,97 +/- 0,3	11,84 +/- 1,4	10,09 +/- 0,5	10 +/- 0
πK2+$d_{\chi 2}$	9,75 +/- 1,3	10,89 +/- 1,4	10,52 +/- 1,2	10,52 +/- 1,1
πK2+ d_{mi}	8,12 +/- 0,3	9,52 +/- 1,4	10,55 +/- 1,3	10,65 +/- 1,3
	Specificity			
N	100	1000	5000	10000
πGS	0,35+/-0,03	0,34+/-0,03	0,35+/-0,03	0,34+/-0,03
πMWST+$d_{\chi 2}$	0,8 +/- 0,03	0,84 +/- 0,04	0,83 +/- 0,02	0,83 +/- 0,02
πMWST+ d_{mi}	0,79 +/- 0,01	0,83 +/- 0,05	0,9 +/- 0,01	0,9 +/- 0,01
πK2+$d_{\chi 2}$	0,89 +/- 0,07	0,81 +/- 0,04	0,82 +/- 0,04	0,82 +/- 0,04
πK2+ d_{mi}	0,98 +/- 0,02	0,9 +/- 0,07	0,84 +/- 0,05	0,83 +/- 0,05
	Sensitivity			
N	100	1000	5000	10000
πGS	0,68+/-0,09	0,69+/-0,08	0,7+/-0,09	0,7+/-0,08
πMWST+$d_{\chi 2}$	0,27 +/- 0,06	0,34 +/- 0,1	0,32 +/- 0,06	0,31 +/- 0,06
πMWST+ d_{mi}	0,25 +/- 0,02	0,35 +/- 0,1	0,49 +/- 0,04	0,5 +/- 0
πK2+$d_{\chi 2}$	0,06 +/- 0,02	0,16 +/- 0,08	0,19 +/- 0,06	0,18 +/- 0,06
πK2+ d_{mi}	0,01 +/- 0,03	0,08 +/- 0,03	0,12 +/- 0,07	0,13 +/- 0,06

Table 1. Editing distance, specificity and sensitivity of learned structures

satisfy the condition $|Pa(X_i)| \leq 1$ for all $X_i \in V$ and we, also, ran it five times retaining the best learned structure. Then, we compute editing distance between equivalence class representatives, skeleton sensitivity and specificity between the learned and the initial structures. Table 1 shows the average of obtained results. We can see that πGS scoring function seems to be less interesting than the ones used by πK2 and πMWST which has not been previously established. Such a result clearly deserves more investigations but it is not the purpose of the present paper.

6 Conclusion

Despite the similarities between Bayesian networks and possibilistic ones and especially those based on the product operator (since they share the same graphical component and even same computations in the propagation process), working with possibility distributions highlights several difficulties when dealing with the learning task. This paper proposes a new evaluation strategy for *product-based* possibilistic networks learning algorithms. The proposed method provides several means to assess learned networks quality, i.e. we could use two families of evaluation measures: graphical ones to compare networks structures and information affinity to compute similarity between learned and initial distributions.

The proposed evaluation strategy presents a clear experimental framework. Thereby, it will be interesting to realize now a comparative and intensive study of existing possibilistic networks algorithms to evaluate score functions quality, learned networks quality and execution time. Future work concerns an approximation of the numerical evaluation measure information affinity in order to make its use possible in complex domains involving a huge number of variables.

References

[1] Dubois, D.: Possibility theory and statistical reasoning. Computational statistics & data analysis **51**(1) (2006) 47–69

[2] Benferhat, S., Smaoui, S.: Hybrid possibilistic networks. International journal of approximate reasoning **44**(3) (2007) 224–243

[3] R. Ayachi, N. Ben Amor, S.B.: A generic framework for a compilation-based inference in probabilistic and possibilistic networks. Information Sciences **257** (2014) 342–356

[4] Borgelt, C., Steinbrecher, M., Kruse, R.: Graphical models: representations for learning, reasoning and data mining. Volume 704. Wiley (2009)

[5] Gebhardt, J., Kruse, R.: Learning possibilistic networks from data. In: International Workshop on Artificial Intelligence and Statistics. (1996) 233–244

[6] Sangüesa, R., Cabós, J., Cortes, U.: Possibilistic conditional independence: A similarity-based measure and its application to causal network learning. International Journal of Approximate Reasoning **18**(1) (1998) 145–167

[7] Dubois, D., Prade, H.: Possibility theory: qualitative and quantitative aspects. In: Quantified representation of uncertainty and imprecision. (1998) 169–226

[8] Joslyn, C.: Towards an empirical semantics of possibility through maximum uncertainty. In: International Fuzzy Systems Association: Artificial Intelligence. (1991) 86–89

[9] Fonck, P.: Propagating uncertainty in a directed acyclic graph. In: Information Processing and Management of Uncertainty Conference. Volume 92. (1992) 17–20

[10] Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann (1988)

[11] Herskovits, E., Cooper, G.F.: An entropy-driven system for construction of probabilistic expert systems from databases. UAI '90, Elsevier (1991) 117–128

[12] Klir, G.J.: Uncertainty and information: foundations of generalized information theory. John Wiley & Sons (2005)

[13] Dubois, D., Prade, H., Sandri, S.: On possibility/probability transformations. In: Fuzzy logic. Volume 12. Springer (1993) 103–112

[14] Henrion, M.: Propagating uncertainty in Bayesian networks by probabilistic logic sampling. In: Uncertainty in Artificial Intelligence. (1986) 149–164

[15] Xiang, Y., Miller, T.: A well-behaved algorithm for simulating dependence structures of Bayesian networks. IJAM **1** (1999) 923–932

[16] Chanas, S., Nowakowski, M.: Single value simulation of fuzzy variable. Fuzzy Sets and Systems **25**(1) (1988) 43–57

[17] Guyonnet, D., Bourgine, B., Dubois, D., Fargier, H., Côme, B., Chilès, J.P.: Hybrid approach for addressing uncertainty in risk assessments. Journal of Environmental Engineering **129**(1) (2003) 68–78

[18] Benferhat, S., Tabia, K., Sedki, K.: Jeffrey's rule of conditioning in a possibilistic framework. ANN MATH **61**(3) (2011) 185–202

[19] Tsamardinos, I., Brown, L.E., Aliferis, C.F.: The max-min hill-climbing Bayesian network structure learning algorithm. Machine learning **65**(1) (2006) 31–78

[20] Shapiro, L.G., Haralick, R.M.: A metric for comparing relational descriptions. Pattern Analysis and Machine Intelligence **7**(1) (1985) 90–94

[21] Jenhani, I., Ben Amor, N., Elouedi, Z.: Decision trees as possibilistic classifiers. International Journal of Approximate Reasoning **48**(3) (2008) 784–807

[22] Lauritzen, S.L., Spiegelhalter, D.J.: Local computations with probabilities on graphical structures and their application to expert systems. Journal of the Royal Statistical Society (1988) 157–224