

# Thèse de Doctorat

**Maroua HADDAD**

*Mémoire présenté en vue de l'obtention du  
grade de Docteur de l'Université de Nantes  
Docteur de l'Université de Tunis  
sous le label de l'Université de Nantes Angers Le Mans*

**École doctorale : Sciences et technologies de l'information, et mathématiques**

**Discipline : Informatique, section CNU 27**

**Unité de recherche : Laboratoire d'Informatique de Nantes-Atlantique (LINA)  
LABoratoire de Recherche Opérationnelle, de Décision  
et de Contrôle de processus (LARODEC)**

**Soutenue le 14/12/2016**

## Learning possibilistic graphical models from data

### JURY

Président : **M. Zied ELOUEDI**, Professeur des Universités, Université de Tunis, Tunisie  
Rapporteurs : **M. Didier DUBOIS**, Directeur de recherche CNRS, Université Paul Sabatier, France  
**M. Eric LEFEVRE**, Professeur des Universités, Université d'Artois, France  
Examineur : **M. Karim TABIA**, Maître de conférences, Université d'Artois, France  
Directeurs de thèse : **M. Philippe LERAY**, Professeur des Universités, Université de Nantes, France  
**M<sup>me</sup> Nahla BEN AMOR**, Professeur des Universités, Université de Tunis, Tunisie



# Contents

<b>Introduction</b>	<b>1</b>
<b>I State-of-the-art</b>	<b>5</b>
<b>1 Possibility theory: An overview</b>	<b>7</b>
1.1 Introduction	7
1.2 Notations and definitions	8
1.2.1 Notations	8
1.2.2 Possibility distribution	8
1.2.3 Possibility and Necessity measures	9
1.2.4 Non-specificity	9
1.2.5 Possibilistic marginalization and conditioning	10
1.3 Possibilistic conditional independence relations	11
1.4 Possibility distribution estimation	12
1.5 Variable sampling	13
1.6 Possibilistic similarity measures	14
1.6.1 Information closeness	14
1.6.2 Sangûesa et al's distance	15
1.6.3 Minkowski distance	15
1.6.4 Information affinity	16
1.7 Possibility theory and links with other uncertainty theories	16
1.7.1 Possibility theory vs probability theory	16
1.7.2 Possibility theory vs belief function theory	18
1.7.3 Possibility theory vs imprecise probability theory	19
1.8 Conclusion	20
<b>2 Graphical representation of knowledge in uncertain frameworks</b>	<b>21</b>
2.1 Introduction	21
2.2 Background and notations on graphs	22
2.3 Bayesian networks	22
2.3.1 Definition	22
2.3.2 Propagation, applications and software solutions	23
2.3.3 Learning from data	24
2.4 Evidential networks	26
2.4.1 Definition	26
2.4.2 Propagation, applications and software solutions	27
2.4.3 Learning from data	27
2.5 Credal networks	27
2.5.1 Definition	27

2.5.2	Propagation, applications and software solutions . . . . .	28
2.5.3	Learning from data . . . . .	28
2.6	Possibilistic networks . . . . .	28
2.6.1	Definition . . . . .	28
2.6.2	Inference, applications and software solutions . . . . .	30
2.7	Learning possibilistic networks from data . . . . .	30
2.7.1	Constraint-based methods . . . . .	31
2.7.2	Score-based methods . . . . .	31
2.7.3	Hybrid methods . . . . .	32
2.7.4	Discussion . . . . .	32
2.8	Conclusion . . . . .	33
 <b>II Contributions</b>		<b>35</b>
<b>3</b>	<b>Benchmarking possibilistic networks learning algorithms and evaluation measures</b>	<b>37</b>
3.1	Introduction . . . . .	37
3.2	Possibilistic networks benchmark generation . . . . .	38
3.2.1	Possibilistic networks generation . . . . .	38
3.2.2	Possibilistic networks sampling . . . . .	38
3.3	Learning evaluation measures . . . . .	41
3.3.1	Graphical measures . . . . .	42
3.3.2	Numerical measures . . . . .	42
3.4	Experimental study . . . . .	43
3.4.1	Possibilistic networks sampling process evaluation . . . . .	43
3.4.2	Manhattan distance and its approximation evaluation . . . . .	44
3.5	Conclusion . . . . .	45
<b>4</b>	<b>Possibilistic networks parameters learning from imperfect data</b>	<b>47</b>
4.1	Introduction . . . . .	47
4.2	Building graphical models from imperfect data . . . . .	48
4.3	Evaluation of existing parameters learning algorithms . . . . .	48
4.3.1	Learning parameters based on transformations . . . . .	50
4.3.2	Learning parameters directly from data . . . . .	50
4.3.3	Discussion . . . . .	51
4.4	Possibilistic-likelihood-based parameters learning algorithm . . . . .	52
4.4.1	Imprecise likelihood . . . . .	52
4.4.2	Parameters learning algorithm from imprecise data . . . . .	54
4.5	Experimental study . . . . .	55
4.5.1	Possibilistic networks parameters learning evaluation . . . . .	55
4.5.2	Particular case: possibilistic classifiers parameters learning evaluation . . . . .	58
4.6	Conclusion . . . . .	59
<b>5</b>	<b>Possibilistic networks structure learning from imprecise data</b>	<b>61</b>
5.1	Introduction . . . . .	61
5.2	New possibilistic score . . . . .	62
5.3	Possibilistic adaptation of Greedy search algorithm . . . . .	63
5.4	Experimental study . . . . .	65
5.5	Conclusion . . . . .	66
 <b>Conclusion</b>		<b>67</b>



# List of Tables

1.1	Example of a joint possibility distribution on two binary variables . . . . .	11
1.2	Example of an imprecise dataset . . . . .	13
2.1	Joint possibility distribution of the network defined by Figure 2.2 in the numerical ( $\pi_x$ ) and the ordinal interpretation ( $\pi_m$ ) . . . . .	29
2.2	Summary table of measures properties . . . . .	33
4.1	Example of a Bayesian network . . . . .	50
4.2	Example (a) of transformation of Bayesian network in a possibilistic network using the optimal transformation (Equation 1.34) . . . . .	50
4.3	Example (b) transformation of Bayesian network in a possibilistic network using Klir transformation adaptation (Equation 1.41) . . . . .	51
4.4	Example of an imprecise dataset . . . . .	51
4.5	Example of learning possibilistic networks parameters . . . . .	51
4.6	Example of an imprecise dataset . . . . .	53
4.7	Example of a network with two variables defined on random sets . . . . .	53
4.8	Example of a possibilistic network with two binary variables . . . . .	54
4.9	Mean information affinity between initial networks and networks learned using DPNL and TPNL . . . . .	57
4.10	Description of Datasets . . . . .	58
4.11	% of correct classifications NBC, PNCPA, PNCTA and NCC classifiers on the datasets of Table 4.10 . . . . .	59
5.1	Example of an imprecise dataset . . . . .	62
5.2	Editing distance between initial and learned networks . . . . .	65
5.3	Manhattan distance between initial and learned networks . . . . .	66

# List of Figures

1	Thesis synopsis . . . . .	2
1.1	Example illustrating the specificity notion . . . . .	10
1.2	$\alpha$ -cut notion . . . . .	13
1.3	Example of $\alpha$ -cut . . . . .	14
2.1	Markov equivalence . . . . .	23
2.2	An example of possibilistic networks . . . . .	29
3.1	Experimental protocol of possibilistic networks sampling evaluation . . . . .	43
3.2	Information affinity between $\pi_0$ and $\pi_l$ w.r.t. the size of generated datasets (average over 100 experiments) . . . . .	43
3.3	Evolution of approximation of Manhattan distance w.r.t. $\Delta$ . . . . .	44
3.4	Experimental protocol of Manhattan distance approximation evaluation . . . . .	45
3.5	Evolution of convergence of Manhattan distance approximation values to Manhattan distance values w.r.t. sub- $\Omega$ . . . . .	45
3.6	Evolution of convergence of Manhattan distance approximation values to Manhattan distance values . . . . .	45
4.1	Cartography of reasoning models w.r.t. available data . . . . .	49
4.2	Proposed experimental protocol of possibilistic networks parameters learning evaluation . . . . .	56
4.3	Mean information affinity between initial networks and learned networks varying datasets size . . . . .	56
4.4	Manhattan distance approximation between initial networks and learned networks varying datasets size . . . . .	56
4.5	Evolution of mean information affinity between initial networks and learned networks distributions using DPNL w.r.t. datasets imprecision percentage . . . . .	57
4.6	Evolution of mean information affinity between initial networks and learned networks distributions using DPNL w.r.t. datasets consistency percentage . . . . .	57
5.1	Example of two markov equivalent graphs composed of two variables . . . . .	63
5.2	Initialization of greedy search . . . . .	64
5.3	Illustration of greedy search . . . . .	64
5.4	Proposed experimental protocol of possibilistic networks structure learning evaluation . . . . .	65

# List of Algorithms

1	Sampling process . . . . .	39
2	Sampling process (imprecision control) . . . . .	40
3	Sampling process (consistency control) . . . . .	41
4	Greedy search algorithm . . . . .	63





# Acknowledgments

*First and foremost, I would like to single out my supervisors:*

*I offer my sincerest gratitude to Pr. Philippe Leray for his patience, his support and the constructive discussions that we had throughout the accomplishment of this work and Pr. Nahla for her continuous encouragements, her motivation, and her valuable guidance that led to a successful fulfillment of this thesis. This dissertation would never have been achieved without their commitment.*

*I would like also thank all members of my thesis committee: M. Zied Elouedi, M. Eric Lefevre, M. Karim Tabia, and particularly M. Didier Dubois for his useful advice and his insights about my research during conferences and his visit to Tunisia.*

*I would like to thank my fellow labmates of DUKe team and those of my laboratory LARODEC for their friendships, support, and feedback. Special thanks go to my friends, they each helped make my time more fun and interesting.*

*Last but not the least, I would like to thank my insanely large family for taking the blows and giving me a chance to thrive especially my nieces, who supported me spiritually throughout writing this thesis and my life in general. This work would not have been possible without your support.*



# Résumé

Le formalisme des modèles graphiques fournit des outils puissants de représentation des connaissances sous incertitude. Il présente un cadre unificateur pour capturer des dépendances complexes entre les variables des systèmes de grande dimension. L'objectif principal de cette thèse concerne l'apprentissage des réseaux possibilistes conçus par Fonck (1990) comme étant la contrepartie possibiliste des réseaux bayésiens développés par Pearl et Jensen (1988,1996). Les réseaux possibilistes représentent une combinaison intéressante entre la théorie des graphes et la théorie des possibilités qui a été inventée par Zadeh (1987) comme une extension de sa théorie des ensembles flous et développée par Dubois et Prade (1988).

A l'instar des réseaux bayésiens, un réseau possibiliste est caractérisé par deux composantes: Une composante *graphique* définie par un graphe orienté acyclique (DAG) où chaque variable du domaine étudié correspond à un nœud et les interactions entre les variables constituent l'ensemble des arcs dans le DAG. La deuxième composante est *numérique* et composée de distributions de possibilité permettant de coder l'incertitude des variables sachant leurs parents dans le DAG. Il existe deux manières de définir la contrepartie possibiliste des réseaux bayésiens: les réseaux possibilistes basés sur le produit, nommés également réseaux possibilistes quantitatifs. Ces réseaux sont très semblables aux réseaux bayésiens théoriquement et algorithmiquement. En effet, les deux modèles partagent la même composante graphique (c'est-à-dire le DAG) et également l'opérateur produit dans le processus calculatoire. Cela n'est pas le cas des réseaux possibilistes basés sur le min, nommés également réseaux possibilistes qualitatifs et qui représentent une sémantique bien différente.

Au début des années 2000, plusieurs travaux de recherche dédiés aux réseaux possibilistes ont été proposés pour raisonner avec des informations incertaines et imprécises. La plupart de ces travaux concernent la propagation des informations dans les réseaux possibilistes ou leur application dans plusieurs domaines comme les systèmes tuteurs intelligents (2006), la spécialisation sociale dans les espaces métropolisés (2014) et la recherche d'information (2015). Cependant, leur apprentissage à partir de données reste un véritable défi. En effet, quelques travaux ont été proposés par Borgelt et Kruse (2000) et Sangüesa et al. (1998) dans ce sens et sont des adaptations directes des méthodes d'apprentissage des réseaux bayésiens sans prendre en considération les spécificités du cadre possibiliste, ce qui les rendait limités et mal justifiés théoriquement. En outre, ces travaux souffrent de l'absence d'une procédure de validation standard et chaque méthode propose une mesure d'évaluation dont les valeurs sont difficiles à interpréter.

Cette thèse présente deux contributions majeures. La première contribution consiste à proposer une stratégie de validation pour les algorithmes d'apprentissage des réseaux possibilistes. Cette stratégie propose trois variantes de processus d'échantillonnage permettant de générer des ensembles de données imprécises à partir de réseaux générés aléatoirement, et deux nouvelles mesures d'évaluation possibilistes: la première est une mesure globale et calcule une approximation de la distance de Manhattan normalisée dans le contexte des réseaux possibilistes pour évaluer l'écart entre leurs distributions jointes. La deuxième mesure, basée sur *information affinity* proposé par Jenhani et al. (2007), est locale et compare les distributions de possibilité conditionnelles de deux réseaux partageant la même structure. Notre deuxième contribution consiste à proposer une approche globale pour l'apprentissage des réseaux possibilistes basés sur le produits permettant à la fois d'apprendre les paramètres (composante numérique) et la structure (composante graphique). Nous proposons une fonction de vraisemblance possibiliste pour paramétrer les réseaux possibilistes et définir une nouvelle fonction de score. Une étude expérimentale détaillée montrant la faisabilité et l'efficacité des méthodes proposées a été aussi proposée pour chaque partie de cette thèse.



# Abstract

The formalism of graphical models provides powerful knowledge representation tools under uncertainty. It presents a unifying framework for capturing complex dependencies among variables of high-dimensional systems. The main focus of this thesis concerns learning possibilistic networks first conceived by Fonck (1990) as possibilistic counterpart of the well known Bayesian networks developed by Pearl and Jensen (1988,1996). Possibilistic networks represent an interesting combination between graph theory and possibility theory which was coined by Zadeh (1987) as an extension of its fuzzy sets theory and developed by Dubois et Prade (1988).

Like Bayesian networks, a possibilistic network has two components: a *graphical* component composed of a directed acyclic graph (DAG) in which each variable of the studied domain corresponds to a node and interactions between variables constitutes the set of edges in the DAG. The second component is *numerical* composed of possibility distributions coding variables uncertainty given their parents in the DAG. There are two different ways to define the possibilistic counterpart of Bayesian networks: quantitative also called *product-based* possibilistic networks. 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, also called *min-based* possibilistic networks, that represent a different semantic.

In the early 2000s, several research works dedicated to possibilistic networks have been proposed to reason with uncertain and imprecise information. Most of these works concern information propagation in possibilistic networks or their application in various domains such as intelligent tutoring systems (2006), social specialization in metropolized spaces (2014) and information retrieval (2015). However, their learning from data remains a real challenge. In fact, only few works address this problem and existing ones proposed by Borgelt and Kruse (2000) and Sangüesa et al. (1998) are direct adaptations of Bayesian networks learning methods without any awareness of specificities of the possibilistic framework which made them limited and theoretically unsound. Moreover, These works suffer from the lack of an accurate and standard validation procedure and each method proposes an evaluation measure whose values are difficult to interpret.

This thesis presents two major contributions. The first one consists on proposing a validation strategy for possibilistic networks learning algorithms. This strategy proposes three variants of sampling process, which consists on generating imprecise datasets from randomly generated networks, and two new possibilistic evaluation measures: the first one is a global measure and presents an approximation of the normalized Manhattan distance in the context of possibilistic networks to assess the gap between their joint distributions. The second measure, based on *information affinity* proposed by Jenhani et al. (2007), is local and compares conditional possibility distributions of two networks sharing the same structure. Our second contribution consists on proposing a global *product-based* possibilistic networks learning approach i.e. including parameters (numerical component) and structure (graphical component) learning. We propose a possibilistic likelihood function to learn possibilistic networks parameters and to define a new score function. A detailed experimental study showing the feasibility and the efficiency of the proposed methods has been also proposed to each part of this thesis.



# Introduction

## Context

Machine learning is the sub-field of artificial intelligence gathering techniques that take up the challenge to help humans to produce useful patterns from data. One of the most known techniques consists on building/learning graphical models from data which also fits within the dependence analysis framework and corresponds to extracting dependency relations from observed data in order to make predictions. The success of graphical models is due to the fact that they meet our requirements of explicitness and clarity by transforming high-dimensional domains into graphs making, thereby, information representation and reasoning easily supported by human mind.

Graphical models were generally defined as a marriage between probability theory and graph theory that provides a natural tool for dealing with two major problems, namely uncertainty and complexity (Murphy, 2001). However, over the last three decades, several graphical models have been proposed in non classical uncertainty frameworks which are semantically different from the probabilistic one. These recently born models were studied to handle some particular situations when all numerical data are not available or precisely defined, which may compromise the application of probabilistic framework. As consequence of the above-mentioned items, graphical models could be redefined as a marriage between an uncertainty theory and graph theory that provides a natural tool for dealing with two major problems, namely uncertainty and/or imprecision and complexity.

This thesis investigates non classical graphical models and more precisely possibilistic networks, introduced by Fonck (1992), which represent the counterpart of Bayesian networks (Pearl, 1988; Jensen, 1996) in the possibility theory (Zadeh, 1978; Dubois and Prade, 1998). The latter offers a natural and simple formal framework to represent imprecise and uncertain information and to describe states of the world in both qualitative and quantitative aspects. Despite the fact that this is a rather young area of research, possibilistic networks have attracted the attention of many researchers. In fact, first works dedicated to them concerned mainly information propagation (Fonck, 1992; Ben Amor et al., 2003; Benferhat and Smaoui, 2007; Ayachi et al., 2014). Then, they have been applied in various real domains such as intelligent tutoring systems (Adina, 2006), social specialization in metropolized spaces (Caglioni et al., 2014) and information retrieval (Chebil et al., 2015; Boughanem et al., 2009). However, contrarily to Bayesian networks, learning possibilistic networks from data has not been deeply studied. In fact, only few works address this problem and existing ones (Borgelt et al., 2009; Sangüesa et al., 1998) are direct adaptations of Bayesian networks learning methods without any awareness of specificities of the possibilistic framework and of advances made concerning possibilistic networks as models of independence (Ben Amor and Benferhat, 2005).

The main achievement of this thesis is proposing a global possibilistic networks learning approach from imperfect data. Our first contribution consists on proposing a validation strategy which will represent a clear experimental framework allowing the realization of a comparative and intensive study possibilistic for networks learning algorithms. The aim of the remaining parts of this thesis consists on proposing a new learning approach from imperfect data including two phases: the first one is dedicated to parameters learning i.e. inferring conditional possibility distributions. The second phase concerns the structure learning i.e. constructing the graph including dependence relations detected from data.



## Thesis overview

Figure 1 presents the thesis organization and inter-dependencies between chapters.

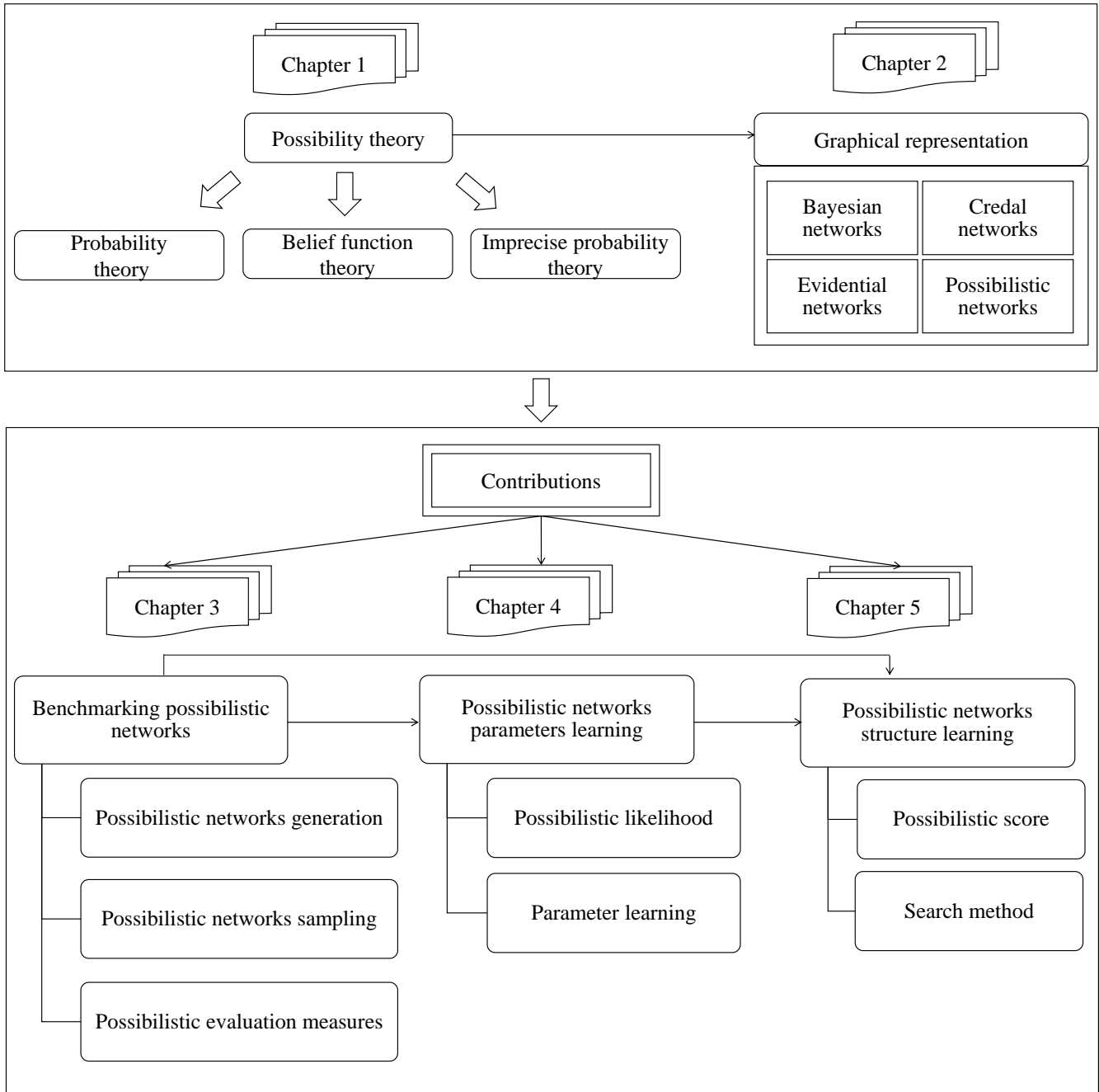


Figure 1: Thesis synopsis

**Chapter 1:** Possibility theory: An overview. This chapter gives the necessary background regarding basic concepts of possibility theory and briefly recalls other uncertainty theories i.e. probability theory, belief function theory and imprecise probability theory highlighting their links with possibility theory.

**Chapter 2:** Graphical representation of knowledge in uncertain frameworks. This chapter overviews mainly researches devoted to possibilistic networks, in particular their learning from data, and briefly introduces other graphical reasoning models that are Bayesian networks, evidential networks and credal networks.

Chapter 3: Benchmarking possibilistic networks learning algorithms and evaluation measures. In this chapter, we provide an evaluation strategy by proposing three variants of sampling process to generate imprecise datasets from possibilistic networks, and two possibilistic evaluation measures to quantify the similarity between two networks.

Chapter 4: Possibilistic networks parameters learning from imperfect data. This chapter provides a new possibilistic likelihood function which represents the first step of proposing a parameters learning algorithm.

Chapter 5: Possibilistic networks structure learning from imperfect data. In this chapter, we start by proposing a new possibilistic score. Then, we apply greedy search algorithm to learn possibilistic networks structure.

## Publications

This research work on learning possibilistic networks was the subject of the following publications:

- Our preliminary work about learning possibilistic networks not described in this thesis handles possibilistic data, has been published in ([Haddad et al., 2013](#))
- A survey of possibilistic networks learning methods has been published in ([Haddad et al., 2015c,a](#))
- Our first contribution dedicated to benchmarking possibilistic networks learning algorithms has been published in ([Haddad et al., 2015b](#))
- An experimental study comparing our second contribution i.e. our possibilistic networks parameter learning method with a probability-possibility transformation based has been published in ([Haddad et al., 2016](#))





## **State-of-the-art**



# Possibility theory: An overview

## Contents

<b>1.1</b>	<b>Introduction</b>	<b>7</b>
<b>1.2</b>	<b>Notations and definitions</b>	<b>8</b>
1.2.1	Notations	8
1.2.2	Possibility distribution	8
1.2.3	Possibility and Necessity measures	9
1.2.4	Non-specificity	9
1.2.5	Possibilistic marginalization and conditioning	10
<b>1.3</b>	<b>Possibilistic conditional independence relations</b>	<b>11</b>
<b>1.4</b>	<b>Possibility distribution estimation</b>	<b>12</b>
<b>1.5</b>	<b>Variable sampling</b>	<b>13</b>
<b>1.6</b>	<b>Possibilistic similarity measures</b>	<b>14</b>
1.6.1	Information closeness	14
1.6.2	Sangüesa et al's distance	15
1.6.3	Minkowski distance	15
1.6.4	Information affinity	16
<b>1.7</b>	<b>Possibility theory and links with other uncertainty theories</b>	<b>16</b>
1.7.1	Possibility theory vs probability theory	16
1.7.2	Possibility theory vs belief function theory	18
1.7.3	Possibility theory vs imprecise probability theory	19
<b>1.8</b>	<b>Conclusion</b>	<b>20</b>

## 1.1 Introduction

Most of researches of the last decades have proved that imperfection, be it uncertainty or imprecision, is unavoidable in real world applications and must be incorporated in information systems. This is due mainly to two imperfection sources: variability of observed phenomena and information incompleteness. To deal with imperfect information, several theories have been developed to make reasoning possible under uncertainty and/or imprecision. For a long time, probability theory has been considered as the unique

normative model to cope with imperfection by presenting a classical well founded framework manipulating uncertain but *precise* information. Nevertheless, probability theory, as good as it is, does not remain the best alternative where imprecision is inherent in the studied domain, where available information are simply preferences or where we are facing incomplete information. Thereby, over the last five decades, a lot of effort has been put into developing new non-classical uncertainty theories. The most visible of these theories are fuzzy sets theory (Zadeh, 1965), evidence theory (Shafer, 1976), imprecise probability theory (Dempster, 1967) and possibility theory (Zadeh, 1978; Dubois and Prade, 1988) and stand out as best alternatives to probability theory (Dubois et al., 1996).

In this thesis, we are interested by possibility theory, first formalized by Shackle (Shackle, 1969), then, re-introduced by Zadeh (Zadeh, 1978) as an extension of its fuzzy sets theory. It was later developed by Dubois and Prade (Dubois and Prade, 1988, 2000; Dubois, 2006). To a large extent, possibility theory is comparable to probability theory because it is based on set-functions. However, contrarily to the probabilistic case, possibility theory is able to offer a natural and simple formal framework representing *imprecise* and uncertain information and is able to describe epistemic states in both qualitative and quantitative aspects (Dubois and Prade, 1998). Among this panoply of theories, this chapter presents basic elements of possibility theory and its links, in order, with probability theory, evidence theory and finally imprecise probability theory.

This chapter is organized as follows: Section 1.2 provides some notations that will be used in the remaining of this chapter and defines the notion of possibility distribution. Section 1.4 describes how we can derive a possibility distribution directly from data using possibilistic histograms. Section 1.5 briefly presents two methods to generate a dataset representative of a given possibility distribution. Section 1.6 is dedicated to possibilistic similarity/dissimilarity measures. Finally, Section 1.7 is devoted to the presentation of some known uncertainty theories and shows how they are related to possibility theory.

## 1.2 Notations and definitions

This section recalls basic concepts of possibility theory i.e. possibility distribution, possibility and necessity measures, non-specificity, marginalization, conditioning and conditional independence.

### 1.2.1 Notations

We first give some notations that will be used in the remaining.

- $V = \{X_1, X_2, \dots, X_n\}$  denotes a set of  $n$  variables describing a studied domain.
- $D_i$  denotes the supposedly finite domain associated with the variable  $X_i$ .
- $x_{ik}$  denotes an instance, a state (a possible value) of  $X_i$ .
- $A_i$  denotes a subset of instances of a variable  $X_i$  i.e.  $A_i \subseteq D_i$ .
- $\Omega = D_1 \times \dots \times D_n$  denotes the universe of discourse, which is the Cartesian product of all variable domains in  $V$ .
- Each element  $\omega \in \Omega$  is called interpretation or event and is denoted by a tuple  $(x_{1k}, \dots, x_{nl})$ .
- The power set  $2^{\text{card}(D_i)}$  is the set of all subsets of  $D_i$  including also  $\emptyset$  and  $D_i$ .

### 1.2.2 Possibility distribution

The basic building block of possibility theory is the notion of possibility distribution  $\pi$  which corresponds to a mapping from the universe of discourse  $\Omega$  to the unit interval  $[0, 1]$ . For any state  $\omega \in \Omega$ ,  $\pi(\omega) = 1$  means that  $\omega$  realization is totally possible according to an agent for variables  $X_1, \dots, X_n$  and  $\pi(\omega) = 0$  means that  $\omega$  is an impossible state. It is generally assumed that at least one state  $\omega$  is totally possible and  $\pi$  is then said to be normalized. More formally, a possibility distribution  $\pi$  is normalized if:

$$\max_{\omega \in \Omega} \pi(\omega) = 1 \quad (1.1)$$

Otherwise,  $\pi$  is said sub-normalized and in this case,  $Inc(\pi) = 1 - \max_{\omega \in \Omega} \pi(\omega)$  is called the inconsistency degree of  $\pi$ . It is evident that  $Inc(\pi) = 0$  if  $\pi$  is normalized. This concept plays an important role in assessing similarity between possibilistic pieces of information by integrating the conflict degree between them in the similarity measure.

Extreme cases of knowledge are presented by:

- *Complete knowledge*:  $\exists \omega_i \in \Omega$  s.t.  $\pi(\omega_i) = 1$  and for all  $\omega_j \in \Omega$  s.t.  $\omega_j \neq \omega_i, \pi(\omega_j) = 0$
- *Total ignorance*:  $\forall \omega \in D_i, \pi(\omega) = 1$  (all values in  $\Omega$  are possible).

The particularity of the possibilistic scale is that it can be interpreted in two different ways:

- Ordinal (qualitative) interpretation: The possibility distribution is a mapping from the universe of discourse  $\Omega$  to an ordinal scale where only the order of values is important i.e. possibility degrees reflect only a specific order between possible values.
- Numerical (quantitative) interpretation: The possibility distribution is a mapping from the universe of discourse  $\Omega$  to a numerical scale where possibility degrees make sense in the ranking scale and could be manipulated by arithmetic operators.

Note that the notion of a possibility distribution could be defined on a variable  $X_i$  and it corresponds to a mapping from  $D_i$  to the unit interval  $[0,1]$ :

$\pi(X_i) : D_i \rightarrow [0,1]$ . In the remaining, for simplicity, we denote  $\pi(X_i = x_{ik})$  by  $\pi(x_{ik})$  and we define possibilistic concepts on a variable.

### 1.2.3 Possibility and Necessity measures

Contrary to probability theory which uses only one measure, namely the probability measure, possibility theory uses two dually related measures, namely possibility and necessity, to assess the plausibility and the certainty of any subset of events of a variable  $X_i$ . Given a possibility distribution  $\pi(X_i)$  defined on  $D_i$  and any subset  $A \subseteq D_i$ :

- Possibility measure is expressed by:

$$\Pi(A) = \max_{x_{ik} \in A} \pi(x_{ik}) \quad (1.2)$$

The possibility measure  $\Pi$  assesses at what level  $A$  is consistent with our knowledge represented by  $\pi$  i.e. at what level it is *possible* that  $x_{ik}$  pertains to  $A$ .

- Necessity measure is expressed by:

$$N(A) = 1 - \Pi(\bar{A}) \quad (1.3)$$

The necessity measure  $N$  assesses at what level  $A$  is implied by our knowledge expressed by  $\pi$  i.e. at what level it is *certain* that  $x_{ik}$  pertains to  $A$ .

**Example 1.2.1.** Let  $\pi$  be the possibility distribution relative to the variable  $X_1$  such that  $D_1 = \{x_{11}, x_{12}, x_{13}\}$  and  $\pi(x_{11}) = 0.4, \pi(x_{12}) = 0.7$  and  $\pi(x_{13}) = 1$  and  $A = \{x_{11}, x_{12}\}$ .

$$\Pi(A) = \max(0.4, 0.7) = 0.7.$$

$$N(A) = 1 - \max(1) = 0.$$

### 1.2.4 Non-specificity

Possibility theory is driven by minimum non-specificity principle (Yager, 1992). Let  $\pi_1$  and  $\pi_2$  be two possibility distributions on  $X_i$ ,  $\pi_1$  is said to be more specific (more informative) than  $\pi_2$  iff:

$$\forall x_{ik} \in D_i, \pi_1(x_{ik}) \leq \pi_2(x_{ik}) \quad (1.4)$$

**Example 1.2.2.** Let  $\pi_1(X_1)$  and  $\pi_2(X_1)$  be two possibility distributions relative to the variable  $X_1$  such that  $D_1 = \{x_{11}, x_{12}, x_{13}\}$  and  $\pi_1(x_{11}) = 0.4, \pi_2(x_{12}) = 0.7, \pi_3(x_{13}) = 1, \pi_2(x_{11}) = 0.5, \pi_2(x_{12}) = 0.9$  and  $\pi_2(x_{13}) = 1$ .  $\pi_1$  is more specific than  $\pi_2$ .



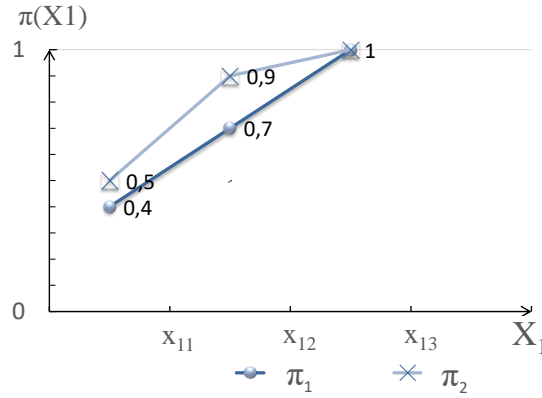


Figure 1.1: Example illustrating the specificity notion

In some situations, it is necessary to measure imprecision relative to a possibility distribution in order to compare it with other distributions and to decide which is the most specific or informative one. This imprecision is measured by non-specificity, denoted by  $nsp$ , and expressed as follows (Klir and Mariano, 1987):

Let  $\pi_{(k)}$  be the  $k^{th}$  degree in a possibility distribution  $\pi$  considered in a decreasing order of  $\pi$  values ( $\pi_{(1)}$  is the highest degree and  $\pi_{(m)}$  is the smallest one)

$$nsp(\pi) = \left[ \sum_{i=1}^m (\pi_{(i)} - \pi_{(i+1)}) \log_2 i \right] + (1 - \pi_{(1)}) \log_2 m \quad (1.5)$$

where  $\pi_{m+1} = 0$  by convention.

**Example 1.2.3.** Let us consider the possibility distributions in Example 1.2.2.

$$nsp(\pi_1) = (1 - 0.7) \log_2(1) + (0.7 - 0.4) \log_2(2) + (0.4 - 0) \log_2(3) = 0.9340.$$

$$nsp(\pi_2) = (1 - 0.5) \log_2(1) + (0.9 - 0.5) \log_2(2) + (0.5 - 0) \log_2(3) = 1.1925.$$

$\pi$  is more specific than  $\pi'$

## 1.2.5 Possibilistic marginalization and conditioning

When we are studying multivariate domains, a possibility distribution  $\pi$  relative to  $V$  defined on the universe of discourse  $\Omega$  must be manipulated in order to calculate uncertainties concerning a subset of variables pertaining to  $V$  using marginalization or to update them after receiving new information using conditioning.

The marginalization is an important notion used to determine how the realization of specific values of some variables affects remaining variables possibility distributions. We derive marginal distributions relative to subsets of variables using the **maximum** operator for both quantitative and qualitative interpretations. The marginalization is expressed as follows:

$$\pi(X_i) = \max_{\{X_1, X_j, X_n\}, j \neq i} \pi(X_1, \dots, X_n) \quad (1.6)$$

The possibilistic conditioning represents another important notion in possibility theory which consists in reviewing a possibility distribution  $\pi$  by a new certain information  $\Phi \subseteq \Omega$ . The two interpretations induce two definitions of possibilistic conditioning (Dubois and Prade, 1998).

The product-based conditioning based on the product operator is defined by the following Bayes-like equation:

$$\pi(\omega|\Phi) = \begin{cases} \frac{\pi(\omega)}{\Pi(\Phi)} & \text{if } \omega \in \Phi \\ 0 & \text{otherwise.} \end{cases} \quad (1.7)$$

The min-based conditioning based on the min operator is defined as follows:

$$\pi(\omega|\Phi) = \begin{cases} 1 & \text{if } \pi(\omega) = \Pi(\Phi) \text{ and } \omega \in \Phi \\ \pi(\omega) & \text{if } \pi(\omega) < \Pi(\Phi) \text{ and } \omega \in \Phi \\ 0 & \text{otherwise.} \end{cases} \quad (1.8)$$

These two definitions of conditioning satisfy a unique equation close to the Bayesian rule, of the form:

$$\forall \omega, \pi(\omega) = \pi(\omega | \Phi) \otimes \Pi(\Phi). \quad (1.9)$$

where  $\otimes$  corresponds to the **minimum** operator for Equation (1.8) and to the **product** operator for Equation (1.7). The min-based conditioning (1.8) corresponds to the least specific solution of Equation (1.9) first proposed by Hisdal (1978).

**Example 1.2.4.** Let  $X_1$  and  $X_2$  be two binary variables defined on  $D_1 = \{x_{11}, x_{12}\}$  and  $D_2 = \{x_{21}, x_{22}\}$  such that its joint possibility distribution is given by Table 1.1.

$X_1$	$X_2$	$\pi(X_1, X_2)$
$x_{11}$	$x_{21}$	0.8
$x_{11}$	$x_{22}$	0.4
$x_{12}$	$x_{21}$	0.6
$x_{12}$	$x_{22}$	1

Table 1.1: Example of a joint possibility distribution on two binary variables

In the numerical setting, using product-based conditioning defined by Equation 1.7, we obtain:

$$\begin{aligned} \pi(X_2 = x_{21}|X_1 = x_{11}) &= \frac{0.8}{\max(0.8, 0.4)} = 1. \\ \pi(X_2 = x_{22}|X_1 = x_{11}) &= \frac{0.4}{\max(0.8, 0.4)} = 0.5. \\ \pi(X_2 = x_{21}|X_1 = x_{12}) &= \frac{0.6}{\max(0.6, 1)} = 0.6. \\ \pi(X_2 = x_{22}|X_1 = x_{12}) &= \frac{1}{\max(0.6, 1)} = 1. \end{aligned}$$

In the ordinal setting, using min-based conditioning defined by Equation 1.8, we obtain:

$$\begin{aligned} \pi(X_2 = x_{21}|X_1 = x_{11}) &= 1. \\ \pi(X_2 = x_{22}|X_1 = x_{11}) &= 0.4. \\ \pi(X_2 = x_{21}|X_1 = x_{12}) &= 0.6. \\ \pi(X_2 = x_{22}|X_1 = x_{12}) &= 1. \end{aligned}$$

## 1.3 Possibilistic conditional independence relations

Representing independence relations between variables of a complex domain enables us to decompose the joint distribution which grows exponentially with the number of variables in the joint distribution. Thereby, using independence relations between variables enables us to replace the joint distribution by conditional distributions of each variable in the context of others variables to which it is dependent. From an operational point of view, two forms of independence can be distinguished:

- *Decompositional independence* which ensures the decomposition of a joint distribution pertaining to tuples of variables into local distributions on smaller subsets of variables. The reasoning machinery can then work at a local level without losing any information.
- *Causal independence* for expressing the lack of causality between variables. This form of independence is always characterized in semantic terms. Roughly speaking, a variable (or set of variables) is said to have no influence on another variable (or set of variables) if our belief in the value of the latter does not change when learning something about the value of the former.

### Possibilistic causal independence

The idea in defining possibilistic causal independence relation based on the possibilistic conditioning is that  $X$  is considered as independent from  $Y$  in the context  $Z$  if:

$$\Pi(X | Y \wedge Z) = \Pi(X | Z), \forall X, Y, Z. \quad (1.10)$$

Since possibility theory has two kinds of conditioning (see Section 1.2.5), this leads to two definitions of causal possibilistic independence:

- **Min-based independence relation** obtained by using the min-based conditioning (Equation (1.8)).
  - **Product independence relation** obtained by using the product-based conditioning (Equation 1.10).
- We can rewrite this form of independence using:

$$\Pi(X \wedge Y | Z) = \Pi(X | Z) * \Pi(Y | Z), \forall X, Y, Z. \quad (1.11)$$

or equivalently,

$$\Pi(X | Y \wedge Z) = \Pi(X | Z), \forall X, Y, Z. \quad (1.12)$$

### Possibilistic decompositional independence

In the possibilistic framework, the standard decompositional independence between  $X$  and  $Y$  in the context  $Z$  is represented by the **non-interactivity** relation, denoted by  $I_{NI}(X, Z, Y)$  (NI for Non-Interactivity) and defined by:

$$\Pi(X \wedge Y | Z) = \min(\Pi(X | Z), \Pi(Y | Z)), \forall X, Y, Z. \quad (1.13)$$

## 1.4 Possibility distribution estimation

Possibility distribution estimation is a crucial notion, especially, when we deal with learning tasks. Estimating a possibility distribution consists on deriving them from a dataset and it depends on the possibilistic scale interpretation. In fact, in the ordinal setting, estimating possibility distributions encounter difficulties since in this case, possibility degrees reflect an order between the world states and could not be quantified by numbers derived from observation occurrences in a dataset. In the numerical interpretation, [Joslyn \(1991\)](#) has proposed a possibility distribution estimation method from imprecise data using possibilistic histograms. Moreover, Joslyn discusses the non-specificity of obtained possibility distributions in some particular cases such as certain and consistent data sets

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  $\hat{f}^{nn}(x_{ik})$  is expressed as follows:

$$\hat{f}^{nn}(x_{ik}) = \frac{N_{ik}}{Nb} \quad (1.14)$$

where  $Nb$  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).

Joslyn normalizes the obtained possibility distribution by dividing it by the maximum. Equation 1.14 becomes:

$$\hat{\pi}^n(x_{ik}) = \frac{N_{ik}}{\max(N_{ik})} \quad (1.15)$$

Equation 1.15 can be defined on a set of variables  $X_i, X_j, \dots, X_w$ . In this case,  $N_{ik}$  becomes  $N_{ik,jl,\dots,wp} = N(\{x_{ik}x_{jl}\dots x_{wp}\} \subseteq \mathcal{D}_{ijw})$ .

$X_1$	$X_2$
$x_{11}, x_{12}$	$x_{21}, x_{22}$
$x_{12}, x_{13}$	$x_{21}, x_{22}$
$x_{11}, x_{12}$	$x_{21}$
$x_{11}, x_{13}$	$x_{21}$
$x_{12}$	$x_{22}$

Table 1.2: Example of an imprecise dataset

**Example 1.4.1.** Table 1.2 presents an example of an imprecise dataset relative to a ternary variable  $X_1$  such that  $D_1 = \{x_{11}, x_{12}, x_{13}\}$  and a binary variable  $X_2$  such that  $D_2 = \{x_{21}, x_{22}\}$ . Using Equation 1.15, we obtain the following distributions:

$$N(x_{11}) = 3, N(x_{12}) = 4 \text{ and } N(x_{13}) = 2.$$

The normalized estimation of  $\pi(X_1)$  is:

$$\hat{\pi}^n(x_{11}) = \frac{3}{4}, \hat{\pi}^n(x_{12}) = \frac{4}{4} \text{ and } \hat{\pi}^n(x_{13}) = \frac{2}{4}.$$

In the same way:

$$N(x_{21}) = 4 \text{ and } N(x_{22}) = 3.$$

$$\hat{\pi}^n(x_{21}) = \frac{4}{4} \text{ and } \hat{\pi}^n(x_{22}) = \frac{3}{4}.$$

We also have:  $N(x_{11}, x_{21}) = 3, N(x_{11}, x_{22}) = 1, N(x_{12}, x_{21}) = 3, N(x_{12}, x_{22}) = 3, N(x_{13}, x_{21}) = 2$   
and

$$N(x_{13}, x_{22}) = 1.$$

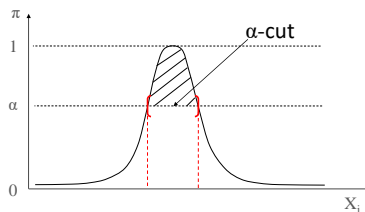
$$\hat{\pi}^n(x_{11}, x_{21}) = \frac{3}{3}, \hat{\pi}^n(x_{11}, x_{22}) = \frac{1}{3}, \hat{\pi}^n(x_{12}, x_{21}) = \frac{3}{3}, \hat{\pi}^n(x_{12}, x_{22}) = \frac{3}{3}, \hat{\pi}^n(x_{13}, x_{21}) = \frac{2}{3} \text{ and } \hat{\pi}^n(x_{13}, x_{22}) = \frac{1}{3}.$$

## 1.5 Variable sampling

In the probabilistic case, sampling a probability distribution consists in generating a random sample representative and in proportion to its probability distribution. In the possibilistic case, sampling a variable corresponds to the generation of a data set representative of its possibility distribution. This notion makes sense only in the numerical interpretation since it is based on random generation of values pertaining to the universe of discourse which contradicts the semantic of the qualitative aspect of possibility theory. So, in the numerical interpretation, two approaches (Chanas and Nowakowski, 1988; Guyonnet et al., 2003) have been proposed to sample a variable  $X_i$  in the possibilistic framework. These methods are based on  $\alpha$ -cut notion proposed by Zadeh (Zadeh, 1975) defined as follows:

$$\alpha\text{-cut}_{X_i} = \{x_{ik} \in D_i \text{ s.t. } \pi(x_{ik}) \geq \alpha\} \quad (1.16)$$

where  $\alpha$  is randomly generated from  $[0,1]$ .

Figure 1.2:  $\alpha$ -cut notion

The epistemic sampling method proposed by Guyonnet et al. (2003) focuses on the generation of imprecise data by returning all values of  $\alpha\text{-cut}_{X_i}$  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 (Chanas and Nowakowski, 1988) which is dedicated to the generation of precise data from the pignistic probability distribution by returning a single value uniformly chosen from  $\alpha$ -cut $_{X_i}$ .

**Example 1.5.1.** Let  $X_1$  be a ternary variable such that  $\pi(x_{11}) = 0.1$ ,  $\pi(x_{12}) = 1$  and  $\pi(x_{13}) = 0.8$ .  
 If  $\alpha = 0.1$ ,  $\alpha$ -cut $_{X_1} = \{x_{11}, x_{12}, x_{13}\}$ .  
 If  $\alpha = 0.7$ ,  $\alpha$ -cut $_{X_1} = \{x_{12}, x_{13}\}$ .  
 If  $\alpha = 1$ ,  $\alpha$ -cut $_{X_1} = \{x_{12}\}$ .

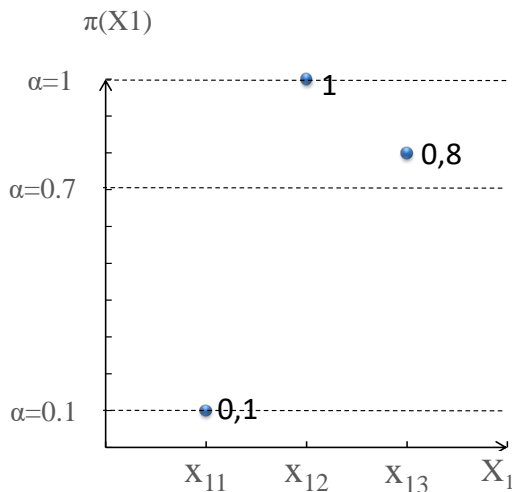


Figure 1.3: Example of  $\alpha$ -cut

Note that the fact that  $\alpha$ -cut returns the set of values whose degrees of possibility higher than  $\alpha$  leads to the construction of data which have to be represented as consonant sets as shown by Example 1.5.1. In fact,  $x_{12}$  is the most possible state so, every returned  $\alpha$ -cut contains this state. Thereby, a less possible state can not appear in the data without being regrouped with all the states with higher degrees of possibility.

## 1.6 Possibilistic similarity measures

The concept of similarity/dissimilarity is fundamentally important in almost every scientific field. From the scientific and mathematical point of view, *similarity measure* is defined as a quantitative degree of how close two objects are. This concept has attracted a lot of attention in probability theory (Kullback and Leibler, 1951; Chan and Darwiche, 2005) to compare two probability distributions. In the possibilistic case, it reflects the degree of closeness between two possibility distributions  $\pi_1$  and  $\pi_2$  defined on the same universe of discourse. As far as we know, there is only one attempt that has been made to propose a possibilistic similarity measure (Jenhani et al., 2007). This section reviews some existing similarity and distance measures that can be used in the possibilistic framework.

### 1.6.1 Information closeness

The information closeness measure (Higashi and Klir, 1983) is one of first works devoted to measure similarity between two possibility distributions based on non-specificity. In the following, we denote  $G(\pi_1, \pi_2)$  the value of information closeness between two distributions  $\pi_1, \pi_2$ .  $\vee$  refers to the maximum operator and  $nsp(\pi)$  is given by equation(1.5).

$$G(\pi_1, \pi_2) = g(\pi_1, \pi_1 \vee \pi_2) + g(\pi_2, \pi_1 \vee \pi_2) \tag{1.17}$$

where  $g(\pi_1, \pi_2) = nsp(\pi_2) - nsp(\pi_1)$ . Thus, the information closeness can also be written as:

$$G(\pi_1, \pi_2) = 2 * nsp(\pi_1 \vee \pi_2) - nsp(\pi_1) - nsp(\pi_2). \quad (1.18)$$

**Example 1.6.1.** Let us consider the following possibility distributions over the same universe of discourse  $\pi_1 = \{0.2, 0.9, 0.3, 1\}$  and  $\pi_2 = \{0.1, 0.5, 1, 0\}$ . Using Equation 1.18, we obtain:

$$G(\pi_1, \pi_2) = 2 * nsp\{1, 1, 0.9, 0.2\} - nsp\{0.2, 0.9, 0.3, 1\} - nsp\{0.1, 0.5, 1, 0\} = 2 * 1.6094 - 1.1584 - 0.5584 = 1.502$$

## 1.6.2 Sangüesa et al's distance

Sangüesa et al's distance (Sangüesa and Cortés, 2000), denoted by *distance*, computes possibility distributions non-specificity difference and is expressed by:

$$distance(\pi_1, \pi_2) = nsp(|\pi_1(x_{ik}) - \pi_2(x_{ik})|) \quad \forall (x_{ik} \in D_i) \quad (1.19)$$

**Example 1.6.2.** Let us reconsider possibility distributions in Example 1.6.1. Using Equation 1.19, we obtain:

$$distance(\pi_1, \pi_2) = nsp\{0.7, 0.4, 0.1\} = 0.9754$$

## 1.6.3 Minkowski distance

In the possibilistic case, we can use the well known Minkowski distance (Riesz, 1910; Dunford et al., 1971), denoted by *MinD*, which includes others as special cases of the generalized form: Manhattan, Euclidean and Maximum distance (or Chebyshev distance or chessboard distance). Similarity measures that can be derived from them are denoted respectively by *MS*, *ES* and *CS*. Minkowski distance between two possibility distributions  $\pi_1$  and  $\pi_2$  is expressed as follows:

$$MinD(\pi_1, \pi_2) = \sqrt[p]{\sum_{k=1}^m |\pi_1(x_{ik}) - \pi_2(x_{ik})|^p} \quad (1.20)$$

Special cases of Minkowski distance are:

- Normalized Manhattan distance, denoted by *MD*, is defined by:

$$MD(\pi_1, \pi_2) = \frac{\sum_{k=1}^m |\pi_1(x_{ik}) - \pi_2(x_{ik})|}{m} \quad (1.21)$$

- Normalized Euclidean distance, denoted by *ED*, is defined by.

$$ED(\pi_1, \pi_2) = \sqrt{\frac{\sum_{k=1}^m (\pi_1(x_{ik}) - \pi_2(x_{ik}))^2}{m}} \quad (1.22)$$

- Normalized Maximum distance, denoted by *CD*, is defined by:

$$CD(\pi_1, \pi_2) = \max_{k=1}^m |\pi_1(x_{ik}) - \pi_2(x_{ik})| \quad (1.23)$$

**Example 1.6.3.** Let us reconsider possibility distributions in Example 1.6.1. Let us compute the distance between the  $\pi_1$  and  $\pi_2$  using distances listed above given by equations 1.21 1.22 and 1.23, respectively:

- $MD(\pi_1, \pi_2) = \frac{0.1+0.4+0.7+1}{4} = 0.55$ .
- $ED(\pi_1, \pi_2) = \sqrt{\frac{0.1^2+0.4^2+0.7^2+1^2}{4}} = 0.64$ .
- $CD(\pi_1, \pi_2) = \max(0.1, 0.4, 0.7, 1) = 1$ .

## 1.6.4 Information affinity

Jenhani et al. (2007) have shown that measuring similarity between two possibility distributions depends on two main criteria distance and inconsistency and have proposed a measure named information affinity (Jenhani et al., 2007) and expressed as follows:

$$Aff(\pi_1, \pi_2) = 1 - \frac{\kappa * MD(\pi_1, \pi_2) + \lambda * Inc(\pi_1, \pi_2)}{\kappa + \lambda} \quad (1.24)$$

where  $\kappa > 0$  and  $\lambda > 0$ . Information affinity is based on two quantities: inconsistency degree  $Inc(\pi_1, \pi_2) = Inc(\pi_1 \wedge \pi_2)$  ( $\wedge$  can be taken as min or product operator<sup>1</sup>) and the normalized Manhattan distance expressed by Equation 1.21 or the normalized euclidean distance expressed by Equation 1.22.

Note that in the remaining, we use Manhattan distance such that  $\wedge$  is the min operator. Since the use of the min operator means that we give less importance to the inconsistency degree, we choose to fix  $\kappa = \lambda = 1$  to avoid penalizing twice the consistency degree.

**Example 1.6.4.** Let us reconsider possibility distributions in Example 1.6.1. Using Equation 1.24, we obtain:  $Aff(\pi_1, \pi_2) = 1 - \frac{1*0.55+1*0.5}{1+1} = 0.475$ .

Note that only Minkowski distance and information affinity satisfy basic properties of a distance i.e. non-negativity, symmetry, upper bound and non-degeneracy, lower bound, large inclusion and permutation (for more details, see (Jenhani et al., 2007)).

## 1.7 Possibility theory and links with other uncertainty theories

In this section, we will briefly describe the links between possibility theory and some established uncertainty frameworks.

### 1.7.1 Possibility theory vs probability theory

Probability theory is considered as the standard uncertainty theory proposed to handle uncertain information and is based on the notion of probability distribution which is a mapping  $p : D_i \rightarrow [0, 1]$  satisfying  $\sum_{x_{ik} \in D_i} p(x_{ik}) = 1$ . Given a probability distribution  $p$ , we can define a probability measure of any subset  $A \subseteq D_i$  by:

$$P(A) = \sum_{x_{ik} \in D_i} p(x_{ik}). \quad (1.25)$$

In the probabilistic setting, a probability distribution  $p$  is transformed into a new probability distribution by the arrival of a new fully certain piece of information  $\Phi \subseteq \Omega$ , as follows:

$$p(\omega | \Phi) = \begin{cases} \frac{p(\omega)}{P(\Phi)} & \text{if } \omega \in \Phi \\ 0 & \text{otherwise.} \end{cases} \quad (1.26)$$

Given three disjoint subsets of variables,  $X$ ,  $Y$  and  $Z$  pertaining to  $V$ , the probabilistic conditional independence between  $X$  and  $Y$  in the context  $Z$ , denoted by  $I_{Prob}(X, Z, Y)$ , is expressed by:

$$P(X | YZ) = P(X | Z), \forall X, Y, Z, \quad (1.27)$$

or equivalently,

$$P(XY | Z) = P(X | Z) * P(Y | Z), \forall X, Y, Z. \quad (1.28)$$

This means that  $X$  is considered as Prob-independent from  $Y$  in the context  $Z$ .

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

One view of possibility theory is to consider a possibility distribution as a family of probability distributions (Dubois, 2006) for which the measure of each subset  $A$  of  $D_i$  will be respectively lower and upper bounded by its necessity and its possibility measures. More formally, if  $\mathcal{P}$  is the set of all probability distributions defined on  $D_i$ , the family of probability distributions  $\mathcal{P}_\pi$  associated with  $\pi$  is defined as follows:

$$\mathcal{P}_\pi = \{p \in \mathcal{P}, \forall A \subseteq D_i, N(A) \leq P(A) \leq \Pi(A)\} \quad (1.29)$$

In order to describe different transformations, several properties were proposed in literature which are:

- **Consistency condition:**  $\pi$  and  $p$  satisfy the consistency condition if  $\Pi$  can be seen as an upper-bound of  $P$ .

$$P(A) < \Pi(A), \forall A \subseteq D_i \quad (1.30)$$

- **Preference preservation:** The preference preservation ensures that the obtained possibility distribution has the same form as the initial probability distribution. Formally,  $\forall (x_{ik}, x_{ij}) \in D_i^2$ ,

$$p(x_{ik}) > p(x_{il}) \implies \pi(x_{ik}) > \pi(x_{ij}) \quad (1.31)$$

$$p(x_{ik}) = p(x_{il}) \implies \pi(x_{ik}) = \pi(x_{ij}) \quad (1.32)$$

- **Maximum specificity:** This principle ensures that between two possibility distributions, the most specific one should be chosen.

Several researches have been proposed to transform a probability distribution into a possibility one (Klir and Parviz, 1992; Dubois et al., 1993, 2004; Mouchaweh et al., 2006; Bouguelid, 2007). Transforming probabilistic distributions to possibilistic ones is useful when weak source of information makes probabilistic data unrealistic, to reduce the complexity of the solution or to combine different types of data. In what follows, we will cite the most common probability possibility transformations. Note that all of them make sense in the numerical interpretation of the possibilistic scale. In the ordinal interpretation, some methods have been proposed to estimate a possibility distribution from infinitesimal probabilities (Giang and Shenoy, 1999; Henrion et al., 1994; Darwiche and Goldszmidt, 1994), for more details see (Sabbadin, 2001).

Let  $P(X_i)$  be a probability distribution relative to a variable  $X_i$  and  $\{p_{(1)}, p_{(2)}, \dots, p_{(m)}\}$  is the descending order of  $P(X_i)$ .

- **Klir Transformation** (Klir and Parviz, 1992):

$$\pi_{(k)} = \frac{p_{(k)}}{p_{(1)}} \quad (1.33)$$

Note that possibility distributions obtained using Klir transformation recovers the distributions obtained using Equation 1.15. Moreover, this transformation generally violates the consistency principle as shown in (Dubois and Prade, 2016).

- **Optimal Transformation** (Dubois et al., 2004) also called asymmetric transformation:

$$\pi_{(k)} = \sum_{j/p_{(j)} \leq p_{(k)}} p_{(j)} \quad (1.34)$$

This transformation is optimal because it gives the most specific distribution i.e. that loses less information, and it is asymmetric since we can not recover the the probability distribution from the obtained possibility one.

- **Symmetric Transformation** (ST) (Dubois et al., 1993):

$$\pi_{(k)} = \sum_{j=1}^m \min(p_{(k)}, p_{(j)}) \quad (1.35)$$

This transformation corresponds to the inverse of the pignistic transformation (Smets, 1989) i.e. it provides a subjective possibility distribution, the least committed distribution assuming minimal statistical knowledge.



– **Variable Transformation (VT)** (Mouchaweh et al., 2006):

$$\pi_{(k)} = \left( \frac{P_{(k)}}{p_{(1)}} \right)^{c \cdot (1-p_{(k)})} \quad (1.36)$$

where  $c$  is a constant belonging to the interval:  $0 \leq c \leq \frac{\log p_{(m)}}{(1-p_{(m)}) \cdot \log(\frac{p_{(m)}}{p_{(1)}})}$ . This transformation needs

less computation than the asymmetric transformation. However, the maximum specificity principle and the preference preservation are not always satisfied.

**Example 1.7.1.** Let us consider the quaternary variable  $X_1$  defined on  $D_1 = \{x_{11}, x_{12}, x_{13}, x_{14}\}$  such that:  $p(x_{11}) = 0.2, p(x_{12}) = 0.35, p(x_{13}) = 0.4$  and  $p(x_{14}) = 0.05$ . By transforming  $p$  into  $\pi$  using different transformations described above, we obtain:

- **KT:**  $\pi(x_{11}) = 0.2/0.4 = 0.5, \pi(x_{12}) = 0.35/0.4 = 0.875, \pi(x_{13}) = 0.4/0.4 = 1, \pi(x_{14}) = 0.05/0.4 = 0.125$ .
- **OT:**  $\pi(x_{11}) = 0.05+0.2 = 0.25, \pi(x_{12}) = 0.05+0.2+0.35 = 0.6, \pi(x_{13}) = 0.05+0.2+0.35+0.4 = 1, \pi(x_{14}) = 0.05$ .
- **ST:**  $\pi(x_{11}) = 0.05 + 0.2 + 0.2 + 0.2 = 0.65, \pi(x_{12}) = 0.05 + 0.2 + 0.35 + 0.35 = 0.95, \pi(x_{13}) = 0.05 + 0.2 + 0.35 + 0.4 = 1, \pi(x_{14}) = 0.05 + 0.05 + 0.05 + 0.05 = 0.2$ .
- **VT:** if  $c = \frac{\log 0.05}{(1-0.05) \cdot \log \frac{0.05}{0.4}} = 1.51, \pi(x_{11}) = \left(\frac{0.2}{0.4}\right)^{1.51 \cdot (1-0.2)} = 0.43, \pi(x_{12}) = \left(\frac{0.35}{0.4}\right)^{1.51 \cdot (1-0.35)} = 0.87, \pi(x_{13}) = 1, \pi(x_{14}) = \left(\frac{0.05}{0.4}\right)^{1.51 \cdot (1-0.05)} = 0.05$ .

Note that if the initial probability distribution are frequencies of states computed from data, the most suitable transformation is the optimal one and if the initial probability distribution is subjective, we use the symmetric transformation (for more details see (Dubois and Prade, 2016))

## 1.7.2 Possibility theory vs belief function theory

Other uncertainty theories have already been formalized such as the case of belief function theory (Shafer, 1976; Smets, 1988). It encodes our knowledge by a *basic belief assignment* also called a mass function which corresponds to the mapping  $m : 2^{\text{card}(D_i)} \mapsto [0, 1]$  such that  $\sum_{A_{ik} \subseteq D_i} (m(A_{ik})) = 1$ . A focal element  $A_{ik} \subseteq D_i$  such that  $m(A_{ik}) > 0$  is called a focal set.

The total amount of belief committed to any event  $A_{ik}$  is expressed by a *belief function*:  $Bel : 2^{\text{card}(D_i)} \rightarrow [0, 1]$ , defined for any  $A_{ik} \subseteq D_i$  by:

$$Bel(A_{ik}) = \sum_{A_{jk} \subseteq A_{ik}, A_{jk} \neq \emptyset} m(A_{jk}) \quad (1.37)$$

$$Bel(\emptyset) = 0$$

The *plausibility function*  $Pl : 2^{\text{card}(D_i)} \rightarrow [0, 1]$  quantifies the degree of plausibility that the actual world belongs to  $A_{ik}$ . For any  $A_{ik} \subseteq D_i$  by  $Pl(A_{ik})$  is expressed by:

$$Pl(A_{ik}) = 1 - Bel(\neg A_{ik}). \quad (1.38)$$

If all the focal elements of  $m$  are singletons, then  $m$  is a Bayesian basic belief assignment and  $m = Bel = Pl$ .

Several definitions of conditioning are developed in evidence theory, we give here the expression of the Dempster's rule of conditioning, let be  $\Phi \subseteq \Omega$ , the rule of conditioning is expressed as follows:

$$Pl(\omega | \Phi) = \frac{Pl(\omega \cap \Phi)}{Pl(\Phi)} \quad (1.39)$$

In evidence theory framework, if the focal elements  $A_{i1}, \dots, A_{im}$  are nested (i.e.,  $A_{i1} \subseteq A_{i2} \subseteq \dots \subseteq \phi_{A_{im}}$ ), then the belief function  $Bel$  is called a *consonant belief function* and for all  $A_{ik}, A_{jk} \subseteq D_i$ , we have:

$$\begin{aligned} Bel(A_{ik} \wedge A_{jk}) &= \min(Bel(A_{ik}), Bel(A_{jk})); \text{ and} \\ Pl(A_{ik} \vee A_{jk}) &= \max(Pl(A_{ik}), Pl(A_{jk})) \end{aligned}$$

It is stated that in this case *belief functions* are *necessity measures* and *plausibility functions* are *possibility measures* i.e.  $Bel = N$  and  $Pl = \Pi$ .

Let  $m$  be a basic belief assignment relative to a variable  $X_i$  with nested focal elements  $A_{i1}, \subseteq A_{i2}, \subseteq \dots, \subseteq \phi_{A_i m}$ . The possibility distribution  $\pi$  is derived using the following equation:

$$\pi(x_{ik}) = \sum_{A_{ik} | x_{ik} \in A_{ik}} m(A_{ik}) \quad (1.40)$$

A random set is said to be *consistent* if there is at least one element  $x_{ik}$  contained in all focal sets  $A_{ik}$  and the possibility distribution induced by a consistent random set is, thereby, normalized.

Note that exploring this link between possibility theory and random sets theory has been extensively studied, in particular, in learning tasks, we cite for instance (Borgelt et al., 2009; Joslyn, 1997). In the latter, the possibility degree of an element is a *contour function* (Shafer, 1976) of a random set and corresponds the probability of the possibility of the event i.e. the probability of the disjunction of all events (focal sets) in which this event is included.

**Example 1.7.2.** Let  $X_1$  be a quaternary variable and  $m$  such that  $m(x_{13}) = 0.6$ ,  $m(x_{11}x_{13}) = 0.2$ ,  $m(x_{11}x_{13}, x_{14}) = 0.1$  and  $m(D_i) = 0.1$ . Using Equation 1.40, we obtain:

$$\pi(x_{11}) = 0.2 + 0.1 + 0.1 = 0.4.$$

$$\pi(x_{12}) = 0.1.$$

$$\pi(x_{13}) = 0.6 + 0.2 + 0.1 + 0.1 = 1.$$

$$\pi(x_{14}) = 0.1 + 0.1 = 0.2.$$

However, it is important to note that in this case, the Dempster rule of conditioning defined by (1.39) corresponds to the product-based conditioning defined by Equation (1.7) and not to the min-based one.

### 1.7.3 Possibility theory vs imprecise probability theory

Contrarily to already presented frameworks in which uncertainty is modeled by a function that maps a state or a subset of states of the world to a *single* value, imprecise probability theory (Dempster, 1967; Augustin et al., 2014) represents uncertainty by an interval specification/probability interval (IP). More formally, to each event  $A$  is attached a probability interval  $[P_*(A), P^*(A)]$  such that:

- $P_*(A) = \inf\{P(A), P \in \mathcal{P}\}$
- $P^*(A) = \sup\{P(A), P \in \mathcal{P}\} = 1 - P_*(\bar{A})$

As far as we know, few researches have been proposed to transform an imprecise probability distribution into a possibility one (De Campos and Huete, 2001; Masson and Denœux, 2006; Hou and Yang, 2010; Destercke et al., 2007).

De Campos and Huete (2001) proposed adaptations of Klir, optimal and symmetric transformations for graphical models to support the joint possibility distribution decomposition by considering conditioning and marginalization. Adaptations of Klir, optimal and symmetric transformations are expressed as follows: Let  $P(X_i)$  be a probability distribution relative to a variable  $X_i$  and  $\{p_{(1)}, p_{(2)}, \dots, p_{(m)}\}$  is the descending order of  $P(X_i)$  ( $p_{(1)}$  is the highest degree and  $p_{(m)}$  is the smallest one),  $u_{(k)} = \min(p_{(k)} + \frac{c_\epsilon}{\sqrt{N}} \sqrt{p_{(k)}(1 - p_{(k)})}, 1)$ ,  $l_{(k)} = \max(p_{(k)} - \frac{c_\epsilon}{\sqrt{N}} \sqrt{p_{(k)}(1 - p_{(k)})}, 0)$ ,  $c_\epsilon = \frac{1}{100} * \frac{1+c_\epsilon}{2}$  and  $N$  be the number of observations in the database.

- Klir transformation adaptation is expressed as follows:

$$\pi_{(k)} = \min\left(\frac{u_{(k)}}{\max_{j=1..m}(l_{(j)}, 1)}\right) \quad (1.41)$$

– Optimal transformation adaptation is expressed as follows:

$$\pi_{(k)} = \min\left(\sum_{j=1}^k u_{(j)}, 1\right) \quad (1.42)$$

– Symmetric transformation adaptation is expressed as follows:

$$\pi_{(k)} = \min\left(\sum_{j=1}^m \min(u_{(k)}, u_{(j)}), 1\right) \quad (1.43)$$

Note that all of these transformations make sense in the case of a numerical interpretation of the possibilistic scale. However, [De Campos and Huete \(2001\)](#) suggests that their transformations could be applied in the ordinal case, but basing on principles that [Dubois \(2006\)](#) restricted to the numerical interpretation.

The underlying idea of ([Masson and Denœux, 2006](#); [Hou and Yang, 2010](#); [Destercke et al., 2007](#)) is inferring possibility distributions from interval based probability distribution. Transformations proposed in ([Masson and Denœux, 2006](#); [Hou and Yang, 2010](#)) construct a possibility distribution dominating all the probability measures defined by IP and satisfy the consistency condition and preference preservation properties (cf. Section 1.7.1). These transformations compute from each possible linear extension (a linear extension  $C_l \subseteq \mathcal{C}$  is a complete order that is compatible with the partial order  $\mathcal{M}$  induced by the considered IP) a possibility distribution, then, they take the one dominating all the probability measures defined by IP. Note that [Masson and Denœux \(2006\)](#) address possibility distribution inference from large datasets and ([Hou and Yang, 2010](#)) is dedicated to small ones. The two methods converge to the optimal transformation in the case we have sufficient data. There are two main drawbacks with these two transformations: i) its computational cost since it considers in the worst case  $N!$  linear extensions where  $N$  is the size of the distribution to transform and ii) the fact that this transformation results in a loss of information and does not guarantee that the obtained distribution is optimal in terms of specificity ([Destercke et al., 2007](#)).

[Destercke et al. \(2007\)](#) suggest that any upper generalized R-cumulative distribution  $\bar{F}$  build from one linear extension  $C_l$  can be viewed as a possibility distribution and ensures that the obtained distribution dominates all the probability measures defined by IP  $[l_i, u_i]$ . Let  $\phi_1, \phi_2, \dots, \phi_n$  be subsets of  $D_i$  such that  $\phi_i = \{x_{ij} \leq_{C_l} x_{ik}\}$ . The upper cumulative distribution  $\bar{F}$  from one linear extension  $C_l$  is computed as follows:

$$\bar{F}(\phi_i) = \min\left(\sum_{x_{ij} \in \phi_i} u_j, 1 - \sum_{x_{ij} \notin \phi_i} l_j\right) \quad (1.44)$$

## 1.8 Conclusion

In this chapter we have presented basic concepts of possibility theory, a non classical theory of uncertainty that offers an appropriate framework to handle uncertainty qualitatively and quantitatively. Then, we have introduced many important concepts relative to possibility theory such as possibility distribution, non-specificity, sampling, similarity etc. Finally, we have briefly reviewed some uncertainty theories, namely, probability theory, evidence theory and imprecise probability theory and we have discussed their link with possibility theory.

Next chapter discusses graphical representations of knowledge under uncertainty and we will mainly focus on possibilistic networks and their learning from imperfect data.



# Graphical representation of knowledge in uncertain frameworks

## Contents

<b>2.1</b>	<b>Introduction</b>	<b>21</b>
<b>2.2</b>	<b>Background and notations on graphs</b>	<b>22</b>
<b>2.3</b>	<b>Bayesian networks</b>	<b>22</b>
2.3.1	Definition	22
2.3.2	Propagation, applications and software solutions	23
2.3.3	Learning from data	24
<b>2.4</b>	<b>Evidential networks</b>	<b>26</b>
2.4.1	Definition	26
2.4.2	Propagation, applications and software solutions	27
2.4.3	Learning from data	27
<b>2.5</b>	<b>Credal networks</b>	<b>27</b>
2.5.1	Definition	27
2.5.2	Propagation, applications and software solutions	28
2.5.3	Learning from data	28
<b>2.6</b>	<b>Possibilistic networks</b>	<b>28</b>
2.6.1	Definition	28
2.6.2	Inference, applications and software solutions	30
<b>2.7</b>	<b>Learning possibilistic networks from data</b>	<b>30</b>
2.7.1	Constraint-based methods	31
2.7.2	Score-based methods	31
2.7.3	Hybrid methods	32
2.7.4	Discussion	32
<b>2.8</b>	<b>Conclusion</b>	<b>33</b>

## 2.1 Introduction

Over the last three decades, a lot of effort has been put into learning graphical models from data but most of the proposed methods are relative to probabilistic models. In particular, Bayesian networks (Pearl, 1988;

Jensen, 1996) have been widely studied and used in real applications (Pourret et al., 2008). However, where imprecision is inherent to the studied domain or where available information are simply preferences, non-classical uncertainty theories such as possibility theory (Zadeh, 1978) and evidence theory (Shafer, 1976) stand out as best alternatives to probability theory (Dubois and Prade, 2006). Therefore, other graphical models have been proposed to model and reason with this form of imperfect information. Among these models, we are interested by possibilistic networks representing the possibilistic counterpart of Bayesian networks.

In this chapter, we will present three graphical reasoning models namely Bayesian, evidential and credal networks by giving their formal descriptions and reviewing their inference and learning methods. Then, we will focus on possibilistic networks, in particular, their learning from imperfect data.

This chapter is organized as follows: Sections 2.3, 2.5 and 2.4 briefly present Bayesian networks, credal networks and evidential networks. Section 2.6 and 2.7 are dedicated to possibilistic networks and how to learn them from data.

## 2.2 Background and notations on graphs

We first give some notations that will be used in the remaining. Let  $V = \{X_1, X_2, \dots, X_n\}$  be a finite set of variables.  $G = (V, E)$  is said to be a **graph** on  $V$  and  $E$  corresponds to the set of **edges** connecting some pairs of nodes in  $V$ . If the edges in  $E$  are directed then they are called **edges** and  $\mathcal{G} = (V, E)$  is said to be a **directed graph**.

- For each edge  $X_1X_2$ , the node  $X_1$  is called its origin and  $X_2$  its end.
- In an edge  $X_1X_2$ , the node  $X_1$  is the **parent** of  $X_2$  and the node  $X_2$  is the **child** of  $X_1$ .
- $Pa(X_i)$  is the sets of parents of a variable  $X_i$ .
- A **path** in a directed graph is a sequence of nodes from one node to another using the edges.
- The path  $X_1 \rightarrow X_2 \rightarrow X_3$  is called a serial connection.
- $X_1 \leftarrow X_2 \rightarrow X_3$  is called a diverging connection.
- $X_1 \rightarrow X_2 \leftarrow X_3$  is called a v-structure.
- A **cycle** is a path visiting each node once and having the same first and last node.
- A **loop** is an *undirected* cycle.
- A **DAG** is a Directed Acyclic (without cycles) Graph.
- The skeleton of a directed graph is the same underlying undirected graph.
- A **singly connected DAG or polytree** is a DAG which contains no loops, in this case the graph obtained by dropping the directions of the links is a tree.
- A **multiply connected DAG** is a DAG which can contain loops.
- **D-separation criterion:** Let be  $Z \subseteq V$ ,  $X$  and  $Y$  two disjoint subsets in  $V \setminus Z$ .  $X$  and  $Y$  are said to be *d-separated* by  $Z$  if  $X$  and  $Y$  are independent given  $Z$ .

## 2.3 Bayesian networks

In this section, we give a formal definition of Bayesian networks (Pearl, 1988; Jensen, 1996). Then, we discuss information propagation in such models, their application in real world and available software solutions. Finally, we review some existing learning methods.

### 2.3.1 Definition

Bayesian networks represent a powerful tool for reasoning and modeling complex domains. A Bayesian network over a set of variables  $V$  consists of two components:

- A *graphical* or *qualitative component* composed of a DAG which encodes a set of independence relations satisfying the Markov assumption, i.e., each variable  $X_i \in V$  is conditionally independent of its non-descendent given its parents.
- A *numerical* or *quantitative component* corresponding to the set of conditional probability distributions relative to each node  $X_i \in V$  in the context of its parents i.e.  $p(X_i | Pa(X_i))$ .

Given a Bayesian network, the global joint probability distribution over the set  $V = \{X_1, \dots, X_N\}$  can be expressed as a product of the  $n$  initial conditional probabilities via the following probabilistic *chain rule*:

$$p(X_1, \dots, X_n) = \prod_{i=1..n} p(X_i | Pa(X_i)). \quad (2.1)$$

Note that two different DAGs can represent the same set of conditional independence relations, and hence distributions. Such graphs are said to be Markov equivalent.

**Definition 2.3.1.** *Two Bayesian networks BN1 and BN2 are equivalent if they encode the same probability distribution.*

Judea Pearl (1991) have shown that two equivalent DAGs satisfy two graphical conditions expressed through the following theorem:

**Theorem 2.3.1.** *Two DAGs are equivalent if and only if they have the same skeleton and the same v-structures.*

**Example 2.3.1.** *Let us consider the Figure 2.1.*

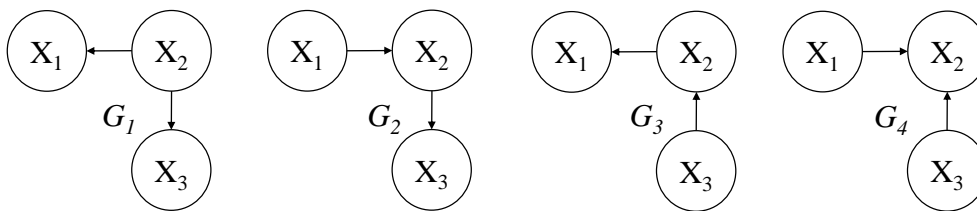


Figure 2.1: Markov equivalence

*In this example,  $G_1$ ,  $G_2$  and  $G_3$  are equivalent and are not equivalent to the v-structure in  $G_4$ . In fact, we can easily demonstrate that:*

$$P(X_1, X_2, X_3)_{G_1} = P(X_1, X_2, X_3)_{G_2} = P(X_1, X_2, X_3)_{G_3} \neq P(X_1, X_2, X_3)_{G_4}.$$

A Markov equivalence class is defined as a set of equivalent Bayesian networks and represented via a Completed Partially Directed Acyclic Graph (CPDAG) having the same skeleton as all the graphs in the equivalence class and all its reversible edges (edges that do not belong to a v-structure and their inversion does not generate a v-structure) are undirected.

### 2.3.2 Propagation, applications and software solutions

Inference is one of the most important tasks we can process on Bayesian networks. It benefits from advantages of joint distribution decomposition into a product of local probability distributions of each variable depending only on its parents to study the impact of some partially observed variables known as *evidence* on remaining ones. This problem is classified as NP-hard except for poly-trees (singly connected graphs) where inference can be performed in polynomial time (Cooper, 1990). Exact inference methods could be mainly classified into two families: message passing algorithms are performed on the particular case of singly connected DAGs (Pearl, 1988; Kim and Pearl, 1983). It consists in combining information deriving

from parents and children of the variable of interest via a message passing mechanism. The second family of methods are dedicated to more generic DAGs i.e. multiply connected networks. These algorithms transform the DAG into a junction tree, composed of cliques of variables, using moralization and triangulation techniques. Then, message passing is performed between obtained cliques (Jensen, 1996; Lauritzen and Spiegelhalter, 1988). Several other works have been proposed to solve this problem either exact or approximate. For exact algorithms, we cite for instance symbolic probabilistic inference (Li and d'Ambrosio, 1994) and polynomial compilation (Chavira and Darwiche, 2007). For approximate methods, we cite for instance Monte Carlo methods (Henrion, 1986a) and Logic sampling (Dagum and Horvitz, 1993).

Bayesian networks are increasingly popular as reasoning tools among those researching the use of artificial intelligence, probability and uncertainty. Thereby, huge number of Bayesian networks based real world applications have been developed in various disciplines we cite as examples: *DIAVAL* in medical diagnosis (Diez et al., 1997), *PINS IN MAPS* in crime risk factors analysis (Oatley and Ewart, 2003), *FINEX* (Forensic Identification by Network Expert systems) in forensic science (Cowell, 2003), *ADVOCATE* (Advanced On-board Diagnosis and Control of Autonomous Systems) in risk management in robotics (Sotelo et al., 2003). Readers interested in real applications may consult (Pourret et al., 2008; Korb and Nicholson, 2010).

Given the widespread popularity of Bayesian networks, it is evident that several softwares and libraries have been developed to manipulate these reasoning models e.g. *Bayesia*<sup>1</sup>, *Netica*<sup>2</sup> and *BNT toolbox* (Murphy, 2014).

### 2.3.3 Learning from data

Learning Bayesian networks has been widely studied and various approaches were proposed to learn both DAG structure and parameters. The most common way to learn parameters from complete data can be easily performed using either the statistical approach or the Bayesian one (Heckerman, 1998). The statistical approach consists on finding the most likely values for a model parameters given a dataset and an underlying model. This is done by maximizing the likelihood function defined as follows:

Given a dataset  $\mathcal{D}$ , a DAG  $G$  and the parameters  $\Theta = \{\theta_1, \theta_2, \dots, \theta_n\}$  relative to  $\{X_1, X_2, \dots, X_n\}$  to be estimated.

The likelihood function is expressed by:

$$L(\Theta, G, \mathcal{D}) = \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \quad (2.2)$$

where for each  $X_i$   $q_i$  is  $\text{card}(Pa(X_i))$  and  $r_i = \text{card}(D_i)$ ,  $\theta_{ijk}$  is the parameter to be estimated when  $X_i = x_{ik}$  and  $Pa(X_i) = x_j$ .

Note that learning Bayesian networks parameters from imperfect data addresses the particular case of missing data. In this case, the most common method is the standard Expectation Maximization (EM) (Lauritzen, 1995).

Existing algorithms to learn the structure of Bayesian networks can be classified into constraint-based approaches, score-based approaches and hybrid methods.

#### Constraint-based approaches

The first family of approaches considers the learning task as a constraint satisfaction problem which consists in identifying conditional independence relations among variables using a statistical hypothesis test, such as  $\chi^2$  test (Chernoff and Lehmann, 1954). Then, it constructs a network that exhibits covered independence relations.

1. <http://www.bayesialab.com/>

2. [http://www.norsys.com/netica\\_c\\_api.htm#download](http://www.norsys.com/netica_c_api.htm#download)



All constraint-based methods such as IC (Inductive Causation) (Pearl and Verma, 1995), PC (Peter and Clark, the inventor names of the algorithm) (Spirtes et al., 2000) and PMMS (Polynomial Max-Min Skeleton) (Brown et al., 2005), are based on the same principle described as follows:

1. Construct an undirected graph detecting all relationships between variables using independence tests i.e. an undirected graph representing all detected dependences.
2. Detect v-structures in the obtained undirected graph ( $X_j \rightarrow X_i \leftarrow X_k$ ) also using independence tests.
3. Orient edges based on a set of orientation rules related to DAG constraints for instance, cycle avoidance.

Note that the two first steps are based on reasoning about conditional independence facts and learns the Markov equivalence class rather than the fully DAG structure.

### Score-based approaches

Contrarily to constraint-based approaches, score-based ones consider the learning task as an optimization problem i.e. it looks for the structure that maximizes a score that assesses the fitness between each possible structure and available data, or looks for the best sub-structures and combines them. So, a score-based method usually consists of:

- a scoring function (to assess the quality of a given network),
- a search method (to traverse the space of possible networks).

**Scoring functions:** Most of scores proposed in the literature, e.g. Akaike information criterion (AIC) (Akaike, 1970), Bayesian information criterion (BIC) (Cooper and Herskovits, 1992) and minimum description length (MDL) (Bouckaert, 1993), are based on the Occam's razor principle: find the most simple model that fits the data.

For instance, the AIC score presents a good compromise between likelihood and complexity and is composed of two components: the likelihood of the structure given the data (Equation 2.2) and its complexity which is controlled via its dimension denoted by  $Dim(G)$  and expressed by:

$$Dim(G) = \sum_{i=1}^n Dim(X_i, G) \quad (2.3)$$

where  $Dim(X_i, G) = (r_i - 1) * \prod_{X_j \in Pa(X_i)} r_j$

So, formally, the AIC score is expressed as follows:

$$AIC = \log L(\Theta, G, \mathcal{D}) - dim(G) \quad (2.4)$$

Most of scores proposed in the literature satisfy two criteria: *decomposability* and *Markov equivalence*:

- A score  $S$  is said to be decomposable if it can be described in terms (generally a sum) of local scores  $s$  i.e. depending only on a node and all its parents.
- A score is said to be Markov equivalent if it assigns the same value to two equivalent graphs.

**Search methods:** Since an exhaustive search cannot be performed due to the exponential growth of the DAGs space w.r.t. the number of variables in the studied domain, score-based approaches use heuristics to traverse the DAGs space. These search methods could be classified into two families: the first family gathers methods which reduce the search space by traversing only a particular space (trees), for instance, we cite maximum weight spanning tree (MWST) (Chow and Liu, 1968), or traversing the space in a specific order (e.g. topological order) as done by K2 algorithm (Cooper and Herskovits, 1992). The second family of score-based approaches traverse all the space of solutions: DAGs or CPDAGs space and performs a greedy search.

In the following, we describe the search algorithms that will be used in the remaining. MWST algorithm (Chow and Liu, 1968) associates a weight i.e. a score to each pair of variables  $X_i, X_j \in$  and finds a subset of edges where the total of their weights is maximized. The obtained structure is a sub-graph in the form of an undirected tree. To transform it to a directed graph, one possible solution is using Depth-first search.

K2 algorithm requires a predefined topological order to perform the search. It consists in finding the best set of parents for a node (variables that come before it in the topological order) in order to maximize the score of sub-networks and combine them. At the beginning, we compute the score for the parentless variable  $X_i$ . Then, in turn, each of the parent candidates  $X_j$  is temporarily added and the score is recomputed. The parent candidate that yields the highest value of the scores is selected permanently added.

Greedy search algorithm is an iterative method which given an initial DAG (an empty DAG, randomly generated network or the tree obtained by MWST algorithm), generates all neighbor structures obtained after performing one of elementary operation, i.e., adding, deleting or reversing an edge. Then, it computes obtained neighbors structures scores and picks the operation that leads to the structure having the highest score. This process is repeated until the already obtained structure has a higher score than DAGs in the list of neighbors. This algorithm will be used in chapter 5.

## Hybrid methods

Hybrid methods combine advantages of both previous approaches and consists on a local search uses independence test in order to provide a neighborhood containing all interesting conditional independence relations and a global optimization is performed in order to search in the space of candidate graphs satisfying observed conditional dependence relations. Tsamardinou et al. (2006) show that their hybrid method Max-Min Hill-Climbing, which constrains a scoring search outperforms classical approaches. These methods are able to scale to distributions with more than thousands of variables.

## 2.4 Evidential networks

Several adaptations of Bayesian networks were proposed in the literature in different frameworks. Within these adaptations we can mention evidential networks proposed in (Xu and Smets, 1994; Smets, 1993; Ben Yaghlane and Mellouli, 1999) and based on belief function theory (cf. Section 1.7.2). In what follows, we present two common definitions of these networks, existing information propagation algorithms and their application in real world. Finally, we discuss their learning from data.

### 2.4.1 Definition

As Bayesian networks, evidential networks have two components:

- a *graphical or qualitative component* composed of a DAG.
- a numerical component which replaces probability distributions by conditional belief functions. In the literature, there are several different ways to define evidential networks. The most common definitions have been proposed by Xu and Smets (1994) and Ben Yaghlane and Mellouli (1999). In the first definition Xu and Smets (1994), conditional belief functions are not specified per variable like the case of conditional probabilities. In fact, when a variable has more than one parent, a conditional belief function has to be defined for this variable given each one of its parent separately. It is evident that this representation is restricted to graphs with only binary relations among variables. The second definition (Ben Yaghlane and Mellouli, 1999) were proposed to address this limitation. In fact, conditional belief functions can be specified in two different manners:
  - Per edge: Each edge between a variable and one of its parent is weighted by a conditional belief function
  - Per child: Each child node is associated with a conditional belief function given all its parents

## 2.4.2 Propagation, applications and software solutions

The two different definitions of evidential networks lead naturally to families of information propagation algorithms. The algorithm proposed by [Xu and Smets \(1994\)](#) for belief propagation is based on the first definition of evidential networks and adapts the message-passing algorithms initially proposed to Bayesian networks (cf. Section 2.3.2) to singly connected evidential networks. For multiply connected networks, it deletes all loops in the DAGs to transform them into singly-connected networks and perform message passing. The propagation algorithms in ([Ben Yaghlane and Mellouli, 1999, 2008](#); [Laâmari and Ben Yaghlane, 2014](#)) are based the second definition of evidential networks and transform the latter into binary joint trees ([Shenoy, 1997](#)) to perform belief propagation.

Evidential networks have been applied in various domains such as: Decision analysis ([Xu, 1997](#)), information fusion and threat assessment ([Benavoli et al., 2009](#)) and reliability analysis ([Simon and Weber, 2009](#)).

## 2.4.3 Learning from data

Few works ([Ben Hariz and Ben Yaghlane, 2014, 2015](#)) have been proposed to learn evidential networks from evidential databases ([Bachtobji et al., 2008](#)) i.e. databases containing certain or/and uncertain data modeled using the belief functions framework.

[Ben Hariz and Ben Yaghlane \(2014\)](#) proposes an evidential-likelihood-based parameter learning approach. This likelihood function is an evidential extension of the probabilistic one expressed by Equation 2.2. In ([Ben Hariz and Ben Yaghlane, 2015](#)), authors propose a constraint based approach for learning evidential networks structure from evidential by extend the probabilistic  $\chi^2$  to learn evidential networks structure. This approach generalizes constraint based methods classically used to learn Bayesian networks structure.

## 2.5 Credal networks

Credal networks ([Cozman, 2000](#)) are based on imprecise probability theory (cf. Section 1.7.3). In what follows, we present these networks by analogy to Bayesian ones. Then, we discuss information propagation in such models, their application in real world and available software solutions. Finally, we discuss their learning from data.

### 2.5.1 Definition

A credal network ([Cozman, 2000](#)) can be viewed as a collection of Bayesian networks over a fixed set of variables sharing the same DAG. More formally, a credal network consists of a DAG where variables uncertainty is generally encoded by local separately specified credal sets i.e. closed and convex sets of probability distributions (cf. Section 1.7.3).

Given a Credal network, the global joint probability distribution over the set  $V$  is the strong extension of the network i.e. the convex hull ( $CH$ ) of the set containing all joint distributions that factorizes the probabilistic chain rule in Equation 2.1 and can be expressed as follows:

$$K(X_1, \dots, X_n) = CH(p(X_1, \dots, X_n)) \quad (2.5)$$

where  $p(X_1, \dots, X_n)$  is computed using the probabilistic chain rule defined in Equation 2.1. Let  $\mathcal{P}$  be the set containing all joint distributions i.e. the strong extension of a credal network. One can compute an interval-based joint probability distribution as follows:

$$\underline{P}(X_1, \dots, X_n) = \min_{p \in \mathcal{P}} p(X_1, \dots, X_n) \quad (2.6)$$

$$\bar{P}(X_1, \dots, X_n) = \max_{p \in \mathcal{P}} p(X_1, \dots, X_n) \quad (2.7)$$

## 2.5.2 Propagation, applications and software solutions

Several exact and approximate algorithms have been proposed to perform information propagation in credal networks. These algorithms could be classified into two families: exact inference algorithms examine potential vertices of the strong extension and compute lower/upper values while approximate inference algorithms can produce either outer approximations (Cozman, 2000; da Rocha and Cozman, 2002, 2003) i.e. intervals that enclose the correct probability interval between lower and upper probabilities or inner approximations (Cano and Moral, 2002; da Rocha et al., 2002) i.e. intervals that are enclosed by the correct probability interval.

Credal networks have been applied in various complex domains such as computer vision problems (Corani et al., 2010; De Campos et al., 2009), military planning (Antonucci et al., 2009) and natural hazards identification (Antonucci et al., 2004). However, few publicly available software solutions have been proposed to manipulate them, we cite, in particular, *SAMIAM*<sup>3</sup>, *JavaBayes* (Cozman, 2001).

## 2.5.3 Learning from data

By analogy to Bayesian networks, learning credal networks from data is performed using maximum likelihood based algorithms. Most of researches attempts made to learn credal networks from data concern their parameters learning. In fact, we distinguish two family of methods: the first one estimates credal networks parameters based on Dirichlet model i.e. using imprecise Dirichlet model (IDM) (Cano et al., 2007) or imprecise sample size Dirichlet model (ISSDM) (Masegosa and Moral, 2014a). The second family considers by analogy to Bayesian networks, a likelihood-based learning approach which takes all the models quantification whose likelihood exceeds a given threshold (Antonucci et al., 2012).

Concerning structure learning of credal networks, existing works have adapted methods initially proposed in the context of Bayesian networks by replacing Bayesian scores with imprecise ones based on IDM, ISSDM and imprecise likelihood (Zaffalon and Hutter, 2005; Masegosa and Moral, 2014b; Cozman, 2014).

## 2.6 Possibilistic networks

As evidential networks, possibilistic networks were proposed as another counterpart of Bayesian networks in the possibilistic framework. Such networks were first proposed (Fonck, 1992). In this section, we present two common definitions of these networks, existing information propagation algorithms and their application in real world.

### 2.6.1 Definition

As Bayesian networks, possibilistic networks have two components:

- a *graphical* or *qualitative component* composed of a DAG.
- a *numerical* or *quantitative 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 (cf. Section 1.2.5) lead naturally to two different ways to define possibilistic networks (Borgelt et al., 2009; Fonck, 1992): quantitative also called *product-based* possibilistic networks based on the *product-based* conditioning expressed by Equation 1.7. 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

3. <http://reasoning.cs.ucla.edu/samiam>

$X_1$	$X_2$	$X_3$	$X_4$	$\pi_\times$	$\pi_m$	$X_1$	$X_2$	$X_3$	$X_4$	$\pi_\times$	$\pi_m$
$x_{11}$	$x_{21}$	$x_{31}$	$x_{41}$	0.12	0.3	$x_{12}$	$x_{21}$	$x_{31}$	$x_{41}$	0.4	0.4
$x_{11}$	$x_{21}$	$x_{31}$	$x_{42}$	0.036	0.3	$x_{12}$	$x_{21}$	$x_{31}$	$x_{42}$	0.12	0.3
$x_{11}$	$x_{21}$	$x_{32}$	$x_{41}$	0.12	0.3	$x_{12}$	$x_{21}$	$x_{32}$	$x_{41}$	1	1
$x_{11}$	$x_{21}$	$x_{32}$	$x_{42}$	0.024	0.2	$x_{12}$	$x_{21}$	$x_{32}$	$x_{42}$	0.2	0.2
$x_{11}$	$x_{22}$	$x_{31}$	$x_{41}$	0.2	0.4	$x_{12}$	$x_{22}$	$x_{31}$	$x_{41}$	0.14	0.4
$x_{11}$	$x_{22}$	$x_{31}$	$x_{42}$	0.4	0.4	$x_{12}$	$x_{22}$	$x_{31}$	$x_{42}$	0.28	0.4
$x_{11}$	$x_{22}$	$x_{32}$	$x_{41}$	0.4	0.4	$x_{12}$	$x_{22}$	$x_{32}$	$x_{41}$	0.7	0.7
$x_{11}$	$x_{22}$	$x_{32}$	$x_{42}$	0.04	0.1	$x_{12}$	$x_{22}$	$x_{32}$	$x_{42}$	0.07	0.1

Table 2.1: Joint possibility distribution of the network defined by Figure 2.2 in the numerical ( $\pi_\times$ ) and the ordinal interpretation ( $\pi_m$ )

not the case of qualitative also called *min-based* possibilistic networks based on *min-based* conditioning defined by Equation 1.8 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:

$$\pi_{\otimes}(X_1, \dots, X_n) = \otimes_{i=1..n} \pi(X_i \mid_{\otimes} Pa(X_i)) \tag{2.8}$$

where  $\otimes$  corresponds to the minimum operator (min) for qualitative possibilistic networks and to the product operator (\*) for quantitative possibilistic networks.

Borgelt and Kruse (2003) define quantitative possibilistic networks differently with a numerical component whose conditional possibility distributions are not necessarily normalized. Moreover, they are based on non-interactivity relation which is applicable only in the ordinal interpretation. In this context, it is also possible to apply the chaining rule (Equation 2.8) to non-normalized possibility distributions.

**Example 2.6.1.** The DAG and conditional distributions in Figure 2.2 represent a possibilistic network composed of four binary variables  $X_1, X_2, X_3$  and  $X_4$  defined respectively on  $D_1 = \{x_{11}, x_{12}\}$ ,  $D_2 = \{x_{21}, x_{22}\}$ ,  $D_3 = \{x_{31}, x_{32}\}$  and  $D_4 = \{x_{41}, x_{42}\}$ . Table 2.1 gives its joint possibility distribution in the numerical ( $\pi_\times$ ) and the ordinal interpretation ( $\pi_m$ ).

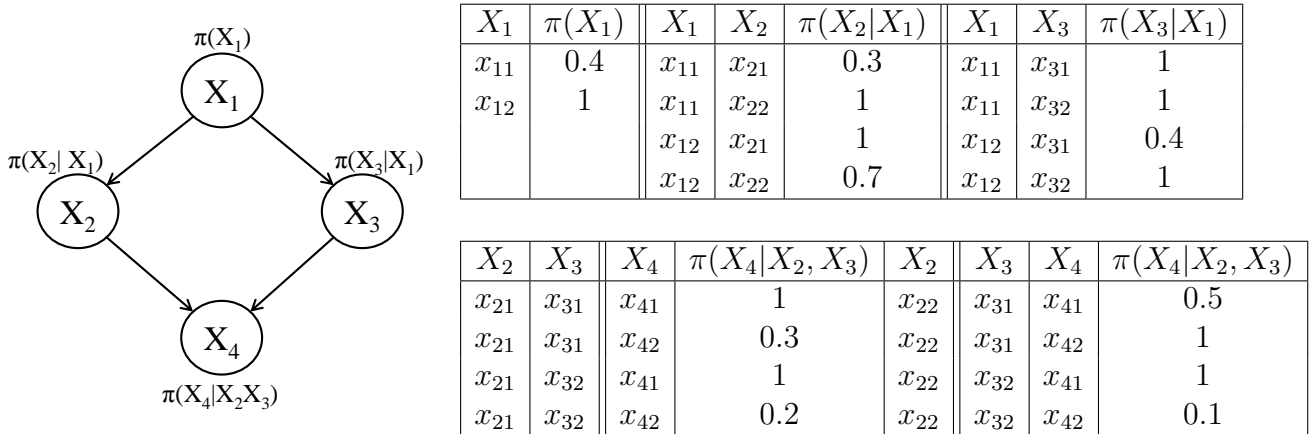


Figure 2.2: An example of possibilistic networks

## 2.6.2 Inference, applications and software solutions

### Inference

Most of inference methods proposed in literature (Fonck, 1994; Gebhardt and Kruse, 1997) represent direct adaptations of methods initially proposed to Bayesian networks. These works show that *product-based* possibilistic networks inference algorithms are very similar to probabilistic ones using the same operator: product. This is not the case of min-based possibilistic networks. In fact, the min operator has particular properties such as the idempotency and specific algorithms have thereby been proposed. We can, in particular, cite the approximative algorithm *anytime* proposed in (Ben Amor et al., 2003) to avoid the transformation of an initial DAG into a junction tree in the case of multiply connected DAGs.

Benferhat and Smaoui (2007) have proposed an inference algorithm in hybrid possibilistic networks where uncertainty encoded by each node and its parents are represented by possibilistic logic. Inference algorithms proposed in (Ayachi et al., 2010) based on knowledge compilation techniques following Darwiche et al.'s work in the probabilistic case (cf. Section 2.3.2).

All of these methods have shown that the use of possibility theory in its ordinal interpretation provides a significant gain in terms of treatments necessary for inference. Recently, another attempt has been made concerning inference in naive structures handling soft evidence (Benferhat and Tabia, 2012) described by possibility distributions relative to observed variables.

### Applications

Possibilistic networks have been proposed to solve real problem in various domains as data fusion (Beckmann et al., 1994), automotive industry (Kruse and Borgelt, 1995), intelligent tutoring systems (Adina, 2006), social specialization in metropolized spaces (Dubois et al., 2015) and information retrieval (Chebil et al., 2015; Boughanem et al., 2009). In what follows, we detail two examples of possibilistic networks applications.

First, we can cite the work presented in (Dubois et al., 2015) where authors explore uncertain knowledge elicitation describing the field of social specialization in metropolized spaces. To this end, they need an uncertainty framework where factors affecting the studied domain can be captured in a qualitative way, more precisely, in a hierarchical form without quantifying them. Moreover, the evaluation of the impact of certain variables on the others is not precise and is modeled by fuzzy membership functions that generate possibility distributions. Therefore, the use of Bayesian networks in this case is not very suitable.

In what follows, we present another example in information retrieval where the concept of total ignorance is unavoidable and better expressed with possibilistic networks. Boughanem et al. (2009) try to select from a collection of documents those likely to be relevant. Their model distinguishes, for example, rejection of a document as irrelevant and selection by the use possibility and necessity measures. The possibility of relevance is used to eliminate irrelevant documents while the necessity of relevance is used to select the most interesting ones. In addition, the weighting of a single word with a unique probability does not capture the dual concept of specialty and non-specialty, easily modeled with the possibility theory.

### Software solutions

Unlike Bayesian networks, only one publicly available software manipulates possibilistic networks: INeS (Induction of Networks Structure) (Borgelt et al., 2009) that implements several methods detailed in Section 2.7.2 dedicated to structure learning. Note that there is no solution for parameters learning.

## 2.7 Learning possibilistic networks from data

In this section, we give an overview of quantitative possibilistic networks learning algorithms. To the best of our knowledge, parameters learning problem has not been studied yet. In fact, existing methods

(Borgelt et al., 2009; Sangüesa et al., 1998) are dedicated to the structure learning and ignore parameters learning problem. In fact, they compute possibility distributions either using the non-normalized estimation described in Section 1.4 as done in (Borgelt et al., 2009) or using probability possibility transformations as in (Sangüesa et al., 1998). So, the output of the proposed methods is a DAG which is not characterized by any numerical data even if the proposed learning process is ensured in the possibilistic framework. Semantically, the resultant structure is closest to a qualification via possibilistic conditional distributions which may represent the qualitative component of several graphical models. That said, the semantic of generated structures fits better with possibilistic networks and more precisely quantitative ones. A recent work (Haddad et al., 2016) has been proposed to learn possibilistic network parameters using the imprecise probability possibility transformation proposed by Destercke et al. (cf. Section 1.7.3). In fact, authors learn imprecise probability distributions from data and approximate them by possibility distributions.

By analogy to Bayesian networks, structure learning methods could be categorized into three families: constraint-based, score-based and hybrid methods.

### 2.7.1 Constraint-based methods

In the possibilistic case, as far as we know, only one attempt has been made to measure conditional independence in order to learn possibilistic networks (Sangüesa et al., 1998). In this work, authors have proposed an independence measure, denoted by  $Dep(X_i, X_j, \alpha)$ , and is expressed as follows:

$$Dep(X_i, X_j, \alpha) = 1 - \sum_{x_{jl} \in D_j} \pi(x_{jl}) \sum_{x_{ik} \in \alpha\text{-set}} |\pi(x_{ik}) - \pi(x_{ik}|x_{jl})| \quad (2.9)$$

where  $\alpha\text{-set} = \{x_{ik} \in D_i \text{ s.t. } |\pi(x_{ik}) - \pi'(x_{ik})| \geq \alpha\}$  and  $\alpha \in [0,1]$ .

### 2.7.2 Score-based methods

Borgelt et al. (2009) have proposed two methods handling imprecise data, possibilistic versions of two learning methods initially proposed to Bayesian networks: K2 and maximum weight spanning tree (cf. Section 2.3.3). These adaptations propose several possibilistic scores: first works propose global scores and following works are based on local ones.

Borgelt and Gebhardt (1997) have proposed a global score named weighted sum of possibility degrees. Given an imprecise dataset  $\mathcal{D}$  and a DAG  $G$  learned from  $\mathcal{D}$ , weighted sum of possibility degrees, denoted by  $Q(G, \mathcal{D})$ , is expressed as follows:

$$Q(G, \mathcal{D}) = \sum_{\{x_{1k}, \dots, x_{nl}\} \in \mathcal{D}} N_{x_{1k}, \dots, x_{nl}} \hat{f}^{nn}(x_{1k}, \dots, x_{nl}) \quad (2.10)$$

$\hat{f}^{nn}(x_{1k}, \dots, x_{nl})$  is computed locally combining  $\hat{f}^{nn}(X_i | Pa(X_i))$  computed by Equation 1.14 and  $N_{x_{1k}, \dots, x_{nl}}$  is the number of occurrences of the tuple  $(x_{1k}, \dots, x_{nl})$ .

Then, Borgelt et al. have proposed other local scores which are:

- Specificity gain (Borgelt and Kruse, 2003), denoted by  $S_{gain}$ , is expressed by:

$$S_{gain}(X_i, X_j) = nsp(\pi(X_i)) + nsp(\pi(X_j)) - nsp(\pi(X_i, X_j)) \quad (2.11)$$

where possibility distributions are computed using Equation 1.14. This measure is the ancestor of several scores (Borgelt and Kruse, 2003) i.e specificity gain ratio, symmetric specificity gain, etc.

– Specificity gain ratio denoted by  $S_{gr}$  and expressed by:

$$S_{gr}(X_i, X_j) = \frac{S_{gain}(X_i, X_j)}{nsp(\pi(X_j))} = \frac{nsp(\pi(X_i)) + nsp(\pi(X_j)) - nsp(\pi(X_i, X_j))}{nsp(\pi(X_j))} \quad (2.12)$$

- Symmetric specificity gain ratios denoted by  $S_{gr}^{(1)}$  et  $S_{gr}^{(2)}$  and expressed by:

$$S_{gr}^{(1)}(X_i, X_j) = \frac{S_{gain}(X_i, X_j)}{nsp(\pi(X_i, X_j))} \quad (2.13)$$

$$S_{gr}^{(2)}(X_i, X_j) = \frac{S_{gain}(X_i, X_j)}{nsp(\pi_i(X_i)) + nsp(\pi(X_j))} \quad (2.14)$$

- Conditional specificity gain denoted by  $S_{cgain}$  and expressed by:

$$S_{cgain}(X_i, X_j) = \sum_{x_j \in D_j} \int_0^{\frac{N(x_j)}{N}} \frac{\left(\frac{N(x_j)}{N}\right)_\alpha}{\sum_{x_j \in D_j} \left(\frac{N(x_j)}{N}\right)_\alpha} \log_2 \frac{\sum_{x_i \in D_i} \left(\frac{N(x_i)}{N}\right)_\alpha}{\sum_{x_j \in D_j} \left(\frac{N(x_i|x_j)}{N}\right)_\alpha} d\alpha \quad (2.15)$$

- Possibilistic mutual information (Borgelt and Kruse, 2003), denoted by  $d_{mi}$ , is expressed by:

$$d_{mi}(X_i, X_j) = - \sum_{\substack{x_{ik} \in D_i \\ x_{jl} \in D_j}} \frac{N_{ik,jl}}{N} \cdot \log_2 \frac{N_{ik,jl}}{\min(N_{ik}, N_{jl})} \quad (2.16)$$

- Possibilistic  $\chi^2$  measure (Borgelt and Kruse, 2003), denoted by  $d_{\chi^2}$ , is expressed by:

$$d_{\chi^2}(X_i, X_j) = \sum_{\substack{x_{ik} \in D_i \\ x_{jl} \in D_j}} \frac{(\min(N_{ik}, N_{jl}) - N_{ik,jl})^2}{\min(N_{ik}, N_{jl})} \quad (2.17)$$

A generalization of these scores to more than two attributes is made as follows: given a variable  $X_i$ , all its parents could be combined into one pseudo-variable. That is, we measure dependence between  $X_i$  and an artificial attribute representing the combination of all parents, i.e. the Cartesian product of their domains.

### 2.7.3 Hybrid methods

These methods combine advantages of the two previous families. In fact, hybrid methods use information captured from conditional independence tests to guide search in DAGs space optimizing a score. Sangüesa et al. (1998) have proposed two hybrid learning methods from precise data: the first one learns trees and the second one learns DAGs. The two methods use the independence measure defined by Equation 2.9. This method learns undirected graphs detecting relations between each node and its parents and children. Then, it combines obtained sub-graphs and orient edges using DAG non-specificity expressed by:

$$nsp(G) = \sum_{X_i \in V} nsp(\pi(X_i | Pa(X_i))) \quad (2.18)$$

where  $nsp(\pi(X_i | Pa(X_i))) = nsp(\pi(X_i, Pa(X_i))) - nsp(\pi(Pa(X_i)))$  and  $nsp(\pi(X_i | Pa(X_i))) = nsp(\pi(X_i))$  si  $Pa(X_i) = \emptyset$ .

This method uses three probability possibility transformations to learn parameters, namely, Klir transformation, optimal transformation and symmetric transformation (cf. Section 1.7.1).

### 2.7.4 Discussion

Only few attempts have been made to learn possibilistic networks structure from data. Moreover, existing works (Borgelt et al., 2009; Sangüesa et al., 1998) have been proposed before advances made concerning possibilistic networks as models of independence (Ben Amor and Benferhat, 2005) ignoring also parameters learning problem.



In what follows, we give a global vision for possibilistic networks structure learning methods detailing limitations of each one. Possibilistic networks structures learned using K2 and maximum weight spanning tree are evaluated using weighted sum of possibility degrees expressed by Equation 2.10. We note a mismatch between global and local scores definitions. In fact, weighted sum of possibility degrees is not decomposable on any local score. This score is close to log-likelihood used to learn Bayesian networks, expressed by  $\sum_{\{x_{1k}, \dots, x_{nl}\} \in \mathcal{D}} N_{x_{1k}, \dots, x_{nl}} \log p(x_{1k}, \dots, x_{nl})$ . The possibilistic adaptation (Borgelt and Gebhardt, 1997) sees disappearing log by proposing weighted sum of possibility degrees without justification.

Concerning the hybrid method (Section 2.7.3), the main problem residing in its conditional independence measure is that it is based on a similarity measure for which we find several contradictory formulations in several works proposed by the same authors (Sangüesa et al., 1998; Sangüesa et al., 1998b). The second problem is the lack of an automatic computation of threshold to decide between the two hypothesis (dependent or independent variables) such as the case of statistical tests. Moreover, this method could fail to return a DAG since it does not take into account acyclicity property during the learning process. To estimate possibility distributions, this hybrid method uses three transformations (Klir and Parviz, 1992; Dubois et al., 1993, 2004) that could not be applied to possibilistic networks (see Section 4.3.1).

Measure	Equation	Data	Function to be optimized
Dep	2.9	precise	distance between the joint distributions and independent distributions
nsp	2.18	precise	specificity
Q	2.10	imprecise	divergence w.r.t. an (unknown) initial network
S <sub>gain</sub>	2.11	imprecise	divergence between joint distribution specificity and independent distributions specificity
d <sub>mi</sub>	2.16	imprecise	difference between the joint distribution and independent distributions
d <sub>χ</sub> <sup>2</sup>	2.17	imprecise	difference between the joint distribution and independent distributions

Table 2.2: Summary table of measures properties

## 2.8 Conclusion

In this chapter, we have introduced basic definitions and concepts related to graphical representation of knowledge in uncertain frameworks. In particular we have presented Bayesian networks and their existing structure learning methods from data. Then, we have briefly introduced two graphical reasoning models namely evidential and credal networks by presenting some propagation algorithms, their application in real world and software solutions proposed to manipulate them.

In the second part of this chapter, we mainly focus on possibilistic networks and in particular, their learning from data. Concerning parameters learning, this problem has not been studied yet and remains an open research area. Moreover, existing structure learning algorithms are direct adaptations of Bayesian networks methods and we have shown that replacing probability distributions by possibility distributions is not satisfactory. Moreover, they have been proposed before advances made on possibilistic framework and use scoring functions whose values are difficult to interpret. These limitations make them limited and theoretically unsound. Thereby, whatever the studied task, learning possibilistic networks is a problem that is not well studied.

Next chapters propose a new possibilistic likelihood function which will represent the key concept of a global possibilistic networks learning algorithm. The proposed likelihood function will be explored to learn possibilistic networks parameters and to define a new score to learn their structure. In the next chapter, we propose a new validation strategy which will represent a clear experimental framework allowing the evaluation of proposed learning algorithms.





## Contributions



# Benchmarking possibilistic networks learning algorithms and evaluation measures

## Contents

<b>3.1</b>	<b>Introduction</b>	<b>37</b>
<b>3.2</b>	<b>Possibilistic networks benchmark generation</b>	<b>38</b>
3.2.1	Possibilistic networks generation	38
3.2.2	Possibilistic networks sampling	38
<b>3.3</b>	<b>Learning evaluation measures</b>	<b>41</b>
3.3.1	Graphical measures	42
3.3.2	Numerical measures	42
<b>3.4</b>	<b>Experimental study</b>	<b>43</b>
3.4.1	Possibilistic networks sampling process evaluation	43
3.4.2	Manhattan distance and its approximation evaluation	44
<b>3.5</b>	<b>Conclusion</b>	<b>45</b>

## 3.1 Introduction

Probabilistic graphical models learning methods, in particular Bayesian networks ones, are tested using randomly generated networks (synthetic) or networks that have been used in real systems, so that the structure of the network is known and can serve as a rigorous gold standard e.g. Asia ([Lauritzen and Spiegelhalter, 1988](#)) and Insurance ([Russell et al., 1995](#)) networks. Assessing the quality of learning algorithms consists in comparing an initial graphical model with the learned one. In the probabilistic case, we can always rely on the following approach which consists in selecting an arbitrary Bayesian network either randomly generated or constructed by an expert and generating a dataset using Forward Sampling ([Henrion, 1986b](#)). Then, we try to recover the initial network using a learning algorithm and we compare the initial network with the learned one.

As far as we know, such evaluation process has not been transposed yet in the possibilistic case. In fact, existing possibilistic networks learning algorithms ([Borgelt et al., 2009](#); [Sangüesa et al., 1998](#)) suffer from the lack of an accurate and standard validation procedure and each method proposes its evaluation measure whose values are difficult to interpret.

This chapter rigorously addresses this problem by proposing a new evaluation strategy to product-based possibilistic networks learning algorithms. First, we will propose a possibilistic networks sampling method and two extensions of the latter in which we control the imprecision and the consistency degrees in the generated datasets. Then, we propose two new possibilistic evaluation measures to assess learning algorithms quality: the first one presents an approximation of Manhattan distance in the context of possibilistic networks i.e. computed between joint possibility distributions of two networks and the second is based on information affinity (cf. Section 1.6.4) and compares conditional possibility distributions of two networks sharing the same structure.

This chapter is organized as follows: Section 3.2 proposes a possibilistic networks benchmark generation methods. Section 3.3 is dedicated to the proposition of new possibilistic learning evaluation measures. The detailed experimental study in Section 3.4 shows the efficiency of the sampling method and the evaluation measure Manhattan distance approximation.

## 3.2 Possibilistic networks benchmark generation

In the possibilistic case, there are currently no publicly available possibilistic networks used in real systems that could be used as gold standard. Thereby, this section proposes a synthetic approach to generate randomly possibilistic networks and to construct imprecise datasets from generated networks.

### 3.2.1 Possibilistic networks generation

Generating a possibilistic network consists in generating its qualitative and quantitative components. Two solutions can be considered: generating randomly the network or transforming a gold Bayesian network into a possibilistic one. In fact, concerning the graphical component, we could use any method proposed in the context of Bayesian networks such as (Xiang and Miller, 1999). For the numerical component, we propose to uniformly generate random values from  $[0,1]$  for each distribution satisfying normalization property, i.e. at least one of states degrees is equal to 1.

The second solution consists on transforming an existing (gold) Bayesian network into a possibilistic one. More precisely, the obtained network shares the same structure with the Bayesian one and its conditional possibility distributions are computed using one of probability possibility transformations discussed in Section 1.7.1.

### 3.2.2 Possibilistic networks sampling

Sampling a possibilistic network consists in generating a dataset representative of its joint distribution. To the best of our knowledge, this problem has not been studied yet. We propose to generalize the variable sampling method proposed by Guyonnet et al. (cf. section 1.5) 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. In what follows, we generalize the sampling variable method to the case of conditioned variables.

#### Sampling conditioned variables

Instantiating a parentless variable corresponds to computing its  $\alpha$ -cut. Instantiating a conditioned variable corresponds to computing also its  $\alpha$ -cut given its sampled parents values. This could not be directly applied to conditional possibility distribution which is composed of more than one distribution. So, to instantiate a conditioned variable  $X_i$ , we compute  $\alpha$ -cut from  $\Pi(X_i | Pa(X_i) \in A)$ , computed as follows:

$$\Pi(x_{ik} | Pa(X_i) \in A) = \max_{a_i \in A} \pi(x_{ik} | a_i) \pi(a_i) \quad (3.1)$$

Equation 3.1 could be obtained applying the following equalities:

$$\pi(x_{ik} | A) = \frac{\pi(A | x_{ik}) \pi(x_{ik})}{\Pi(A)} = \max_{a_i \in A} \pi(a_i | x_{ik}) \pi(x_{ik}), \quad (\Pi(A) = 1, A \text{ is a cut and has at least one value})$$

whose possibility degree which is equal to 1).

Using  $\pi(a_i|x_{ik})\pi(x_{ik}) = \pi(x_{ik}|a_i)\pi(a_i)$ , we obtain Equation 3.1.

### Sampling process

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.

The proposed sampling process is formally described by Algorithm 1.

---

#### Algorithm 1 Sampling process

---

Input: Possibilistic network

Output: Observation

**begin**

  % Process nodes in a topological order

**foreach**  $X_i \in V$  **do**

**if**  $X_i$  is parentless **then**

      | observation( $X_i$ )= $\alpha$ -cut( $X_i$ )

**else**

      | Compute  $\Pi(X_i|\text{observation}((Pa(X_i))))$  using Equation 3.1

      | observation( $X_i$ )= $\alpha$ -cut( $X_i$ ) from  $\Pi(X_i|\text{observation}((Pa(X_i))))$

**end**

**end**

  Return observation

**end**

---

**Example 3.2.1.** Let us consider the possibilistic network in Figure 2.2. The topological order is  $X_1, X_2, X_3, X_4$ . Applying the described sampling process we obtain:

1.  $X_1: \alpha = 0.3: \alpha\text{-cut}(X_1) = \{x_{11}, x_{12}\}$ .

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)  $\alpha\text{-cut}(X_2) = \{x_{21}, x_{23}\}.$

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)  $\alpha\text{-cut}(X_3) = \{x_{31}\}.$

4.  $X_4: \alpha = 0.2:$

(a)  $\pi'(x_{41}) = \max(1 * 1 * 1, 1 * 1 * 0.6) = 1, \pi'(x_{42}) = \max(1 * 1 * 0.3, 1 * 1 * 1) = 1,$

$\pi'(x_{43}) = \max(1 * 1 * 0.4, 1 * 1 * 0.5) = 0.5.$

(b)  $\alpha\text{-cut}(X_4) = \{x_{41}, x_{42}, x_{43}\}.$

The obtained observation is then  $(\{x_{11}, x_{12}\}, \{x_{21}, x_{23}\}, \{x_{31}\}, \{x_{41}, x_{42}, x_{43}\})$ .

The sampling process generates a particular case of imprecise datasets i.e. obtained data relative to a variable  $X_i$  are conditionally consonant with respect to the sampled values of its parents. This is due the fact that the sampling process is based on the  $\alpha$ -cut notion which returns generally most possible values as observed ones.

In what follows, we propose to parametrize this sampling process in order to generate more generic imprecise data by controlling the consistency degree and the imprecision degree in generated datasets.

## Imprecision control

The aim of controlling the imprecision degree in generated datasets is to create different forms of imprecision around the most possible value i.e. varying the values in the dataset but we conserve the most possible combination of  $\Omega$ . Given an imprecision degree  $\theta_{imp}$  and a variable  $X_i$  such that the  $\alpha$ -cut( $X_i$ ) presents values returned by the sampling process, we generate all subsets of  $\alpha$ -cut including the most possible value and we assign a probability  $\theta_{imp}^{card(S_{X_i})-1} * (1 - \theta_{imp})^{card(\alpha\text{-cut}(X_i)) - card(S_{X_i})}$  to each subset  $S_{X_i} \subseteq \alpha\text{-cut}(X_i)$  i.e. we assign  $\theta_{imp}$  for each observed state different to the most possible one in  $S_{X_i}$  and  $(1 - \theta_{imp})$  for each non observed state. Note that if  $\theta_{impr} = 0$ , the algorithm returns necessarily the most possible value contrarily to the case of  $\theta_{impr} = 1$ , the algorithm returns necessarily  $\alpha\text{-cut}(X_i)$ . Finally, we sample this probability distribution and we replace  $\alpha\text{-cut}(X_i)$  by the sampled subset in the dataset.

The proposed sampling process is formally described by Algorithm 2.

---

### Algorithm 2 Sampling process (imprecision control)

---

Input: Possibilistic network

Output: Observation

**begin**

  % Process nodes in a topological order **foreach**  $X_i \in V$  **do**

**if**  $X_i$  is parentless **then**

      | observation( $X_i$ )= $\alpha$ -cut( $X_i$ )

**else**

      | Compute  $\Pi(X_i|\text{observation}((Pa(X_i))))$  using Equation 3.1

      | observation( $X_i$ )= $\alpha$ -cut( $X_i$ ) from  $\Pi(X_i|\text{observation}((Pa(X_i))))$

**end**

**end**

**foreach**  $S_{X_i} \subseteq \text{cut}(X_i)$  **do**

    |  $p(S_{X_i}) = \theta_{imp}^{card(S_{X_i})-1} * (1 - \theta_{imp})^{card(\alpha\text{-cut}(X_i)) - card(S_{X_i})}$

**end**

  observation( $X_i$ )= $\text{sample}(p)$

  Return observation

**end**

---

**Example 3.2.2.** Let  $X_1$  be a quaternary variable  $X_1$  such that  $D_1 = \{x_{11}, x_{12}, x_{13}, x_{14}\}$  and the  $\alpha$ -cut returned by the sampling process is equal to  $\{x_{11}, x_{12}, x_{14}\}$  with  $x_{11}$  the most possible value. We generate all subsets (4 subsets) of  $\alpha$ -cut including  $x_{11}$  i.e.,  $\{x_{11}\}$ ,  $\{x_{11}, x_{12}\}$ ,  $\{x_{11}, x_{14}\}$  and the original set  $\{x_{11}, x_{12}, x_{14}\}$ . Then we assign for each subset a probability based on  $\theta_{imp}$ :  $P(\{x_{11}\}) = (1 - \theta_{imp})^2$ ,  $P(\{x_{11}, x_{12}\}) = \theta_{imp} * (1 - \theta_{imp})$ ,  $P(\{x_{11}, x_{14}\}) = \theta_{imp} * (1 - \theta_{imp})$  and the original set  $\{x_{11}, x_{12}, x_{14}\} = \theta_{imp}^2$ . Finally, we sample this probability distribution and we replace the original set by the sampled subset in the dataset.

## Consistency control

The objective of controlling consistency is to give chance to combinations that do not include the most possible value to be generated in the dataset. Given a consistency degree  $\theta_{cons}$  and a variable  $X_i$  such that the  $\alpha$ -cut( $X_i$ ) presents values returned by the sampling process, we generate all subsets (nb-subsets denotes the number of subsets) of  $\alpha$ -cut and we assign a probability equal to  $\theta_{cons}$  to  $\alpha$ -cut( $X_i$ ) and a probability equal to  $\frac{1-\theta_{cons}}{nb\text{-subsets}-1}$  to remaining subsets i.e. to each  $S_{X_i}$ . Note that if  $\theta_{cons} = 0$ , the algorithms returns necessarily a subset  $S_{X_i} \subset \alpha\text{-cut}(X_i)$  and cannot in any case return  $\alpha\text{-cut}(X_i)$  contrarily to the case of  $\theta_{cons} = 1$ , the algorithm returns necessarily  $\alpha\text{-cut}(X_i)$ . Finally, we sample this probability distribution and



we replace the  $\alpha$ -cut( $X_i$ ) by the sampled subset in the dataset. The proposed sampling process is formally described by Algorithm 3.

---

**Algorithm 3** Sampling process (consistency control)
 

---

```

Input: Possibilistic network
Output: Observation
begin
  % Process nodes in a topological order foreach  $X_i \in V$  do
    if  $X_i$  is parentless then
      | observation( $X_i$ )= $\alpha$ -cut( $X_i$ )
    else
      | Compute  $\Pi(X_i|\text{observation}((Pa(X_i))))$  using Equation 3.1
      | observation( $X_i$ )= $\alpha$ -cut( $X_i$ ) from  $\Pi(X_i|\text{observation}((Pa(X_i))))$ 
    end
     $p(\alpha\text{-cut}(X_i))=\theta_{cons}$ 
    foreach  $S_{X_i} \subset \text{cut}(X_i)$  do
      |  $p(S_{X_i}) = \frac{1-\theta_{cons}}{nb\text{-subsets}-1}$ 
    end
  end
  observation( $X_i$ )= $\text{sample}(p)$ 
  Return observation
end

```

---

**Example 3.2.3.** Let  $X_1$  be a quaternary variable such that  $D_1 = \{x_{11}, x_{12}, x_{13}, x_{14}\}$  such that the  $\alpha$ -cut returned by the sampling process is equal to  $\{x_{11}, x_{12}, x_{14}\}$ . We generate all subsets (7 subsets) of  $\alpha$ -cut i.e.,  $\{x_{11}\}$ ,  $\{x_{12}\}$ ,  $\{x_{14}\}$ ,  $\{x_{11}, x_{12}\}$ ,  $\{x_{11}, x_{14}\}$ ,  $\{x_{12}, x_{14}\}$  and the original set  $\{x_{11}, x_{12}, x_{14}\}$  such that the latter has probability  $P(\{x_{11}, x_{12}, x_{14}\}) = \theta_{cons}$  and the probability of remaining sets is  $\frac{1-\theta_{cons}}{7-1}$ . Finally, we sample this probability distribution and we replace the original set by the sampled subset in the dataset.

### 3.3 Learning evaluation measures

An evaluation measure assesses learned possibilistic networks quality and quantifies the efficiency of the learning method graphically or numerically. Evaluation measures could be classified into two families: The first family gathers graphical measures which are used to compare the structure of the initial network and the learned one. The second family gathers numerical measures which compare the initial network and the learned one using a possibilistic dissimilarity/similarity measure between their joint possibility distribution as done by KL divergence (Kullback and Leibler, 1951) in the probabilistic case or between their conditional possibility distributions. Such a measure has been proposed to compare two possibility distributions  $\pi$  and  $\pi'$  defined in  $D_i$  s.t.  $\pi(x_{ik}) \geq \pi'(x_{ik}) \forall x_{ik} \in D_i$  (Borgelt et al., 2009). This hypothesis is restrictive for comparing two possibilistic networks.

In this section, we briefly recall graphical evaluation measures initially proposed in the Bayesian networks context that could be used in the possibilistic case. Then, we propose two new numerical evaluation measures, namely Manhattan distance approximation and mean information affinity. The first measure is global i.e. computed between joint possibility distributions of two networks and mean information affinity is local i.e. computed between conditional possibility distributions of two networks sharing the same structure.

### 3.3.1 Graphical measures

To evaluate possibilistic networks structure learning algorithms, we can use measures proposed in the context of Bayesian networks, e.g. sensitivity, specificity and editing distance. For more details, see (Fielding and Bell, 1997; Shapiro and Haralick, 1985).

- Editing distance: number of operations required to transform learned possibilistic network DAG into the initial one (add, reverse or delete an edge increases the editing distance by 1).
- Specificity: ratio of edges correctly identified as not belonging to the learned possibilistic networks DAG over the true number of edges not present in the initial possibilistic network DAG.
- Sensitivity: ratio of correctly identified edges over the total number of initial possibilistic network edges.

Note that, it is necessary to take into account Markov equivalence properties when computing these measures if the used score is Markov equivalent. 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-structures.

### 3.3.2 Numerical measures

Studying possibilistic networks parameters learning algorithms behavior requires numerical evaluation measures to compare the learned network and the initial one by quantifying the gap between their joint and conditional possibility distributions. As a direct solution, we have proposed in (Haddad et al., 2015c) to use information affinity expressed by Equation 1.24 between the initial and the learned networks joint possibility distributions. However, like the case of KL divergence (Kullback and Leibler, 1951) in the probabilistic case, information affinity involves heavy computing if the number of variables increases. This can be explained by the fact that they involve all  $\omega \in \Omega$ . Consequently, to use these measures efficiently, we can approximate them. For KL divergence such approximation exists, but, for information affinity, it has not been studied yet and is not an option. In fact, the inconsistency degree computed in information affinity is between two distributions i.e. joint distributions in our case and could not be obtained locally. Thereby, we propose an extension of information affinity, named mean information affinity and denoted by *Mean-aff*. It consists on computing similarity between two possibilistic networks by aggregating the local similarity measures i.e. computed conditional possibility distributions. Mean information affinity is expressed as follows:

$$Mean-Aff(\pi_0, \pi_l) = \frac{1}{n} \sum_{i=1}^n \frac{1}{q_i} \sum_{j=1}^{q_i} Aff(\pi_0(X_i|Pa(X_i) = x_j), \pi_l(X_i|Pa(X_i) = j)) \quad (3.2)$$

Note that this measure is restricted to the case of comparing two possibilistic networks sharing the same structure. In order to compare two networks with different structures, we propose to use Manhattan distance in Equation 1.21 to compute the gap between their joint possibility distributions. Like information affinity, the size of the joint domain  $\Omega$  makes the computing of *MD* impossible. However, we may consider restricting the set  $\Omega$  from which this measure is computed, so that the computation becomes efficient. A natural choice for such a randomly generated sample of  $\Omega$  denoted by *Sub- $\Omega$* . Equation 1.21 becomes:

$$Approx-MD(\pi_0, \pi_l) = \sum_{\omega \in \text{Sub-}\Omega} \frac{|\pi_0(\omega) - \pi_l(\omega)|}{card(\text{Sub-}\Omega)} \quad (3.3)$$

To generate the *Sub- $\Omega$* , for each  $X_i$ , we generate one value from  $D_i$  uniformly chosen. Note that all the generated elements of *Sub- $\Omega$*  are distinct.

## 3.4 Experimental study

The first set of experiments evaluates the efficiency of our sampling method described in Algorithm 1. The second set of experiments evaluates the efficiency of Manhattan distance in comparing possibilistic networks and the convergence of its approximation values.

### 3.4.1 Possibilistic networks sampling process evaluation

The objective of this experiment is to study the convergence of the joint possibility distribution computed from generated data using Equation 1.14, denoted by  $\pi_l$ , to the theoretical one, i.e. computed using Equation 2.8, denoted by  $\pi_0$  as shown by Figure 3.1. Specifically, we generate synthetic datasets using Algorithm 1 containing 100, 1000, 5000 and 10000 observations from 100 randomly generated possibilistic networks composed of  $nb$  nodes where  $nb$  is randomly generated in  $[5,10]$ . Then, we compute information affinity between  $\pi_0$  and  $\pi_l$ .

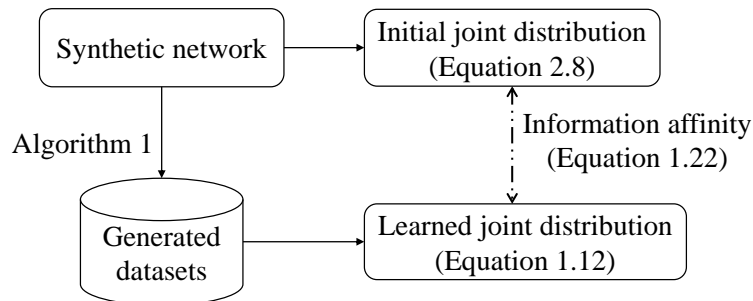


Figure 3.1: Experimental protocol of possibilistic networks sampling evaluation

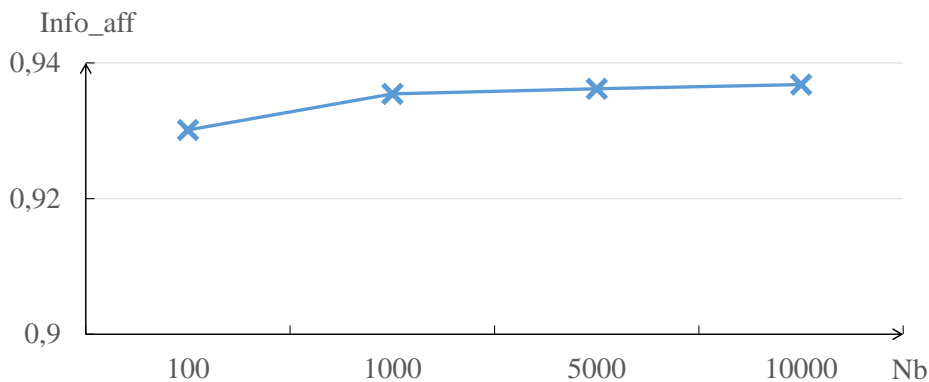


Figure 3.2: Information affinity between  $\pi_0$  and  $\pi_l$  w.r.t. the size of generated datasets (average over 100 experiments)

Figure 3.2 presents the mean of obtained values with a standard deviation around 0.04. Obtained results show 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 dataset becomes more informative and representative of the joint possibility distribution, i.e. most possible  $\omega_i$  appears more frequently, less possible appears less frequently and so on until reaching the least possible  $\omega_i$  or impossible  $\omega_i$ . Consequently, we deflate considerably the gap between the initial possibility distribution and the learned one.

### 3.4.2 Manhattan distance and its approximation evaluation

The first objective of this set of experiments is to evaluate Manhattan distance efficiency in the context of comparing two possibilistic networks. We generate randomly 25 possibilistic networks composed of  $nb$  variables generated from  $\{5, 10, 20, 30\}$  variables varying the maximum number of parents between  $\{2, 4, 8\}$  and the maximum number of variables domains cardinality between  $\{2, 5, 10\}$ . Then, for each network, we generate random possibilistic networks having the *same structure* as the initial one but in which we increase the dissimilarity between the generated network and the initial one. The dissimilarity corresponds to  $\Delta = \frac{\text{Number of dissimilar parameters}}{\text{Total number of parameters}}$ . Two parameters relative to a variable  $X_i$  are considered dissimilar if the difference of their degrees is greater than 0.1. Figure 1.21 presents obtained values.

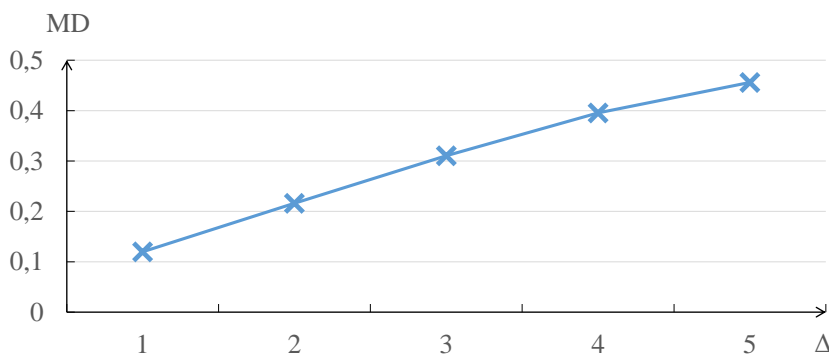


Figure 3.3: Evolution of approximation of Manhattan distance w.r.t.  $\Delta$

Figure 3.3 shows that Manhattan distance values increase when we increase  $\Delta$ , as expected. This is an obvious result, because when we increase the dissimilarity between two networks, we inflate considerably the gap between their joint possibility distributions.

The second objective of this set of experiments is to evaluate Manhattan distance approximation i.e. fixing the size of Sub- $\Omega$  and verifying if values obtained with this approximation (Equation 3.3) converge to the ones obtained with Manhattan distance (Equation 1.21). So, we rely on the following process: we generate in each experiment two possibilistic networks composed of  $nb$  variables generated from  $\{5, 10, 20, 30\}$  variables varying the maximum number of parents between  $\{2, 4, 8\}$  and the maximum number of variables domains cardinality between  $\{2, 5, 10\}$ . Then, we compute Manhattan distance between the two generated possibilistic networks joint distributions computed with the chain rule (Equation 2.8) and its approximation. The whole process is illustrated by Figure 3.4.

The first result of this set of experiments consists in fixing the size of Sub- $\Omega$ . In fact, for each possibilistic networks, several sizes (100, 1000, 2000, 5000 and 10000) were chosen to observe the behavior of Manhattan distance. Figure 3.5 shows the mean of the absolute difference between Manhattan distance values and its approximation values w.r.t. different sizes of Sub- $\Omega$ . Best values are obtained when the size of Sub- $\Omega$  is 5000. Note that in the case of small networks 15% of the cardinality of  $\Omega$  is sufficient to compute Manhattan distance approximation. Consequently, the size of Sub- $\Omega$  is the minimum between 5000 and 15% of the cardinality of  $\Omega$ .

The second result of this set of experiments illustrated in 3.6 shows that values obtained by the approximation converge to theoretical values obtained with Manhattan distance. Consequentially, we can use Manhattan distance approximation to compare two possibilistic networks. Note that Manhattan distance is insensitive to the structure of studied networks. In fact, increasing dissimilarity between two possibilistic networks structures (computed using editing distance) does not increase necessarily the dissimilarity between the two networks.

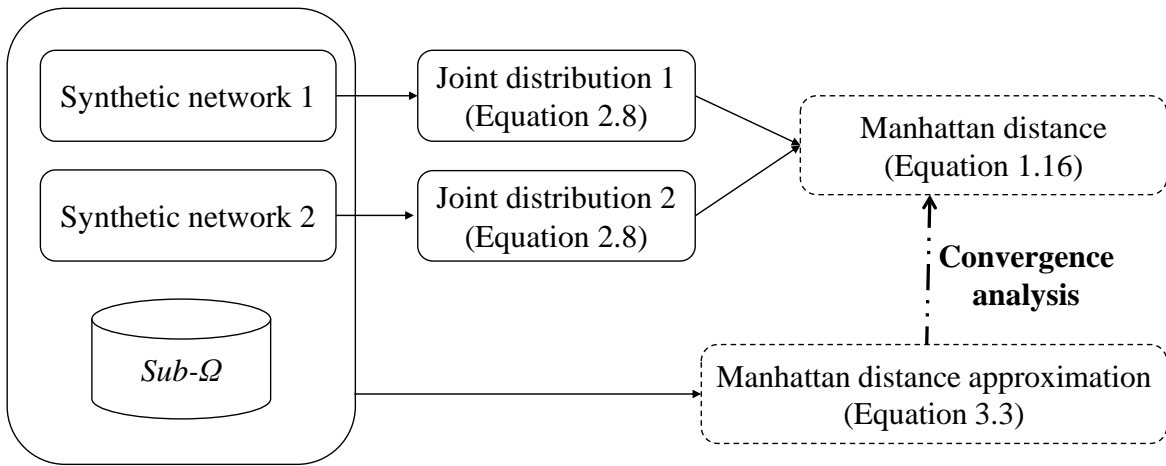


Figure 3.4: Experimental protocol of Manhattan distance approximation evaluation

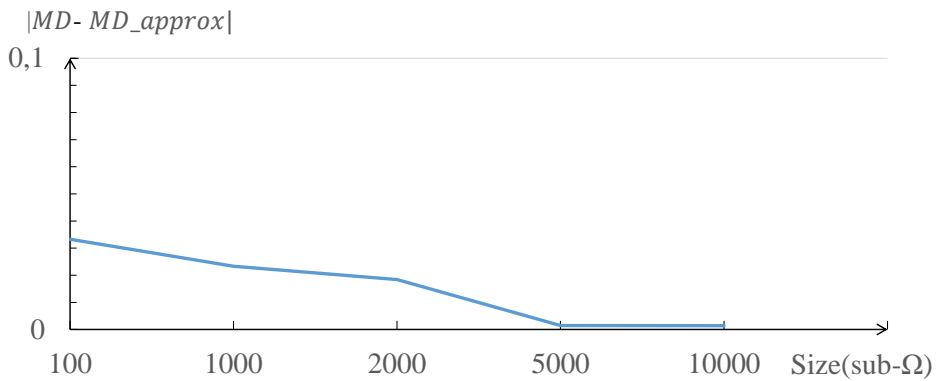


Figure 3.5: Evolution of convergence of Manhattan distance approximation values to Manhattan distance values w.r.t. sub-Ω

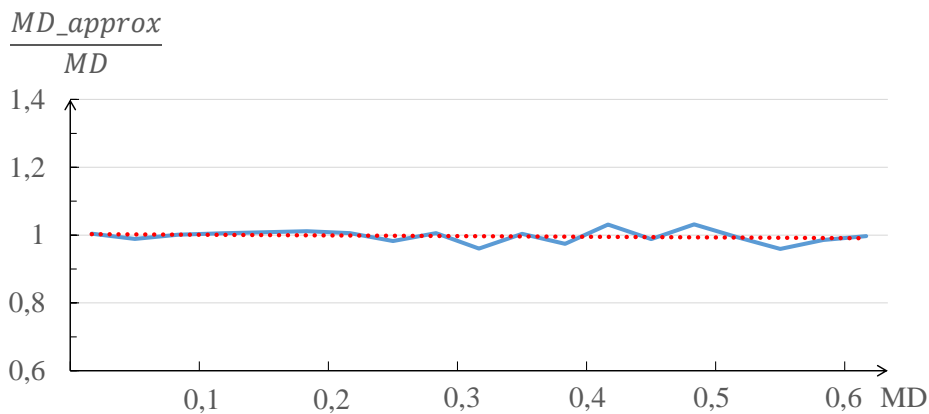


Figure 3.6: Evolution of convergence of Manhattan distance approximation values to Manhattan distance values

### 3.5 Conclusion

In this chapter, we propose a new evaluation strategy for *product-based* possibilistic networks learning algorithms. The first step of the evaluation process consists in generating random possibilistic networks

since there are currently no publicly available ones that can be used as gold networks. Then, we propose a sampling method to generate datasets from possibilistic networks in which data relative to a variable are conditionally consonant to its parents. To overcome this limitation, we propose two variants of this sampling method generating more generic form of imperfect datasets by controlling their consistency and imprecision degrees. Obtained benchmarks (generated networks and datasets) could be used to evaluate many applications e.g. approximate inference in possibilistic networks. Moreover, generated imprecise datasets could be used in many problems such as classification/clustering techniques handling this kind of imperfect data.

The second main contribution of this chapter concerns learning evaluation measures. We propose two new possibilistic measures, namely, mean information affinity and approximation of Manhattan distance. The first measure is restricted to the case of comparing two networks sharing the same structure and the second one is more generic and compares two possibilistic networks without any restriction.

The proposed evaluation strategy presents a clear experimental framework allowing the realization of a deep comparative study of the proposed possibilistic networks learning methods regarding existing ones. Next chapter details our second contribution that consists on a new approach to learn possibilistic networks parameters based on a new possibilistic likelihood function.



# Possibilistic networks parameters learning from imperfect data

## Contents

<b>4.1</b>	<b>Introduction</b>	<b>47</b>
<b>4.2</b>	<b>Building graphical models from imperfect data</b>	<b>48</b>
<b>4.3</b>	<b>Evaluation of existing parameters learning algorithms</b>	<b>48</b>
4.3.1	Learning parameters based on transformations	50
4.3.2	Learning parameters directly from data	50
4.3.3	Discussion	51
<b>4.4</b>	<b>Possibilistic-likelihood-based parameters learning algorithm</b>	<b>52</b>
4.4.1	Imprecise likelihood	52
4.4.2	Parameters learning algorithm from imprecise data	54
<b>4.5</b>	<b>Experimental study</b>	<b>55</b>
4.5.1	Possibilistic networks parameters learning evaluation	55
4.5.2	Particular case: possibilistic classifiers parameters learning evaluation	58
<b>4.6</b>	<b>Conclusion</b>	<b>59</b>

## 4.1 Introduction

The main concern of recent research endeavors dedicated to graphical models is how to learn them from imperfect data. In this chapter, we cover the problem of parameters learning of possibilistic networks from imprecise datasets, i.e., containing multi-valued data. As discussed in Chapter 2, only few works address this problem and existing ones (Borgelt et al., 2009; Sangüesa et al., 1998) are direct adaptations of Bayesian networks learning methods without any awareness of specificities of the possibilistic framework which made them theoretically unsound. The main limitation of existing works is that they assume that learning the parameters, i.e. possibility distributions coding variables uncertainty, and the structure i.e. the graph of the possibilistic network are two separated tasks.

In this chapter, we first discuss how the choice of the graphical model depends on data we learn from and we correspond to each type of data the most suitable reasoning model to present it. Then, we propose to test existing methods proposed to estimate possibility distributions in the context of possibilistic networks



and we show that none of these methods is satisfactory. Thereby, we explore the link between random sets theory (additive) and possibility theory (maxitive) to propose a new possibilistic likelihood function which will be deployed to learn possibilistic networks parameters.

This chapter is organized as follows: Section 4.2 presents a cartography relative to different graphical reasoning models described in Chapter 2 w.r.t. available data. In Section 4.3, we test possibility distributions estimation methods invoked in Chapter 1 in the context of possibilistic networks. Section 4.4 proposes a new possibilistic likelihood function which will be the basis of a possibilistic networks parameters learning method presented in Section 4.4.2. Section 4.5 shows the efficiency of the proposed possibilistic-likelihood-based learning method when applied to possibilistic networks and possibilistic classifiers.

## 4.2 Building graphical models from imperfect data

Building graphical models could be classified into two methods: the first family gathers expert elicitation techniques which consist in elaborating the synthesis of experts opinions when available data are very limited or unattainable because of physical constraints or lack of resources. As examples of such methods, we cite (Pearl, 1988) for Bayesian networks, (Antonucci, 2011) for credal networks, (Ben Amor et al., 2009) for *min-based* possibilistic networks and finally (Zhou et al., 2016) for evidential networks.

The second family gathers graphical models learning methods which depend chiefly on data nature i.e. perfect or imperfect data. In real world applications, data are inseparably connected with imperfection resultant from unreliable data sources such as measurement devices or aggregating opinions of different experts. In this case, choosing the most suitable graphical model that can handle available imperfect data corresponds to the most difficult task in a modeling problem. At first stage, we should check if available data are complete i.e. a unique value is assigned to each variable in the dataset of the studied domain, otherwise, data are considered incomplete. If we are faced to complete data i.e. we are in a *perfect* situation, probability theory is the most adequate framework to handle precisely described domains. Consequently, the most suitable graphical models handling this case are Bayesian networks. In the case of *insufficient* complete data i.e. few data are available (small dataset), the imprecision could be due to limited data and in this context, we may appeal to imprecise probability theory and consequently, we can represent this imprecision by a credal network.

In the case of incomplete data, two cases are considered: the partial ignorance defined by multi-valued data in which a variable is not precisely described and to which we assign more than one value. In such case, evidential, credal and possibilistic networks stand out as best alternatives (Smets and Kennes, 1994) to Bayesian networks. Note that credal networks are known by their high computational complexity. So, evidential and possibilistic network are preferred and more suited to perform propagation algorithms (See Chapter 2). The extreme case of partial ignorance corresponds to total ignorance of the variable real value of considered as missing. In this case, we can use Bayesian networks learned by the standard Expectation Maximization method.

In Figure 4.1, we propose a cartography relative to different graphical reasoning models w.r.t. available data (source and data nature).

## 4.3 Evaluation of existing parameters learning algorithms

In this section, we propose to test two solutions inspired by existing methods proposed to estimate possibility distributions (described in Sections 1.7.1 1.7.3 and 1.4). The first naive solution is to use probability possibility transformations. First, we learn probability distributions of each node  $P(X_i|Pa(X_i))$  from precise data. Then, we transform the obtained distributions into possibility ones. The second method is to use estimated possibility distributions using possibilistic histograms described in subsection 1.4. We will show that these two solutions are unsatisfactory.

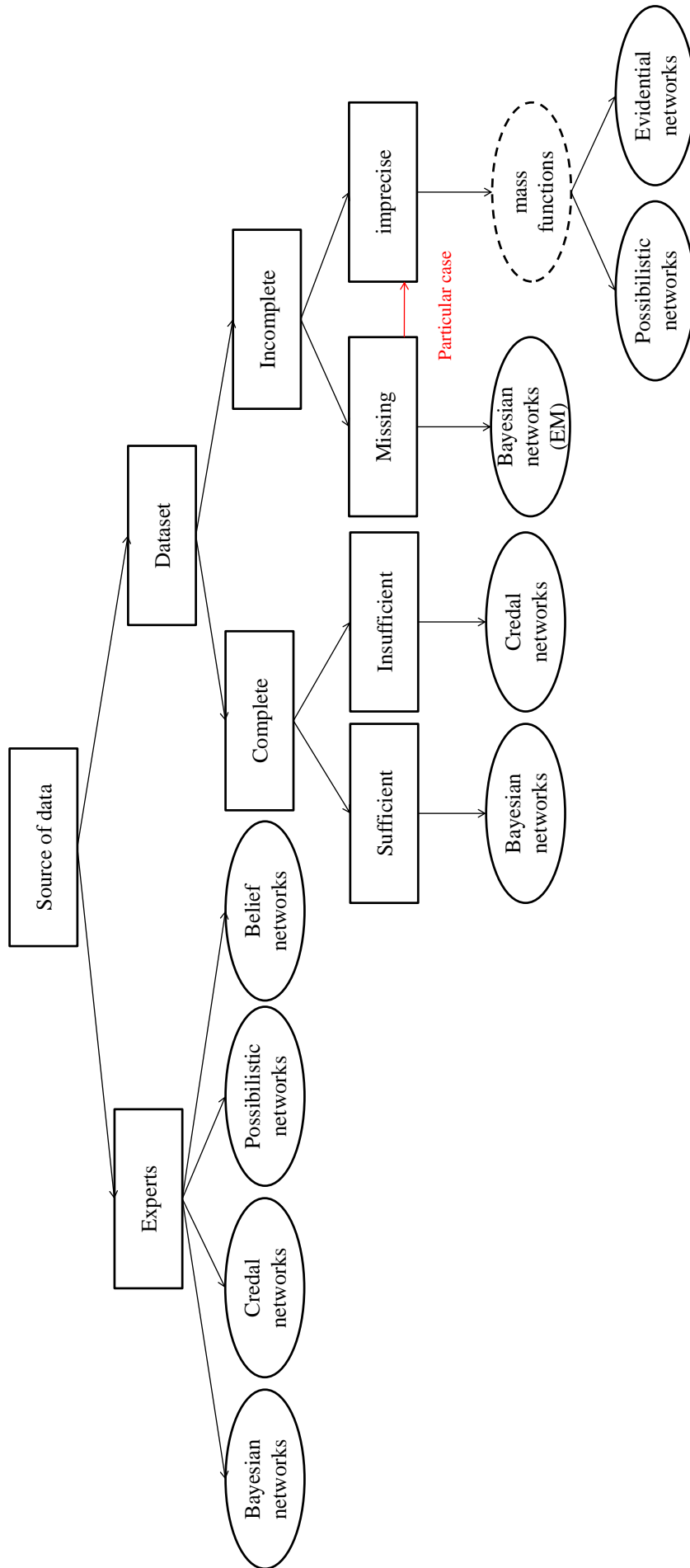


Figure 4.1: Cartography of reasoning models w.r.t. available data

### 4.3.1 Learning parameters based on transformations

The first solution consists in learning a Bayesian network from precise data then transforming obtained probability distributions to possibility ones (cf. Sections 1.7.1 and 1.7.3). As probability possibility transformations make sense in the numerical interpretation of the the possibilistic scale, we can learn only product-based possibilistic networks parameters using these transformations. This approach is based on the relationship between maximum likelihood proposed in the probabilistic framework and possibility distributions (Dubois, 2006): *When prior probabilities are lacking, likelihood functions can be interpreted as possibility distributions, by default.* Recall that transformation methods manipulate probability distributions or imprecise probability distributions. In what follows, we present an example of a transformation of each category, i.e. *optimal transformation* (cf. Section 1.7.1) and *Klir transformation adaptation* (cf. Section ).

**Example 4.3.1.** *Let us consider the Bayesian network described by Table 4.1 be composed of two variables  $X_1$  and  $X_2$  such that  $D_1 = \{x_{11}, x_{12}, x_{13}\}$  and  $D_2 = \{x_{21}, x_{22}\}$ .*

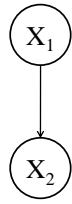
	$X_1$	$P(X_1)$	$X_1$	$X_2$	$P(X_1 X_2)$	$X_1$	$X_2$	$P(X_1, X_2)$
	$x_{11}$	0.2	$x_{11}$	$x_{21}$	0.4	$x_{11}$	$x_{21}$	0.08
	$x_{12}$	0.3	$x_{11}$	$x_{22}$	0.6	$x_{11}$	$x_{22}$	0.12
	$x_{13}$	0.5	$x_{12}$	$x_{21}$	0.9	$x_{12}$	$x_{21}$	0.27
			$x_{12}$	$x_{22}$	0.1	$x_{12}$	$x_{22}$	0.03
			$x_{13}$	$x_{21}$	0.3	$x_{13}$	$x_{21}$	0.15
			$x_{13}$	$x_{22}$	0.7	$x_{13}$	$x_{22}$	0.35

Table 4.1: Example of a Bayesian network

Tables 4.2 and 4.3 represent possibilistic networks obtained by transforming this Bayesian network.  $\pi(X_1, X_2)$  represents the joint possibility distribution obtained by transforming the initial Bayesian network joint distribution.  $\pi_t(X_1, X_2)$  is the joint possibility distribution computed from conditional possibility distributions using Equation 2.8. Example (a) uses optimal transformation and Example (b) uses Klir transformation adaptation using confidence intervals. Note that  $\pi(X_1, X_2)$  and  $\pi_t(X_1, X_2)$  values are different.


	$X_1$	$\pi(X_1)$	$X_1$	$X_2$	$\pi(X_2 X_1)$	$X_1$	$X_2$	$\pi_t(X_1, X_2)$	$\pi(X_1, X_2)$
	$x_{11}$	0.2	$x_{11}$	$x_{21}$	0.4	$x_{11}$	$x_{21}$	0.08	0.11
	$x_{12}$	0.5	$x_{11}$	$x_{22}$	1	$x_{11}$	$x_{22}$	0.2	0.23
	$x_{13}$	1	$x_{12}$	$x_{21}$	1	$x_{12}$	$x_{21}$	0.5	0.65
			$x_{12}$	$x_{22}$	0.1	$x_{12}$	$x_{22}$	0.05	0.03
			$x_{13}$	$x_{21}$	0.3	$x_{13}$	$x_{21}$	0.3	0.38
			$x_{13}$	$x_{22}$	1	$x_{13}$	$x_{22}$	1	1

Table 4.2: Example (a) of transformation of Bayesian network in a possibilistic network using the optimal transformation (Equation 1.34)

As shown by Example 4.3.1, using possibility probability transformations to estimate possibilistic networks parameters leads to a loss of information i.e. we cannot recover the transformed joint possibility distribution by aggregating transformed conditional possibility distributions.

### 4.3.2 Learning parameters directly from data

The second solution consists on applying Joslyn method (cf. Section 1.4) to learn possibilistic networks parameters from imprecise data.

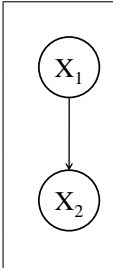
	$X_1$	$\pi(X_1)$	$X_1$	$X_2$	$\pi(X_2 X_1)$	$X_1$	$X_2$	$\pi_t(X_1, X_2)$	$\pi(X_1, X_2)$
	$x_{11}$	0.78	$x_{11}$	$x_{21}$	0.68	$x_{11}$	$x_{21}$	0.54	0.28
	$x_{12}$	1	$x_{11}$	$x_{22}$	1	$x_{11}$	$x_{22}$	0.78	0.41
	$x_{13}$	1	$x_{12}$	$x_{21}$	1	$x_{12}$	$x_{21}$	1	0.89
			$x_{12}$	$x_{22}$	0.22	$x_{12}$	$x_{22}$	0.22	0.11
			$x_{13}$	$x_{21}$	0.51	$x_{13}$	$x_{21}$	0.51	0.51
			$x_{13}$	$x_{22}$	1	$x_{13}$	$x_{22}$	1	1

Table 4.3: Example (b) transformation of Bayesian network in a possibilistic network using Klir transformation adaptation (Equation 1.41)

**Example 4.3.2.** Let us consider the imprecise dataset described by Table 1.2. If we apply Joslyn method, we obtain the possibility distributions presented in Table 4.5. To estimate conditional possibility distributions, we compute for example  $\hat{\pi}^n(x_{21}|x_{11})$  by dividing  $\hat{f}^{nn}(x_{11}, x_{21}) = 0.6$  by  $\hat{f}^{nn}(x_{11}) = 0.6$  obtained by Equation 1.14, then, we normalize possibility degrees. Note that obtained values of  $\hat{\pi}^n(X_1, X_2)$ , the joint possibility distribution estimated from data and  $\hat{\pi}_l^n(X_1, X_2)$ , the joint possibility distribution computed from conditional distributions (Equation 2.8), are different.

$X_1$	$X_2$
$x_{11}, x_{12}$	$x_{21}, x_{22}$
$x_{12}, x_{13}$	$x_{21}, x_{22}$
$x_{11}, x_{12}$	$x_{21}$
$x_{11}, x_{13}$	$x_{21}$
$x_{12}$	$x_{22}$

Table 4.4: Example of an imprecise dataset

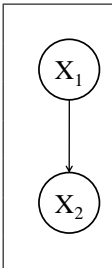
	$X_1$	$\hat{\pi}^n(X_1)$	$X_1$	$X_2$	$\hat{\pi}^n(X_1 X_2)$	$X_1$	$X_2$	$\hat{\pi}_l^n(X_1, X_2)$	$\hat{\pi}^n(X_1, X_2)$
	$x_{11}$	0.75	$x_{11}$	$x_{21}$	1	$x_{11}$	$x_{21}$	0.75	1
	$x_{12}$	1	$x_{11}$	$x_{22}$	0.33	$x_{11}$	$x_{22}$	0.25	0.33
	$x_{13}$	0.5	$x_{12}$	$x_{21}$	1	$x_{12}$	$x_{21}$	1	1
			$x_{12}$	$x_{22}$	1	$x_{12}$	$x_{22}$	1	1
			$x_{13}$	$x_{21}$	1	$x_{13}$	$x_{21}$	0.5	0.66
			$x_{13}$	$x_{22}$	0.5	$x_{13}$	$x_{22}$	0.25	0.33

Table 4.5: Example of learning possibilistic networks parameters

As shown by Example 4.5, using Joslyn’s method to estimate possibilistic networks parameters leads to a loss of information i.e. we cannot recover the joint possibility distribution by aggregating obtained conditional possibility distributions.

### 4.3.3 Discussion

In the previous subsections, we have applied two existing methods to learn possibilistic networks parameters: probability possibility transformation and direct possibility distribution estimation. We have shown with two simple examples that none of these methods is satisfactory. Examples 4.3.1 and 4.3.2 show that these methods could be applied only with the joint possibility distribution and not with local distributions separately. Transformation methods detailed in Section 1.7.1 have been already studied in (Ben Slimen et al., 2013) and this paper confirms our conclusion. Remaining transformation methods present the same

inconvenient. This is due to the fact that marginalization notion of probability measures using sum operator is very different to marginalization of possibility measures applying maximum.

Therefore, we should apply these methods cautiously: do we need primarily estimating local parameters (usual interest of probabilistic and possibilistic graphical models) or directly estimate the joint distribution ignoring the interest of graphical decomposition?

Since probability possibility transformations are restricted to precise datasets which represent a particular case of data we use to learn possibilistic networks parameters, we investigate, in the following, the use of Equation 1.14 and we propose a new possibilistic-likelihood-based parameters learning method.

## 4.4 Possibilistic-likelihood-based parameters learning algorithm

In the probabilistic case, learning Bayesian networks parameters is performed satisfying *maximum likelihood* principle. As far as we know, such a measure has not been proposed in the context of possibilistic networks parameters learning. Moreover, working on parameters in the possibilistic framework highlights several difficulties when dealing with the learning task, in particular, when we handle uncertain and imprecise data. This is due to the fact that learning is usually viewed as an objective task while possibility theory has been almost always based on the subjective opinions. That is to say, the absence of a learning possibilistic networks parameters method could be justified by the fact that learning leads commonly to additive assessment, i.e., based on computing frequency of observations while possibility theory is, by definition, maxitive, i.e., the possibility of a disjunction of events is the maximum of the possibilities of each event in this disjunction.

This is to some extent true, especially, when we deal with measurement devices leading to precise observations (one possible value per variable). In this case, probability theory remains the most adequate alternative. However, when measurement devices provide imprecise data and we want to model data as they have been collected i.e. including imprecision due to the physical measurement itself, non-classical uncertainty theories stand out as best alternatives. In our case, we choose to use possibility theory since it is able to offer a natural and simple formal framework representing imprecise and uncertain information. The latter refers to the study of maxitive and minitive set-functions and can be interpreted as an approximation of upper and lower frequentist set probabilities in the presence of imprecise observations. Thereby, if we want to learn parameters from data in the possibilistic framework, two steps are primordial: the first one (additive) focuses in counting the occurrence of observations in the dataset to estimate non-normalized distributions. While the second step (maxitive) aims to approximate the latter by possibility distributions.

### 4.4.1 Imprecise likelihood

The likelihood function is central to the process of learning parameters of a model, i.e., for a fixed dataset and an underlying model, maximum likelihood finds the most likely values for the model parameters based on a dataset. In the following, we propose a possibilistic likelihood function exploring the link between possibility theory and random sets theory in Definition 1.40 (cf. Section 1.7.2). Consequently, the formulation of our likelihood function is made in two steps: first, we propose a likelihood function defined on random sets (additive). Then, we propose a possibilistic likelihood function exploring the link between possibility theory (maxitive) and random sets theory and we show that the two functions are different but lead to the same possibility distributions. The proposed likelihood function will be used later to learn possibilistic networks parameters.

#### Random sets likelihood

The random sets likelihood extends the probabilistic one described in Section 2.3.3 by replacing the probability distribution by mass functions (cf. Section 1.7.2).

**Definition 4.4.1.** Let  $G$  be a DAG and  $\{m_1, m_2, \dots, m_n\}$  be the parameters relative to  $\{X_1, X_2, \dots, X_n\}$  to be estimated and  $\mathcal{D}_{ij} = \{d_{ij}^{(l)}\}$  be a dataset relative to a variable  $X_i$  and its parents  $Pa(X_i) = j$ ,  $d_{ij}^{(l)} \subseteq D_{ij}$ . The number of occurrences of each  $A_{ik} \subseteq D_i$  such that  $Pa(X_i) = j$  ( $j \subseteq D_j$ ), denoted by  $N_{ijk}$ , is the number of times  $A_{ijk}$  appears in  $\mathcal{D}_{ij}$ :  $N_{ijk} = \text{card}(\{l \text{ s.t. } A_{ijk} = d_{ij}^{(l)}\})$ . We express the likelihood function as follows:

$$mL(m, G, \mathcal{D}) = \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} N_{ijk} \log m_{ijk} \tag{4.1}$$

where  $mL$  is expressed by random sets of domains variables i.e. for each  $X_i$ ,  $q_i = 2^{\text{card}(Pa(X_i))}$  and  $r_i = 2^{\text{card}(D_i)}$ ,  $m_{ijk}$  is the parameter to be estimated when  $X_i = A_{ik}$  and  $Pa(X_i) = A_j$ .

For numerical stability reasons, we propose the log-likelihood function. Equation 4.1 becomes:

$$mLL(m, G, \mathcal{D}) = \sum_{i=1}^n \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log m_{ijk} \tag{4.2}$$

Note that mass functions associated to random sets are probability distributions, the partial derivative of the  $mLL(m, G, \mathcal{D})$  follows the same principle of the partial derivative of the probabilistic likelihood function and reaches its maximum in  $\hat{m}_{ijk} = \frac{N_{ijk}}{\sum_{k=1}^{r_i} N_{ijk}}$ .

**Example 4.4.1.** Let us consider the imprecise dataset  $\mathcal{D}$  in Table 4.6 and the network in Table 4.7 composed of two variables  $X_1$  and  $X_2$ .  $mLL(m, G, \mathcal{D})$  relative to this network is:

$X_1$	$X_2$	nb of occurrences
$x_{12}$	$x_{22}$	3
$x_{12}$	$x_{21}, x_{22}$	3
$x_{11}, x_{12}$	$x_{22}$	3
$x_{11}$	$x_{21}, x_{22}$	1

Table 4.6: Example of an imprecise dataset

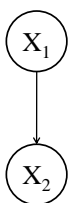
	$X_1$	$m(X_1)$	$X_1$	$X_2$	$m(X_2 X_1)$
		$x_{11}$	0.1	$x_{11}$	$x_{21}$
	$x_{12}$	0.6	$x_{11}$	$x_{22}$	0.3
	$x_{11}, x_{12}$	0.3	$x_{11}$	$x_{21}, x_{22}$	0.5
			$x_{12}$	$x_{21}$	0.1
			$x_{12}$	$x_{22}$	0.7
			$x_{12}$	$x_{21}, x_{22}$	0.2
			$x_{11}, x_{12}$	$x_{21}$	0.1
			$x_{11}, x_{12}$	$x_{22}$	0.5
			$x_{11}, x_{12}$	$x_{21}, x_{22}$	0.4

Table 4.7: Example of a network with two variables defined on random sets

$$mLL(m, G, \mathcal{D}) = \log(0.1) + 6 \log(0.6) + 3 \log(0.3) + 3 \log(0.7) + 3 \log(0.2) + 3 \log(0.5) + \log(0.5) = -7.6655.$$

	$X_1$	$\pi(X_1)$	$X_1$	$X_2$	$\pi(X_2 X_1)$
	$x_{11}$	0.1	$x_{11}$	$x_{21}$	0.2
	$x_{12}$	1	$x_{11}$	$x_{22}$	1
			$x_{12}$	$x_{21}$	0.1
			$x_{12}$	$x_{22}$	1

Table 4.8: Example of a possibilistic network with two binary variables

### Possibilistic likelihood

Computing random sets likelihood function is computationally expensive. In fact, a random set relative to a variable  $X_i$  is defined on  $2^{\text{card}(D_i)}$  and its cardinality grows exponentially with the the number of values in  $D_i$  (Dubois and Prade, 1990). Consequently, we propose to investigate the link between possibility distributions and mass functions presented in Equation 1.40 and to define a possibilistic approximation of random sets likelihood function. In the following, we will replace mass functions in Equation 4.2 by possibility distributions defined on singletons and we will study the link between the two proposed likelihood function by checking if the possibility distributions computed by maximizing the random sets likelihood recover the ones obtained by maximizing the possibilistic one.

We express the possibilistic likelihood function as follows:

**Definition 4.4.2.** Let  $G$  be a DAG and  $\{\pi_1, \pi_2, \dots, \pi_n\}$  be the parameters relative to  $\{X_1, X_2, \dots, X_n\}$  to be estimated and  $\mathcal{D}_{ij} = \{d_{ij}^{(l)}\}$  be a dataset relative to a variable  $X_i$  and its parents  $Pa(X_i) = j$ ,  $d_{ij}^{(l)} \subseteq D_i$ . The number of occurrences of each  $x_{ik} \in D_i$  such that  $Pa(X_i) = j$ , denoted by  $N_{ijk}$ , is the number of times  $x_{ijk}$  appears in  $\mathcal{D}_{ij}$ :  $N_{ijk} = |\{l \text{ s.t. } x_{ijk} \subseteq d_{ij}^{(l)}\}|$ . We express the possibilistic likelihood as follows:

$$\pi LL(\pi, G, \mathcal{D}) = \sum_{i=1}^n \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log \pi_{ijk} \quad (4.3)$$

where for each  $X_i$ ,  $q_i = \text{card}(Pa(X_i))$  and  $r_i = \text{card}(D_i)$ ,  $\pi_{ijk}$  is the parameter to be estimated when  $X_i = x_{ik}$  and  $Pa(X_i) = x_j$ .

**Example 4.4.2.** Let us reconsider the imprecise dataset  $\mathcal{D}$  in Table 4.6 and the possibilistic network in Table 4.8 composed of two variables  $X_1$  and  $X_2$ .  $\pi LL(\pi, G, \mathcal{D})$  relative to this network is:

$$\pi LL(\pi, G, \mathcal{D}) = 4 \log(0.1) + 9 \log(1) + 1 \log(0.2) + 4 \log(1) + 3 \log(0.1) + 9 \log(1) = -7.6990.$$

### 4.4.2 Parameters learning algorithm from imprecise data

In what follows, we use the possibilistic likelihood in Definition 4.4.2 to learn possibilistic networks parameters i.e. to find the most likely values for their parameters based on a dataset.

**Proposition 4.4.1.** Given a DAG, a fixed parameter  $\pi_{ijk}$  and an imprecision degree  $S_i$  (prefixed value) relative to the variable  $X_i$  the maximum possibilistic likelihood estimates are the parameter values that maximize  $\pi LL(\pi, G, \mathcal{D})$ . We assume that  $\sum_{k=1}^{r_i} \pi_{ijk}$  is a constant equal to  $S_i$ ,  $\pi LL(\pi, G, \mathcal{D})$  reaches it maximum in:

$$\hat{\pi}_{ijk} = \text{argmax}(\pi LL(\pi, G, \mathcal{D})) = \frac{N_{ijk}}{\sum_{k=1}^{r_i} N_{ijk}} * S_i \quad (4.4)$$

*Proof.* Let  $S_i$  be  $\sum_{k=1}^{r_i} \pi_{ijk}$ . So, the parameters  $\pi_{ijk}$  are related by the following formula:  $\pi_{ijr_i} = S_i - \sum_{k=1}^{r_i-1} \pi_{ijk}$ . Then,  $\pi LL(\pi, G, \mathcal{D})$  could also be rewritten as follows:

$$\pi LL(\pi, G, \mathcal{D}) = \sum_{i=1}^n \sum_{j=1}^{q_i} \left( \sum_{k=1}^{r_{i-1}} N_{ijk} \log \pi_{ijk} \right) + N_{ijr_i} \log \left( S_i - \sum_{k=1}^{r_i-1} \pi_{ijk} \right)$$

So, its derivative w.r.t. a parameter  $\pi_{ijk}$  is:

$$\frac{\partial \pi LL(\pi, G, \mathcal{D})}{\partial \pi_{ijk}} = \frac{N_{ijk}}{\pi_{ijk}} = \frac{N_{ijr_i}}{S - \sum_{k=1}^{r_i-1} \pi_{ijk}} = \frac{N_{ijk}}{\pi_{ijk}} - \frac{N_{ijr_i}}{\pi_{ijr_i}}$$

So, the value  $\hat{\pi}_{ijk}$  of the parameter of  $\pi_{ijk}$  maximizing the possibilistic likelihood sets this derivative equal to 0 and satisfies thereby:

$$\frac{N_{ijk}}{\hat{\pi}_{ijk}} = \frac{N_{ijr_i}}{\hat{\pi}_{ijr_i}}$$

We have:

$$\frac{N_{ij1}}{\hat{\pi}_{ij1}} = \frac{N_{ij2}}{\hat{\pi}_{ij2}} = \dots = \frac{N_{ijr_{i-1}}}{\hat{\pi}_{ijr_{i-1}}} = \frac{N_{ijr_i}}{\hat{\pi}_{ijr_i}} = \frac{\sum_{k=1}^{r_i} N_{ijk}}{\sum_{k=1}^{r_i} \hat{\pi}_{ijk}} = \frac{\sum_{k=1}^{r_i} N_{ijk}}{S_i}$$

$$\text{So, } \hat{\pi}_{ijk} = \frac{N_{ijk}}{\sum_{k=1}^{r_i} N_{ijk}} * S_i. \quad \square$$

Consequently, given a network structure, Equation 4.4 will be applied for each in the context of its parents. Note that  $S_i$  corresponds to the imprecision degree relative to a variable  $X_i$  and could be fixed by an expert, inferred from the dataset to learn from or based on variables description. To obtain normalized possibility distributions, we divide every obtained distribution by its maximum. This operation will eliminate the effect of the imprecision degree. However, it remains possible to fix an imprecision degree per value of variables of the studied domain.

It is evident that random sets likelihood and possibilistic likelihood functions are not equivalent. However, parameters obtained by combining random sets likelihood and the mass possibility transformation, denoted by  $\text{trans}_{m \rightarrow \pi}$  in Definition 1.40 leads to the ones obtained by maximizing directly possibilistic likelihood in Definition 4.4.1.

**Proposition 4.4.2.**  $\text{argmax}(\pi LL(\pi_{ijk}, G, \mathcal{D})) = \frac{\text{trans}_{m \rightarrow \pi}(\text{argmax}(mLL(m_{ijk}, G, \mathcal{D})))}{\text{trans}_{m \rightarrow \pi}(\text{argmax}(mLL(m_j, G, \mathcal{D})))}$

$$\text{Proof: } \text{argmax}(\pi LL(\pi_{ijk}, G, \mathcal{D})) = \frac{N_{ijk}}{\sum_{k=1}^{r_i} N_{ijk}} = \frac{\frac{N_{ijk}}{N}}{\sum_{k=1}^{r_i} \frac{N_{ijk}}{N}} = \frac{\sum_{A_{ijk} | x_{ijk} \in A_{ijk}} \frac{N_{A_{ijk}}}{N}}{\sum_{A_j | j \in A_j} \frac{N_{A_j}}{N}} = \frac{\text{trans}_{m \rightarrow \pi} \frac{N_{A_{ijk}}}{N}}{\text{trans}_{m \rightarrow \pi} \frac{N_{A_j}}{N}} = \frac{\text{trans}_{m \rightarrow \pi}(\text{argmax}(mLL(m_{ijk}, G, \mathcal{D})))}{\text{trans}_{m \rightarrow \pi}(\text{argmax}(mLL(m_j, G, \mathcal{D})))}. \quad \square$$

## 4.5 Experimental study

The objective of these two sets of experiments is the evaluation of our proposed possibilistic networks parameters learning approach. The first set is dedicated to generic possibilistic networks and the second one concerns the particular case of possibilistic classifiers.

### 4.5.1 Possibilistic networks parameters learning evaluation

To evaluate our parameters learning method, we generate synthetic datasets containing 100, 500 and 1000 imprecise observations from 15 randomly generated possibilistic networks composed of {10, 20, 30} variables (3 datasets<sup>1</sup> for each dataset size). We also vary the maximum number of parents between {2, 4, 8} and the maximum number of variables domains cardinality between {2, 5, 10} in generated possibilistic networks. Then, we compare the initial network using mean information affinity and approximation of

1. Datasets are available in <https://sites.google.com/site/karimtabiasite/mappos>



Manhattan distance. The first part of this set of experiments uses datasets generated using the sampling process proposed in Algorithm 1. Figures 4.3 and 4.4 present obtained results.

In the second part of this set of experiments, we compare our direct proposed possibilistic network learning method (DPNL) with the transformation-based learning method (TPNL) described in Section 2.7 using mean information affinity. These experiments are carried out on datasets containing 1000 observations. Obtained results are presented in Table 4.9. Note that this part of the experimental study was carried out under the project PEPS Fascido 2015 MAPPOS<sup>2</sup>. This set of experiments is detailed in Figure 4.2.

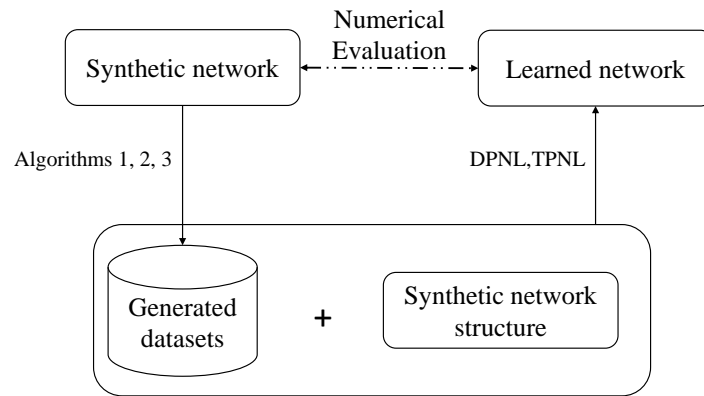


Figure 4.2: Proposed experimental protocol of possibilistic networks parameters learning evaluation

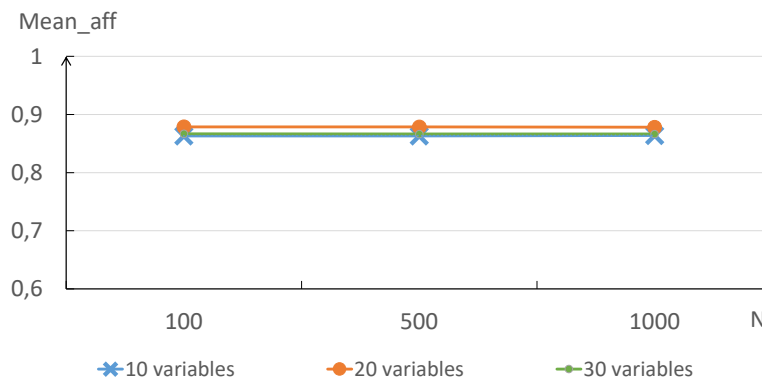


Figure 4.3: Mean information affinity between initial networks and learned networks varying datasets size

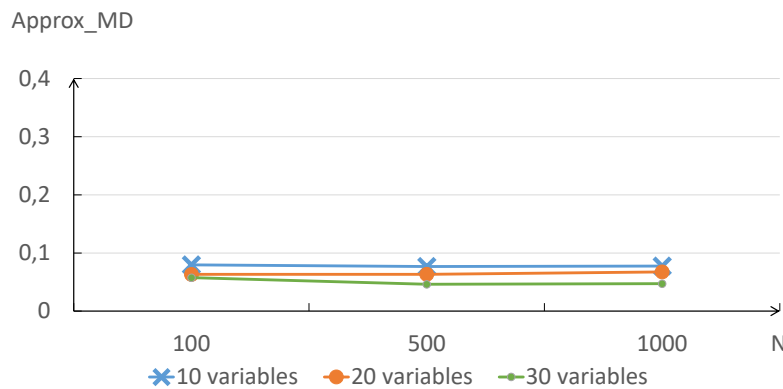


Figure 4.4: Manhattan distance approximation between initial networks and learned networks varying datasets size

2. <https://sites.google.com/site/karimtabiasite/mappos>

Figures 4.3 and 4.4 show that mean information affinity and approximation of Manhattan distance of learned networks do not seem to be affected by the number of variables, and number of observations.

number of variables	Mean-aff	
	TPNL	DPNL
10	0.63	0.86
20	0.65	0.86
30	0.67	0.86

Table 4.9: Mean information affinity between initial networks and networks learned using DPNL and TPNL

The results of mean information affinity in Table 4.9 show that learned possibilistic networks using our approach DPNL are very close/similar to the reference ones. Concerning the comparison of our approach with TPNL, the results of Table 4.3 show that DPNL leads to better networks than TPNL. This is expected as the datasets generation process and DPNL approach have the same view of possibility degrees. Moreover, TPNL uses a probability possibility transformation which generates necessarily a loss of information to move from the probabilistic framework to the possibilistic one. Note that the obtained similarity values do not seem to be affected by the number of variables, variable domains size, etc.

The second part is dedicated to studying the behavior of DPNL using mean information affinity when we vary the imprecision percentage (resp. the consistency percentage) in generated datasets obtained using Algorithm 2 (resp. Algorithm 3). In what follows, we choose to vary  $\theta_{imp}$  and  $\theta_{cons}$  between 20%, 40%, 60% and 80%.

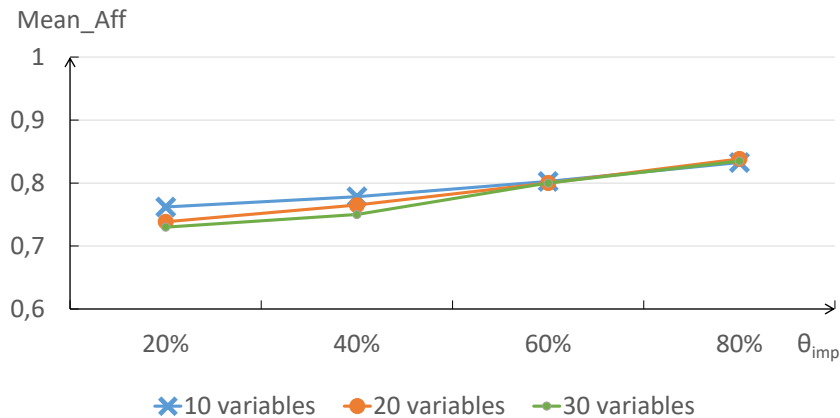


Figure 4.5: Evolution of mean information affinity between initial networks and learned networks distributions using DPNL w.r.t. datasets imprecision percentage

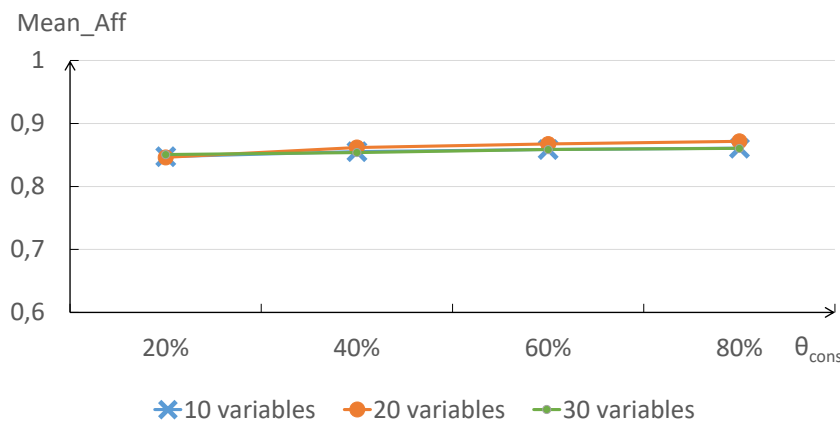


Figure 4.6: Evolution of mean information affinity between initial networks and learned networks distributions using DPNL w.r.t. datasets consistency percentage

Figures 4.5 and 4.6 show that reducing the imprecision degree or consistency degree of learning datasets affects the quality of *local* conditional possibility distributions. In fact, this operation introduces noise to the dataset which allows the emergence of corrupted dependencies not relevant to the problem. However, varying the consistency or the imprecision degrees have not a considerable effect on the *global* joint distributions of learned networks. This is an expected result since if we compute the information affinity between joint distributions directly from datasets generated from the same network using the three sampling algorithms (Algorithms 1, 2 and 3), we obtain similar distributions. Note that we have not presented the evolution of Manhattan distance approximation because obtained values are close to the ones obtained in the previous experimentation i.e. in which we use datasets generated by Algorithm 1.

## 4.5.2 Particular case: possibilistic classifiers parameters learning evaluation

The objective of this set of experiments is evaluating the predictive power of possibilistic network classifiers regarding credal network classifiers (NCC) Corani and Zaffalon (2008) and Bayesian network classifiers (NBC) Friedman et al. (1997). More precisely, we compare on many datasets the classification efficiency of NCC and the possibilistic classifiers PNCPA and PNCTA obtained using respectively our approach DPNL and TPNL. Moreover, we compare our results to naive Bayes classifier (NBC) as a baseline. Note that this part of the experimental study was carried out under the project PEPS Fascido 2015 MAPPOS<sup>3</sup>.

The evaluation mode used in this experiment is a 10-fold cross validation. The experimental study is carried out on the following datasets where some data values are missing. The first four datasets of Table 4.10 are real datasets<sup>4</sup> used in the literature for evaluating classifiers with missing data. The remaining ones are collected from different sources.

Name	#instances	#variables	#classes	% missing
Breast	286	9	2	4%
Housevotes	435	16	2	24%
Mushroom	8124	22	2	31%
Post-operative	90	8	3	3 %
Audiology	226	70	24	98%
Sick	3772	30	2	20%
Primary-tumor	339	18	21	46%
Kr-vs-kp	3196	37	2	0%
Soybean	683	36	19	18%
Crx	690	16	2	2%

Table 4.10: Description of Datasets

Results of Table 4.11 show that classifiers NBC, PNCPA and PNCTA have most of the time comparable results in terms of correct classification rates on some datasets but they show real performances on some other datasets. This is also valid for the results of the NCC classifier. Now, comparing PNCPA and PNCTA, this latter achieves better results on two datasets while the former has better classification rates on the two other datasets. It is not obvious what makes a given approach better, a thorough analysis of the properties of the datasets is needed to help understanding such results.

3. <https://sites.google.com/site/karimtabiasite/mappos>

4. <http://sci2s.ugr.es/keel/missing.php>

Dataset	NBC	PNCPA	PNCTA	NCC
Breast	72.88%	72.73%	70.27%	74.08%
Housevotes	90.11%	89.19%	58.71%	90.26%
Mushroom	95.73%	77.35%	85.34%	99.56%
Post-operative	68.11%	67.78%	71.11%	67.57%
Audiology	72.79%	55.90%	11.54%	99.55
Sick	96.97%	95.53%	94.41%	97.54%
Primary-tumor	49.54%	28.42%	43.42%	77.11%
Kr-vs-kp	87.82%	85.86%	86.89%	88.16%
Soybean	92.66%	92.56%	75.51%	92.56%
Crx	85.38%	85.80%	91.01%	86.34%

Table 4.11: % of correct classifications NBC, PNCPA, PNCTA and NCC classifiers on the datasets of Table 4.10

## 4.6 Conclusion

In this chapter, we have shown that learning possibility distributions from data and in particular conditional ones seems to be unnatural due to the semantic of the possibilistic framework. In fact, learning implies additive assessments while possibility theory is by definition maxitive. However, despite this constraint, the simplicity of representation of imperfect information in the possibilistic case, contrarily to other non-classical uncertainty frameworks, highlights the interest of learning possibility distributions even if they are only approximations of available data. Then, we propose a new approach to learn *product-based* possibilistic networks parameters from imprecise data based on a possibilistic likelihood function. The latter explores the link between random sets theory and possibility theory. The extended experimental study evaluates the efficiency of the evaluation strategy and shows that the possibilistic likelihood could be efficiently used to learn possibilistic networks parameters and possibilistic classifiers.

This work presents a first step in proposing a global *product-based* possibilistic networks learning approach i.e. including parameters and structure learning. So, in the next chapter, we intend to investigate the use of the possibilistic likelihood to learn possibilistic networks structure.



# Possibilistic networks structure learning from imprecise data

## Contents

<b>5.1 Introduction</b>	<b>61</b>
<b>5.2 New possibilistic score</b>	<b>62</b>
<b>5.3 Possibilistic adaptation of Greedy search algorithm</b>	<b>63</b>
<b>5.4 Experimental study</b>	<b>65</b>
<b>5.5 Conclusion</b>	<b>66</b>

## 5.1 Introduction

Learning graphical models structure consists in constructing the graph structure from data i.e. selecting a *good* model among different competing models that best describes (having the best fitness degree) observed data. The most common way to learn the structure of a graphical model, in particular, a Bayesian network is to use a scoring function combined with an optimization method.

As discussed Chapter 2, only a few methods for learning possibilistic networks structure have been presented in the literature and most of them are theoretically unsound. Within these methods, the unique attempt to learn them from *imprecise* data was carried out by Borgelt et al. (2009) and it addresses the problem of learning a DAG structure in the possibilistic framework (the process computes non-normalized marginal possibility distributions) but the output is a DAG without any quantification.

The aim of this chapter is to define and study a new scoring function based on the possibilistic likelihood proposed in Chapter 4. Combined with greedy search algorithm, this score will be used to learn possibilistic networks structure from imprecise data. A detailed experimental study showing the efficiency of the proposed method w.r.t. existing methods is also presented.

This chapter is organized as follows: Section 5.2 proposes a new possibilistic-likelihood-based score. Section 5.3 describes greedy search algorithm. Section 5.4 presents the experimental results showing the efficiency of the proposed structure learning method regarding existing possibilistic networks learning algorithms.

## 5.2 New possibilistic score

In this section, we try to apply Occam's razor principle in the context of learning possibilistic networks structure which is based on the following insight: the model to be selected is the one that is best balanced in terms of simplicity and fitness of given data. In fact, we propose a new possibilistic-likelihood-based score named *possibilistic AIC* ( $AIC_{pos}$ ), the possibilistic counterpart of the AIC score, which includes two terms: likelihood function (Equation 4.3) to quantify fitness between the graph and the data and complexity computed via the dimension of the graph. The latter corresponds to the sum over variables  $X_i$  of the number of parameters required to represent  $\pi(X_i|Pa(X_i))$ .

**Definition 5.2.1.** *The dimension of a possibilistic network  $G$  denoted by  $Dim(G)$  is the number of parameters required to represent its conditional possibility distributions and is expressed as follows:*

$$Dim(G) = \sum_{i=1}^n dim(X_i, G) \quad (5.1)$$

where

$$Dim(X_i, G) = |D_i| * \prod_{X_j \in Pa(X_i)} |D_j| \quad (5.2)$$

So, we define  $AIC_{pos}$  as follows:

$$AIC_{pos}(G|\mathcal{D}) = \pi LL(G, \mathcal{D}) - Dim(G) \quad (5.3)$$

In the following, we will check if  $AIC_{pos}$  satisfies decomposability and score equivalence properties:

–  $AIC_{pos}$  is decomposable as follows:

$$AIC_{pos}(G|\mathcal{D}) = \sum_{i=1}^n aic_{pos}(X_i|Pa(X_i)) \quad (5.4)$$

where

$$aic_{pos}(X_i|D) = \pi LL(X_i|Pa(X_i), \mathcal{D}) - Dim(X_i, G) \quad (5.5)$$

–  $AIC_{pos}$  does not satisfy Markov equivalence as shown by the following example:

**Example 5.2.1.** *Let us re-consider the imprecise dataset  $\mathcal{D}$  in Table 5.1, the two Markov equivalent graphs in Figure 5.1 composed of two variables  $X_1$  and  $X_2$ .*

$X_1$	$X_2$	nb of occurrences
$x_{12}$	$x_{22}$	3
$x_{12}$	$x_{21}, x_{22}$	3
$x_{11}, x_{12}$	$x_{22}$	3
$x_{11}$	$x_{21}, x_{22}$	1

Table 5.1: Example of an imprecise dataset

For  $G_1$ ,  $\pi(x_{11}) = 0.4, \pi(x_{12}) = 0.9, \pi(x_{21}|x_{11}) = 0.25, \pi(x_{22}|x_{11}) = 1, \pi(x_{21}|x_{12}) = \frac{1}{3}$  and  $\pi(x_{22}|x_{12}) = 1$ . The dimension of  $G_1$  is 6.

$AIC_{pos}(G_1|\mathcal{D}) = 4 \log(0.4) + 9 \log(0.9) + \log(0.25) + 4 \log(1) + 3 \log(\frac{1}{3}) + 9 \log(1) - 6 = -10.03$ .

For  $G_2$ ,  $\pi(x_{21}) = 0.4, \pi(x_{22}) = 1, \pi(x_{11}|x_{21}) = 0.25, \pi(x_{12}|x_{21}) = 0.75, \pi(x_{12}|x_{22}) = 0.4$  and  $\pi(x_{11}|x_{22}) = 0.9$ . The dimension of  $G_2$  is 6.

$AIC_{pos}(G_2|\mathcal{D}) = 4 \log(0.4) + 10 \log(1) + \log(0.25) + 3 \log(0.75) + 4 \log(0.4) + 9 \log(0.9) - 6 = -11.10$ .

$AIC_{pos}$  will be combined with greedy search algorithm to learn possibilistic networks structure.

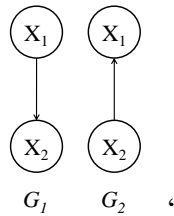


Figure 5.1: Example of two Markov equivalent graphs composed of two variables

### 5.3 Possibilistic adaptation of Greedy search algorithm

In this section, we propose an adaptation of greedy search initially proposed to learn Bayesian networks structure (cf. Section 2.3.3) in the possibilistic case. The underlying idea of greedy search is making the locally optimal choice at each step with the hope of finding a global optimum. Our choice could be justified by the fact that greedy search represents the most common solution for finding a high scoring network in the probabilistic case. Moreover, the decomposability property satisfied by our score  $AIC_{pos}$  allows us to efficiently evaluate the elementary operators (addition, reversing or deletion of an edge) performed by greedy search. In fact, it reduces the number of calculations by locally estimating the change in the score between two neighboring structures, instead of recalculating entirely to the new structure. Our possibilistic adaptation of greedy search is described by Algorithm 4.

---

#### Algorithm 4 Greedy search algorithm

---

**Require:** Initial graph  $G$ , dataset  $\mathcal{D}$

```

begin
  Continue  $\leftarrow$  True
   $AIC_{pos\_max} \leftarrow AIC_{pos}(G|\mathcal{D})$  repeat
    Generate  $Nbh(G)$  the neighborhood of  $G$ 
    % by deleting, reversing or adding an edge
    Compute  $AIC_{pos}(G'|\mathcal{D})$  for each graph  $G'$  in  $Nbh(G)$ 
     $G_{new} = \operatorname{argmax}_{G' \in Nbh(G)} AIC_{pos}(G'|\mathcal{D})$ 
    if  $AIC_{pos}(G_{new}|\mathcal{D}) > AIC_{pos\_max}$  then
      |  $AIC_{pos\_max} \leftarrow AIC_{pos}(G_{new}|\mathcal{D})$ 
      |  $G = AIC_{pos}(G_{new}|\mathcal{D})$ 
    else
      | Continue = False
    end
  until Continue;
end
Return  $G$ 

```

---

**Example 5.3.1.** Let the DAG in Figure 5.2 be the initial graph given to greedy search algorithm. In the first iteration, greedy search generates a neighborhood defined by the three operators addition (Insert), delete (Delete) and reversing (Reverse) edge. The resultant DAGs are presented in Figure 5.3. Then, greedy search computes obtained neighbors structures scores and pick the operation that leads to the the structure having the highest score. If we assume that the graph (f) (Iteration #1 of greedy search) is the best structure, we retain this structure and we perform the same steps to obtain the neighborhood composed of graphs in Figure 5.3 (Iteration #2 of greedy search). Now, if we assume that all the scores of obtained obtained graphs are inferior to the one of graph (f). Greedy search algorithm ends and returns graph (f). Otherwise, we re-perform the same steps until obtaining the structure with the highest score.



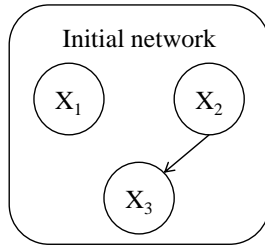


Figure 5.2: Initialization of greedy search

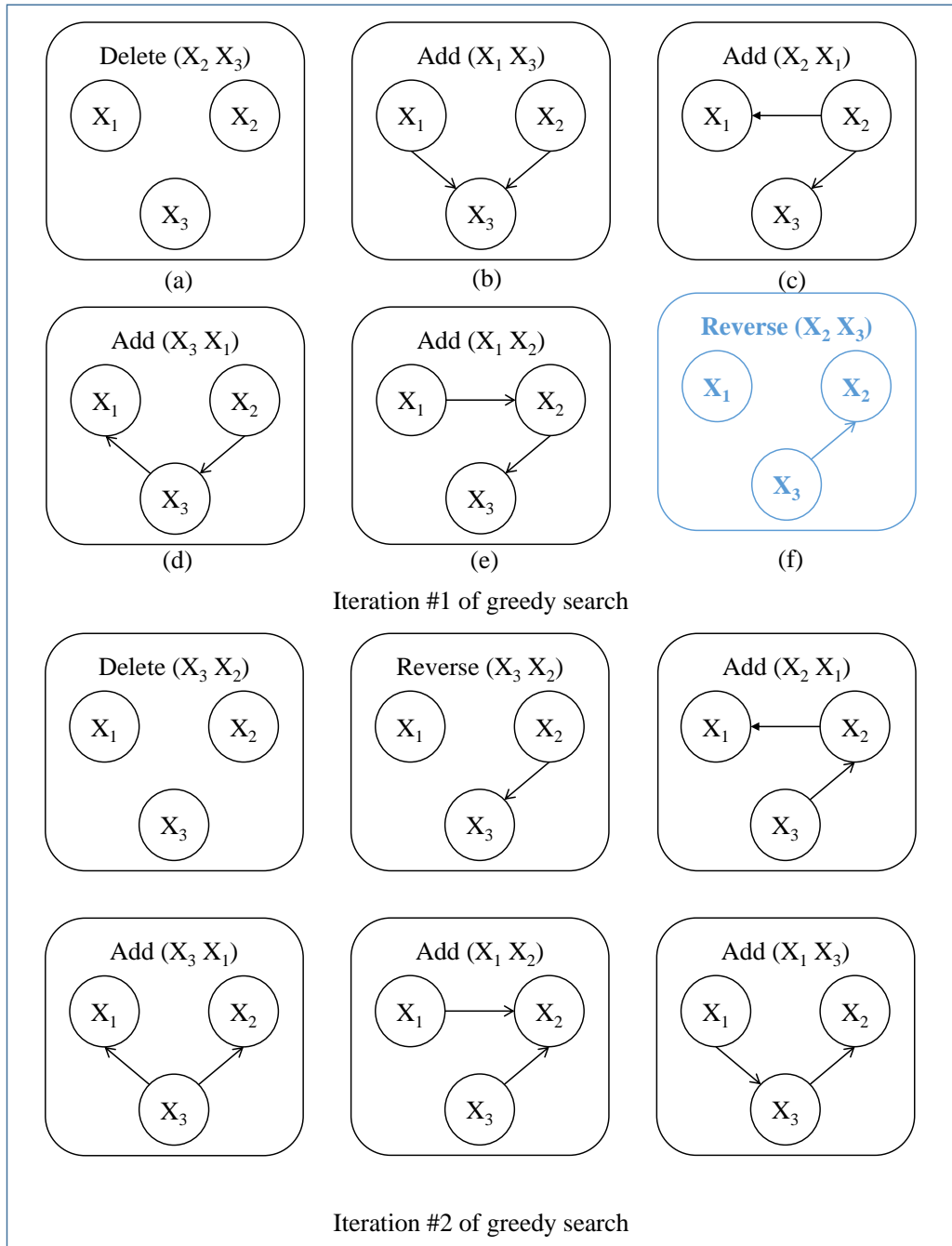


Figure 5.3: Illustration of greedy search

## 5.4 Experimental study

To evaluate our proposed possibilistic adaptation of greedy search, we use synthetic datasets containing 1000 imprecise observations generated using Algorithms 1, 2 and 3 from 8 randomly generated possibilistic networks composed of  $\{10, 20\}$  variables. We also vary the maximum number of parents between  $\{2, 4\}$  and the maximum number of variable domain cardinality between  $\{2, 5\}$  in generated networks. Then, we apply existing possibilistic learning structure algorithms which handle imprecise data, i.e. the possibilistic adaptation of k2 ( $\pi$ K2) and maximum weight spanning tree ( $\pi$ MWST) detailed in Chapter 2 (cf. Section 2.7) and our approach ( $\pi$ GS). In the current work,  $\pi$ K2 and  $\pi$ MWST are tested using two scores *possibilistic mutual information* ( $d_{mi}$ ) and *possibilistic  $\chi^2$  measure* ( $d_{\chi^2}$ ). The choice of these two scores is justified by the fact that Borgelt et al. showed that these two measures yield to good structures Borgelt et al. (2009). Note that  $\pi$ K2 treats variables in a predefined order, we generate 5 orders in each experiment and we retain the best structure.  $\pi$ GS is tested with three scores i.e.  $AIC_{pos}$  and the sum over variables in  $V$  of  $d_{\chi^2}$  and  $d_{mi}$  denoted respectively by  $\sum d_{\chi^2}$  and  $\sum d_{mi}$ . Then, we compute editing distance and Manhattan distance approximation between the learned and the initial possibilistic networks. Note that we can compute Manhattan distance approximation only when applying our proposed method i.e.  $\pi$ GS +  $AIC_{pos}$ . In fact, the other methods do not learn possibilistic networks but they return only networks structure without any numerical quantification. Tables 5.2 and 5.3 present the means of obtained results.

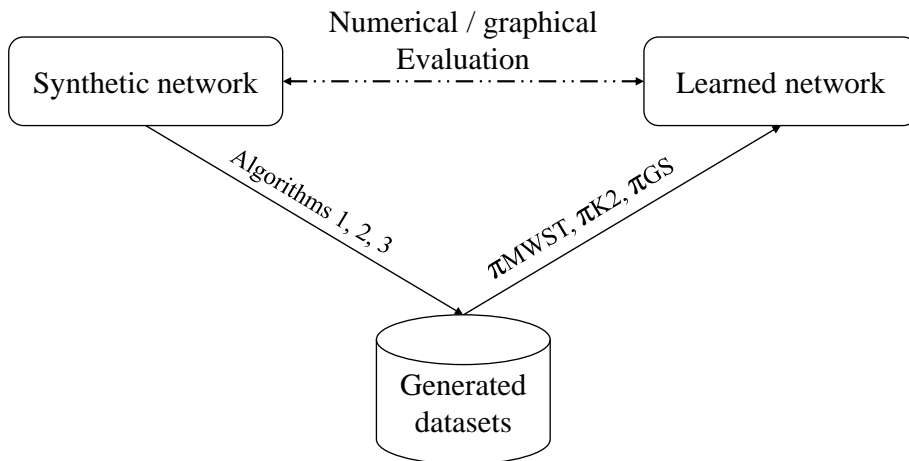


Figure 5.4: Proposed experimental protocol of possibilistic networks structure learning evaluation

Method \ n	Editing distance	
	10	20
$\pi$ GS + $AIC_{pos}$	19.77	31.55
$\pi$ GS + $\sum d_{\chi^2}$	28.83	51.66
$\pi$ GS + $\sum d_{mi}$	35.66	49.55
$\pi$ MWST + $d_{\chi^2}$	23.44	47.33
$\pi$ MWST + $d_{mi}$	22.77	47.55
$\pi$ K2 + $d_{\chi^2}$	27.44	42.22
$\pi$ K2 + $d_{mi}$	28.38	42.77

Table 5.2: Editing distance between initial and learned networks

Table 5.2 shows that  $\pi$ GS combined with  $AIC_{pos}$  is more interesting than  $\pi$ K2,  $\pi$ MWST and  $\pi$ GS combined with  $\sum d_{\chi^2}$  and  $\sum d_{mi}$  in term of editing distance. It is an expected result since greedy search

		Manhattan distance	
		n	
Method		10	20
		$\pi\text{GS} + AIC_{pos}$	0.05

Table 5.3: Manhattan distance between initial and learned networks

outperforms k2 and MWST in the probabilistic case. Table 5.3 shows that possibilistic networks learned using  $\pi\text{GS}$  combined with  $AIC_{pos}$  lead to good models in term of Manhattan distance between them and initial networks.

## 5.5 Conclusion

In this chapter, we have proposed the possibilistic counterpart of the probabilistic score  $AIC$ .  $AIC_{pos}$  is composed of two main components: likelihood function and complexity term computed via the dimension of the network. Then, we use this score to guide the greedy search algorithms trying to find the most simple structure that fits a datasets. We have shown that the proposed method lead to better structures regarding existing methods in term of editing distance. Such a result is preliminary and clearly deserves more investigations but is encouraging.



# Conclusion

Possibilistic networks present one of recent powerful frameworks for compactly representing uncertain and imprecise information in both ordinal and numerical settings. Contrarily to Bayesian networks, most of existing research endeavors devoted to possibilistic networks assume that they are elicited from experts. This may not be always obvious and deprives us of exploring the wealth of information compiled in existing datasets .

This is the core research problem of this Phd thesis in which we are interested by the learning of possibilistic networks (structure and parameters) from data.

Our contribution is twofold: The first one arose following our deep study of existing works relative to possibilistic networks that allows us to identify the importance of proposing formal learning algorithms. In fact, we have proposed a new possibilistic likelihood function to learn possibilistic networks parameters from imprecise datasets. Then, using this likelihood function, we have defined a new score named  $AIC_{pos}$  which represents the key component of greedy search algorithm proposed to learn possibilistic networks structure.

The second contribution concerns the evaluation of different proposed learning algorithms. In fact, in contrast with probabilistic models, research has not directly addressed this issue. So that we have proposed a global and experimental framework allowing the evaluation of any possibilistic learning algorithm. Then, we apply the developed evaluation protocol to our proposed learning algorithms to compare them with existing work addressing the same topic.

Experimental results show that given a DAG structure and a dataset, our parameters learning algorithm is efficient in term of quality and outperforms the existing work. However, concerning the evaluation of the proposed structure learning algorithm, obtained results are not clear-cut. In fact, first results show that we obtain better structures than those obtained using existing algorithms, but, such a conclusion deserves more investigations. First, editing distance penalizes so much the learned structures and it is obviously not sufficient simply to evaluate a structure learning algorithm. Second,  $AIC_{pos}$  does not satisfy the Markov equivalence property. This limitation deprives us to use graphical measures in an efficient way i.e. computed between CPDAGs (graph representative of a Markov equivalence class) instead of DAGs. Third, our proposed structure learning algorithm performs a greedy search which could be improved by constraining the latter as done in the probabilistic case by Max-Min Hill-Climbing algorithm, an hybrid method which is based on constraining a scoring search.

This conclusion leads us directly to multiple both short-and long-term perspectives that this work opens.

As short-term perspectives, we can distinguish direct improvements of our proposals. First, we aim to make a comparative study, between our proposed parameter learning method regarding the existing one, on a large number of benchmarks containing different types of imprecise data. This work is carried out under the project PEPS Fascido 2015 MAPPOS<sup>1</sup>. Second, we intend to propose a new conditional independence test with the aim of adapting Max-Min Hill-Climbing algorithm, to the possibilistic case. This method outperforms all the other existing learning algorithms proposed in the probabilistic case. to the possibilistic case. Third, we endeavor to distribute our toolbox gathering all methods proposed in this thesis. A part of our generated synthetic networks and datasets are already available<sup>1</sup> to researchers who are working in this area. As we lack possibilistic networks benchmarks, we believe that this can be a useful tool for evaluating

---

1. <https://sites.google.com/site/karimtabiasite/mappos>

new proposals related to the manipulation of possibilistic networks.

As long-term perspectives, we endeavor to test structure learning approaches proposed to learn probabilistic classifiers as augmented possibilistic classifiers and multinets, in the possibilistic framework. Learning these models from data has not been studied yet in the possibilistic frameworks. Another line of research concerns the proposition of an approximate method to perform information propagation in possibilistic networks using our proposed sampling method.

This thesis raises other open questions related for example to the nature of data to use if we try to learn qualitative possibilistic networks.



# Bibliography

- Adina, C. (2006). Possibilistic networks for uncertainty knowledge processing in student diagnosis. *Annals of Dunarea de Jos*, pages 69–73. 1, 30
- Akaike, H. (1970). Statistical predictor identification. *Annals of the Institute of Statistical Mathematics*, 22(1):203–217. 25
- Antonucci, A. (2011). The imprecise noisy-or gate. In *Information Fusion*, pages 1–7. 48
- Antonucci, A., Brühlmann, R., Piatti, A., and Zaffalon, M. (2009). Credal networks for military identification problems. *International Journal of Approximate Reasoning*, 50(4):666–679. 28
- Antonucci, A., Cattaneo, M. E., and Corani, G. (2012). Likelihood-based robust classification with bayesian networks. In *Advances in Computational Intelligence*, pages 491–500. 28
- Antonucci, A., Salvetti, A., and Zaffalon, M. (2004). Assessing debris flow hazard by credal nets. In *Soft Methodology and Random Information Systems*, pages 125–132. 28
- Augustin, T., Coolen, F. P., de Cooman, G., and Troffaes, M. C. (2014). *Introduction to imprecise probabilities*. John Wiley & Sons. 19
- Ayachi, R., Ben Amor, N., and Benferhat, S. (2014). A generic framework for a compilation-based inference in probabilistic and possibilistic networks. *Information Sciences*, 257:342–356. 1
- Ayachi, R., Ben Amor, N., Benferhat, S., and Haenni, R. (2010). Compiling possibilistic networks: Alternative approaches to possibilistic inference. In *Proceedings of the Twenty-Sixth Annual Conference on Uncertainty in Artificial Intelligence*, pages 40–47. 30
- Bachtobji, M. A., Ben Yaghlane, B., and Khaled, M. (2008). A new algorithm for mining frequent itemsets from evidential databases. In *Proceedings of Information Processing and Management of Uncertainty*, volume 8, pages 1535–1542. 27
- Beckmann, J., Gebhardt, J., Klawonn, F., and Kruse, R. (1994). Possibilistic inference and data fusion. In *Proceedings of Second European Congress on Fuzzy and Intelligent Technologies*, pages 46–47, Aachen. 30
- Ben Amor, N., Ben Yaghlane, A., and Khalfallah, F. (2009). Building possibilistic network parameters using elicited expert opinions. In *International Conference on Modeling, Simulation, and Applied Optimization*. 48
- Ben Amor, N. and Benferhat, S. (2005). Graphoid properties of qualitative possibilistic independence relations. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 13(01):59–96. 1, 32
- Ben Amor, N., Benferhat, S., and Mellouli, K. (2003). Anytime propagation algorithm for min-based possibilistic graphs. *Soft Computing*, 8(2):150–161. 1, 30
- Ben Hariz, N. and Ben Yaghlane, B. (2014). Learning parameters in directed evidential networks with conditional belief functions. In *International Conference on Belief Functions*, pages 294–303. Springer. 27
- Ben Hariz, N. and Ben Yaghlane, B. (2015). Learning structure in evidential networks from evidential databases. In *European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 301–311. Springer. 27



- Ben Slimen, Y., Ayachi, R., and Ben Amor, N. (2013). Probability-possibility transformation. In *Fuzzy Logic and Applications*, volume 8256, pages 122–130. 51
- Ben Yaghlane, B. and Mellouli, K. (1999). Updating directed belief networks. In *Proceedings of the 5th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 43–54. 26, 27
- Ben Yaghlane, B. and Mellouli, K. (2008). Inference in directed evidential networks based on the transferable belief model. *International Journal of Approximate Reasoning*, 48(2):399–418. 27
- Benavoli, A., Ristic, B., Farina, A., Oxenham, M., and Chisci, L. (2009). An application of evidential networks to threat assessment. *IEEE Transactions on Aerospace and Electronic Systems*, 45(2):620–639. 27
- Benferhat, S. and Smaoui, S. (2007). Hybrid possibilistic networks. *International Journal of Approximate Reasoning*, 44(3):224–243. 1, 30
- Benferhat, S. and Tabia, K. (2012). Inference in possibilistic network classifiers under uncertain observations. *Annals of Mathematics and Artificial Intelligence*, 64(2-3):269–309. 30
- Borgelt, C. and Gebhardt, J. (1997). Learning possibilistic networks with a global evaluation method. In *Proceedings of 5th European Congress on Intelligent Techniques and Soft Computing*, volume 2, pages 1034–1038. 31, 33
- Borgelt, C. and Kruse, R. (2003). Operations and evaluation measures for learning possibilistic graphical models. *Artificial Intelligence*, 148(1):385–418. 29, 31, 32
- Borgelt, C., Steinbrecher, M., and Kruse, R. (2009). *Graphical models: representations for learning, reasoning and data mining*, volume 704. Wiley. 1, 19, 28, 30, 31, 32, 37, 41, 47, 61, 65
- Bouckaert, R. R. (1993). Probabilistic network construction using the minimum description length principle. In *Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 41–48. 25
- Boughanem, M., Brini, A., and Dubois, D. (2009). Possibilistic networks for information retrieval. *International Journal of Approximate Reasoning*, 50(7):957–968. 1, 30
- Bouguelid, M. S. (2007). Contribution à l’application de la reconnaissance des formes et la théorie des possibilités au diagnostic adaptatif et prédictif des systèmes dynamiques. Thèse de doctorat. *Université de Reims Champagne-Ardenne, France*. 17
- Brown, L. E., Tsamardinos, I., and Aliferis, C. F. (2005). A comparison of novel and state-of-the-art polynomial bayesian network learning algorithms. In *Proceedings of the Twentieth National Conference on Artificial Intelligence*, pages 739–745. 25
- Caglioni, M., Moreno, D., Dubois, D., Prade, H., Tettamanzi, A. G., and et al. (2014). Mise en oeuvre pratique de réseaux possibilistes pour modéliser la spécialisation sociale dans les espaces métropolisés. *23 ème rencontres francophones sur la logique floue et ses applications*, pages 267–274. 1
- Cano, A., Gómez-Olmedo, M., and Moral, S. (2007). Credal nets with probabilities estimated with an extreme imprecise dirichlet model. In *Proceedings of the Fifth International Symposium on Imprecise Probability: Theories and Applications*, pages 57–66. 28
- Cano, A. and Moral, S. (2002). Using probability trees to compute marginals with imprecise probabilities. *International Journal of Approximate Reasoning*, 29(1):1–46. 28
- Chan, H. and Darwiche, A. (2005). A distance measure for bounding probabilistic belief change. *International Journal of Approximate Reasoning*, 38(2):149–174. 14
- Chanas, S. and Nowakowski, M. (1988). Single value simulation of fuzzy variable. *Fuzzy Sets and Systems*, 25(1):43–57. 13, 14
- Chavira, M. and Darwiche, A. (2007). Compiling bayesian networks using variable elimination. In *International Joint Conference on Artificial Intelligence*, pages 2443–2449. 24

- Chebil, W., Soualmia, L. F., Omri, M. N., and Darmoni, S. J. (2015). Indexing biomedical documents with a possibilistic network. *Journal of the Association for Information Science and Technology*, 1, 30
- Chernoff, H. and Lehmann, E. L. (1954). The use of maximum likelihood estimates in  $\chi^2$  tests for goodness of fit. *The Annals of Mathematical Statistics*, 25(3):579–586. 24
- Chow, C. and Liu, C. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467. 25, 26
- Cooper, G. F. (1990). The computational complexity of probabilistic inference using bayesian belief networks. *Artificial intelligence*, 42(2):393–405. 23
- Cooper, G. F. and Herskovits, E. (1992). A bayesian method for the induction of probabilistic networks from data. *Machine learning*, 9(4):309–347. 25
- Corani, G., Giusti, A., Migliore, D., and Schmidhuber, J. (2010). Robust texture recognition using credal classifiers. In *Proceedings of the British Machine Vision Conference*, pages 1–10. 28
- Corani, G. and Zaffalon, M. (2008). Learning reliable classifiers from small or incomplete data sets: the naive credal classifier 2. *Journal of Machine Learning Research*, 9:581–621. 58
- Cowell, R. G. (2003). Finex: a probabilistic expert system for forensic identification. *Forensic Science International*, 134(2):196–206. 24
- Cozman, F. G. (2000). Credal networks. *Artificial intelligence*, 120(2):199–233. 27, 28
- Cozman, F. G. (2001). Javabayes-bayesian networks in java. <http://www.cs.cmu.edu/~javabayes>. 28
- Cozman, F. G. (2014). Learning imprecise probability models: Conceptual and practical challenges. *International Journal of Approximate Reasoning*, 55(7):1594–1596. 28
- da Rocha, J. C. F. and Cozman, F. G. (2002). Inference with separately specified sets of probabilities in credal networks. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 430–437. 28
- da Rocha, J. C. F. and Cozman, F. G. (2003). Inference in credal networks with branch-and-bound algorithms. In *International Symposium on Imprecise Probabilities and Their Applications*. 28
- da Rocha, J. C. F., Cozman, F. G., and de Campos, C. P. (2002). Inference in polytrees with sets of probabilities. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 217–224. 28
- Dagum, P. and Horvitz, E. (1993). A bayesian analysis of simulation algorithms for inference in belief networks. *Networks*, 23(5):499–516. 24
- Darwiche, A. and Goldszmidt, M. (1994). On the relation between kappa calculus and probabilistic reasoning. In *Proceedings of the Tenth international conference on Uncertainty in artificial intelligence*, pages 145–153. 17
- De Campos, C. P., Zhang, L., Tong, Y., and Ji, Q. (2009). Semi-qualitative probabilistic networks in computer vision problems. *Journal of Statistical Theory and Practice*, 3(1):197–210. 28
- De Campos, L. M. and Huete, J. F. (2001). Measurement of possibility distributions. *International Journal of General System*, 30(3):309–346. 19, 20
- Dempster, A. P. (1967). Upper and lower probabilities induced by a multivalued mapping. *The Annals of Mathematical Statistics*, pages 325–339. 8, 19
- Destercke, S., Dubois, D., and Chojnacki, E. (2007). Transforming probability intervals into other uncertainty models. In *In Proceedings of European Society for Fuzzy Logic and Technology*, volume 2, pages 367–373. 19, 20
- Diez, F. J., Mira, J., Iturralde, E., and Zubillaga, S. (1997). Diaval, a bayesian expert system for echocardiography. *Artificial Intelligence in Medicine*, 10(1):59–73. 24

- Dubois, D. (2006). Possibility theory and statistical reasoning. *Computational Statistics & Data Analysis*, 51(1):47–69. [8](#), [17](#), [20](#), [50](#)
- Dubois, D., Foulloy, L., Mauris, G., and Prade, H. (2004). Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. *Reliable Computing*, 10(4):273–297. [17](#), [33](#)
- Dubois, D., Fusco, G., Prade, H., and Tettamanzi, A. (2015). Uncertain logical gates in possibilistic networks: An application to human geography. In *International Conference on Scalable Uncertainty Management*, pages 249–263. [30](#)
- Dubois, D. and Prade, H. (1988). *Possibility theory*. Springer. [8](#)
- Dubois, D. and Prade, H. (1990). Consonant approximations of belief functions. *International Journal of Approximate Reasoning*, 4(5):419–449. [54](#)
- Dubois, D. and Prade, H. (1998). Possibility theory: qualitative and quantitative aspects. In *Quantified Representation of Uncertainty and Imprecision*, volume 1, pages 169–226. Springer. [1](#), [8](#), [10](#)
- Dubois, D. and Prade, H. (2000). Possibility theory in information fusion. In *Information Fusion*, volume 1, pages 6–19. [8](#)
- Dubois, D. and Prade, H. (2006). Représentations formelles de l’incertain et de l’imprécis. *Concepts et méthodes pour l’aide à la décision - outils de modélisation*, 1:99–137. [22](#)
- Dubois, D. and Prade, H. (2016). Practical methods for constructing possibility distributions. *International Journal of Intelligent Systems*, 31(3):215–239. [17](#), [18](#)
- Dubois, D., Prade, H., and Sandri, S. (1993). On possibility/probability transformations. In *Fuzzy Logic*, volume 12, pages 103–112. [17](#), [33](#)
- Dubois, D., Prade, H., and Smets, P. (1996). Representing partial ignorance. *Systems, Man and Cybernetics, Part A: IEEE Transactions on Systems and Humans*, 26(3):361–377. [8](#)
- Dunford, N., Schwartz, J. T., Bade, W. G., and Bartle, R. G. (1971). *Linear operators*. Wiley-interscience New York. [15](#)
- Fielding, A. H. and Bell, J. F. (1997). A review of methods for the assessment of prediction errors in conservation presence/absence models. *Environmental conservation*, 24(01):38–49. [42](#)
- Fonck, P. (1992). Propagating uncertainty in a directed acyclic graph. In *Proceedings of the fourth Information Processing and Management of Uncertainty Conference*, volume 92, pages 17–20. [1](#), [28](#)
- Fonck, P. (1994). Conditional independence in possibility theory. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, pages 221–226. [30](#)
- Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163. [58](#)
- Gebhardt, J. and Kruse, R. (1997). Background and perspectives of possibilistic graphical models. In *Qualitative and Quantitative Practical Reasoning*, volume 1244, pages 108–121. Springer. [30](#)
- Giang, P. H. and Shenoy, P. P. (1999). On transformations between probability and Spohnian disbelief functions. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 236–244. [17](#)
- Guyonnet, D., Bourguine, B., Dubois, D., Fargier, H., Côme, B., and Chilès, J.-P. (2003). Hybrid approach for addressing uncertainty in risk assessments. *Journal of Environmental Engineering*, 129(1):68–78. [13](#)
- Haddad, M., Ben Amor, N., and Leray, P. (2013). Imputation of possibilistic data for structural learning of directed acyclic graphs. In *Fuzzy Logic and Applications*, pages 68–76. [3](#)
- Haddad, M., Ben Amor, N., and Leray, P. (2015a). Apprentissage des réseaux possibilistes à partir de données. *Revue d’Intelligence Artificielle*, 992:229–252. [3](#)

- Haddad, M., Leray, P., and Amor, N. B. (2015b). Evaluating product-based possibilistic networks learning algorithms. In *Proceedings of Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 312–321. [3](#)
- Haddad, M., Leray, P., and Amor, N. B. (2015c). learning possibilistic networks from data : a survey. In *16th World Congress of the International Fuzzy Systems Association and the 9th Conference of the European Society for Fuzzy Logic and Technology*, pages 194–201. [3](#), [42](#)
- Haddad, M., Leray, P., Levray, A., and Tabia, K. (2016). Possibilistic networks parameter learning: Preliminary empirical comparison. In *8èmes journées francophones de réseaux bayésiens (JFRB 2016)*. [3](#), [31](#)
- Heckerman, D. (1998). A tutorial on learning with bayesian networks. In *Learning in graphical models*, pages 301–354. [24](#)
- Henrion, M. (1986a). Propagating uncertainty in bayesian networks by probabilistic logic sampling. In *Uncertainty in Artificial Intelligence 2 Annual Conference on Uncertainty in Artificial Intelligence*, pages 149–163, Amsterdam, NL. Elsevier Science. [24](#)
- Henrion, M. (1986b). Propagating uncertainty in bayesian networks by probabilistic logic sampling. In *Uncertainty in Artificial Intelligence*, pages 149–164. [37](#)
- Henrion, M., Provan, G., Del Favero, B., and Sanders, G. (1994). An experimental comparison of numerical and qualitative probabilistic reasoning. In *Proceedings of the Tenth international conference on Uncertainty in artificial intelligence*, pages 319–326. [17](#)
- Higashi, M. and Klir, G. J. (1983). Fuzzy sets as a basis for a theory of possibility. *International Journal of General Systems*, 9:103–115. [14](#)
- Hisdal, E. (1978). Conditional possibilities independence and non interaction. *Fuzzy Sets and Systems*, 1:283–297. [11](#)
- Hou, Y. and Yang, B. (2010). Probability-possibility transformation for small sample size data. In *Fuzzy Systems and Knowledge Discovery*, volume 4, pages 1720–1724. [19](#), [20](#)
- Jenhani, I., Ben Amor, N., Elouedi, Z., Benferhat, S., and Mellouli, K. (2007). Information affinity: a new similarity measure for possibilistic uncertain information. In *Proceedings of Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 840–852. [14](#), [16](#)
- Jensen, F. V. (1996). *An introduction to Bayesian networks*, volume 74. UCL press London. [1](#), [22](#), [24](#)
- Joslyn, C. (1991). Towards an empirical semantics of possibility through maximum uncertainty. In *Fourth World Congress of the International Fuzzy Systems Association: Artificial Intelligence*, pages 86–89. [12](#)
- Joslyn, C. (1997). Measurement of possibilistic histograms from interval data. *International Journal Of General System*, 26(1-2):9–33. [19](#)
- Judea Pearl, T. V. (1991). Equivalence and synthesis of causal models. In *Proceedings of Sixth Conference on Uncertainty in Artificial Intelligence*, pages 220–227. [23](#)
- Kim, J. H. and Pearl, J. (1983). A computational model for causal and diagnostic reasoning in inference systems. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, volume 1, pages 190–193. [23](#)
- Klir, G. J. and Mariano, M. (1987). On the uniqueness of possibilistic measure of uncertainty and information. *Fuzzy Sets and Systems*, 24(2):197–219. [10](#)
- Klir, G. J. and Parviz, B. (1992). Probability-possibility transformations: a comparison. *International Journal of General System*, 21(3):291–310. [17](#), [33](#)
- Korb, K. B. and Nicholson, A. E. (2010). *Bayesian artificial intelligence*. CRC press. [24](#)
- Kruse, R. and Borgelt, C. (1995). Learning probabilistic and possibilistic networks : Theory and applications. In *Eighth European Conference on Machine Learning*, pages 3–16. [30](#)

- Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86. [14](#), [41](#), [42](#)
- Laâmari, W. and Ben Yaghlane, B. (2014). Reasoning in singly-connected directed evidential networks with conditional beliefs. In *Proceedings of the 8th Hellenic Conference on Artificial Intelligence: Methods and Applications*, pages 221–236. [27](#)
- Lauritzen, S. L. (1995). The EM algorithm for graphical association models with missing data. *Computational Statistics & Data Analysis*, 19(2):191–201. [24](#)
- Lauritzen, S. L. and Spiegelhalter, D. J. (1988). Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 157–224. [24](#), [37](#)
- Li, Z. and d’Ambrosio, B. (1994). Efficient inference in bayes networks as a combinatorial optimization problem. *International Journal of Approximate Reasoning*, 11(1):55–81. [24](#)
- Masegosa, A. R. and Moral, S. (2014a). Imprecise probability models for learning multinomial distributions from data. applications to learning credal networks. *International Journal Approximate Reasoning*, 55(7):1548–1569. [28](#)
- Masegosa, A. R. and Moral, S. (2014b). Imprecise probability models for learning multinomial distributions from data. applications to learning credal networks. *International Journal of Approximate Reasoning*, 55(7):1548–1569. [28](#)
- Masson, M.-H. and Dencœux, T. (2006). Inferring a possibility distribution from empirical data. *Fuzzy Sets and Systems*, 157(3):319–340. [19](#), [20](#)
- Mouchaweh, M., Bouguelid, M., Billaudel, P., and Riera, B. (2006). Variable probability-possibility transformation. *Twenty fifth European Annual Conference on Human Decision Making and Manual Control*, pages 417–428. [17](#), [18](#)
- Murphy, K. (2001). An introduction to graphical models. *Technical report*, pages 1–19. [1](#)
- Murphy, K. (2014). Bayesian network toolbox (bnt),(version 1.0. 7) computer software. [24](#)
- Oatley, G. C. and Ewart, B. W. (2003). Crimes analysis software: ‘pins in maps’, clustering and bayes net prediction. *Expert Systems with Applications*, 25(4):569–588. [24](#)
- Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann. [1](#), [21](#), [22](#), [23](#), [48](#)
- Pearl, J. and Verma, T. S. (1995). A theory of inferred causation. *Studies in Logic and the Foundations of Mathematics*, 134:789–811. [25](#)
- Pourret, O., Naïm, P., and Marcot, B. (2008). *Bayesian networks: a practical guide to applications*, volume 73. John Wiley & Sons. [22](#), [24](#)
- Riesz, F. (1910). Untersuchungen über systeme integrierbarer funktionen. *Mathematische Annalen*, 69(4):449–497. [15](#)
- Russell, S., Binder, J., Koller, D., and Kanazawa, K. (1995). Local learning in probabilistic networks with hidden variables. In *International Joint Conference on Artificial Intelligence*, volume 95, pages 1146–1152. [37](#)
- Sabbadin, R. (2001). Towards possibilistic reinforcement learning algorithms. In *The Tenth IEEE International Conference on Fuzzy Systems*, volume 1, pages 404–407. IEEE. [17](#)
- Sangüesa, R., Cabós, J., and Cortes, U. (1998). Possibilistic conditional independence: A similarity-based measure and its application to causal network learning. *International Journal of Approximate Reasoning*, 18(1):145–167. [1](#), [31](#), [32](#), [33](#), [37](#), [47](#)
- Sangüesa, R. and Cortés, U. (2000). Prior knowledge for learning networks in non-probabilistic settings. *International Journal of Approximate Reasoning*, 24(1):103–120. [15](#)

- Sangüesa, R., Cortés, U., and Gisolfi, A. (1998b). A parallel algorithm for building possibilistic causal networks. *International Journal of Approximate Reasoning*, 18(3):251–270. 33
- Shackle, G. L. S. (1969). *Decision, order, and time in human affairs*. Cambridge University Press. 8
- Shafer, G. (1976). *A mathematical theory of evidence*, volume 1. Princeton university press Princeton. 8, 18, 19, 22
- Shapiro, L. G. and Haralick, R. M. (1985). A metric for comparing relational descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(1):90–94. 42
- Shenoy, P. P. (1997). Binary join trees for computing marginals in the shenoy-shafer architecture. *International Journal of approximate reasoning*, 17(2):239–263. 27
- Simon, C. and Weber, P. (2009). Evidential networks for reliability analysis and performance evaluation of systems with imprecise knowledge. *IEEE Transactions on Reliability*, 58(1):69–87. 27
- Smets, P. (1988). Belief functions. In *Non Standard Logics for Automated Reasoning*, pages 253–286. 18
- Smets, P. (1989). Constructing the pignistic probability function in a context of uncertainty. In *UAI*, volume 89, pages 29–40. 17
- Smets, P. (1993). Belief functions: the disjunctive rule of combination and the generalized bayesian theorem. *International Journal of Approximate Reasoning*, 9(1):1–35. 26
- Smets, P. and Kennes, R. (1994). The transferable belief model. *Artificial intelligence*, 66(2):191–234. 48
- Sotelo, M. A., Bergasa, L. M., Flores, R., Ocana, M., Doussin, M.-H., Magdalena, L., Kalwa, J., Madsen, A. L., Perrier, M., Roland, D., et al. (2003). Advocate ii: Advanced on-board diagnosis and control of autonomous systems ii. In *Computer Aided Systems Theory*, pages 302–313. 24
- Spirtes, P., Glymour, C. N., and Scheines, R. (2000). *Causation, prediction, and search*. MIT press. 25
- Tsamardinos, I., Brown, L. E., and Aliferis, C. F. (2006). The max-min hill-climbing bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78. 26
- Xiang, Y. and Miller, T. (1999). A well-behaved algorithm for simulating dependence structures of Bayesian networks. *International Journal of Applied Mathematics*, 1:923–932. 38
- Xu, H. (1997). Valuation-based systems for decision analysis using belief functions. *Decision Support Systems*, 20(2):165–184. 27
- Xu, H. and Smets, P. (1994). Evidential reasoning with conditional belief functions. In *Proceedings of the Tenth international conference on Uncertainty in artificial intelligence*, pages 598–605. 26, 27
- Yager, R. R. (1992). On the specificity of a possibility distribution. *Fuzzy Sets and Systems*, 50(3):279–292. 9
- Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8:338–353. 8
- Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning. *Journal of Information Science*, 8:199–249. 13
- Zadeh, L. A. (1978). Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100:9–34. 1, 8, 22
- Zaffalon, M. and Hutter, M. (2005). Robust inference of trees. *Annals of Mathematics and Artificial Intelligence*, 45(1-2):215–239. 28
- Zhou, K., Martin, A., and Pan, Q. (2016). The belief noisy-or model applied to network reliability analysis. preprint *arXiv:1606.01116*. 48





# Thèse de Doctorat

**Maroua HADDAD**

**Apprentissage de modèles graphiques possibilistes à partir de données**

**Learning possibilistic graphical models from data**

## Résumé

Ce travail s'intègre dans le cadre de l'apprentissage automatique des réseaux possibilistes, la contrepartie possibiliste des réseaux bayésiens qui représentent une combinaison intéressante entre la théorie des possibilités et les modèles graphiques. Cette thèse présente deux contributions majeures. La première contribution consiste à proposer une stratégie de validation pour les algorithmes d'apprentissage des réseaux possibilistes. Cette stratégie propose un processus d'échantillonnage permettant de générer des ensembles de données imprécises à partir de ces modèles et deux nouvelles mesures d'évaluation. Notre deuxième contribution consiste à proposer une approche globale pour l'apprentissage des paramètres et de la structure des réseaux possibilistes. Nous proposons une fonction de vraisemblance possibiliste pour apprendre les paramètres des réseaux possibilistes et définir une nouvelle fonction de score pour apprendre la structure de ces modèles. Une étude expérimentale détaillée montrant la faisabilité et l'efficacité des méthodes proposées a été aussi proposée.

## Mots clés

Réseaux possibilistes, données imprécises, apprentissage automatique, théorie des possibilités, échantillonnage des réseaux possibilistes, distance possibiliste.

## Abstract

This work fits within the framework of learning possibilistic networks, the possibilistic counterpart of Bayesian networks, which represent an interesting combination between possibility theory and graphical models. This thesis presents two major contributions. The first one consists on proposing a validation strategy for possibilistic networks learning algorithms. This strategy proposes a sampling process to generate imprecise datasets from these models and two new evaluation measures. Our second contribution consists on proposing a global approach to learn the structure and the parameters of possibilistic networks. We propose a possibilistic likelihood function to learn possibilistic networks parameters and to define a new score function used to learn the structure of these models. A detailed experimental study showing the feasibility and the efficiency of the proposed methods has been also proposed.

## Key Words

Possibilistic networks, imprecise data, machine learning, possibility theory, sampling possibilistic networks, possibilistic distance.