Available online at www.sciencedirect.com
ScienceDirect
Applied Soft Computing 11 (2011) 1511–1528
Applied Soft Computing
www.elsevier.com/locate/asoc

Probabilistic graphical models in artificial intelligence

P. Larrañaga [a], S. Moral [b,*]

[a] Artificial Intelligence Department, Technical University of Madrid, Madrid, Spain
[b] Computer Science and Artificial Intelligence Department, University of Granada, Granada, Spain

Received 19 April 2007; received in revised form 14 January 2008; accepted 15 January 2008
Available online 19 January 2008

Abstract

In this paper, we review the role of probabilistic graphical models in artificial intelligence. We start by giving an account of the early years when there was important controversy about the suitability of probability for intelligent systems. We then discuss the main milestones for the foundations of graphical models starting with Pearl's pioneering work. Some of the main techniques for problem solving (abduction, classification, and decision-making) are briefly explained. Finally, we propose some important challenges for future research and highlight relevant applications (forensic reasoning, genomics and the use of graphical models as a general optimization tool).

Keywords: Probability; Uncertain reasoning; Bayesian networks; Gaussian networks; Credal networks; Factor graphs; Kikuchi approximations; Decision-making; Classification; Optimization; Metaheuristics; Genomic; Forensic

1. Introduction

Although probabilistic methods are now fundamental for building intelligent systems, this has not always been the case. In the early years of artificial intelligence (years characterized by excessive enthusiasm), probability was not considered to be a basic tool. Researchers were more concerned with developing new general purpose procedures, many of them based only on intuition, rather than looking at existing well-established fields such as probability and statistics.

This period came to an end, however, and programs that had worked in simple examples were proved to be completely unsuitable for solving more complex situations. As a result, the focus changed and it became clear that in the long term it was preferable to invest effort in developing well-founded theories based on the existing body of knowledge of general science. Probability then began to play a fundamental role and had to be adapted to the new problems to which it was to be applied, something which was achieved by following a deep, sound mathematical methodology instead of attempting to build programs to solve particular examples.

In general, *probabilistic graphical models* comprise any model that uses the language of graphs to facilitate the representation and resolution of complex problems that use probability as representation of uncertainty. The most important particular case is the *Bayesian network model* in which a directed acyclic graph is used to represent a joint probability distribution about several variables.

This paper attempts to explain the reasons for the present success of probabilistic graphical models, but with special emphasis in Bayesian networks. We offer a summary of the early years of probability when there was important controversy about the suitability of probability (Section 2). We then summarize the main contributions that made probability a powerful tool for modeling complex real systems, for which the appearance of Pearl's book *Probabilistic Reasoning in Intelligent Systems* [1] was fundamental, in addition to other very important contributions that are highlighted in Section 3. Although Bayesian networks were originally designed for computing conditional probabilities, they soon proved useful for different tasks and in Section 4 we show some of these: clustering, abductive reasoning, classification, and decision-making. The applications of graphical models are diverse. In this paper we examine some of the ones which we believe to be really innovative and significant. First, in Section 5, we examine the application of graphical models as a general optimization approach for hard problems. This is performed by

* Corresponding author. Tel.: +34 958242819.
E-mail address: smc@decsai.ugr.es (S. Moral).

1568-4946/$ – see front matter
doi:10.1016/j.asoc.2008.01.003

two main methodologies: estimation of distribution algorithms and inference-based methods. In Section 6 we discuss two of the most challenging applications of graphical models: forensic reasoning and genomics.

Although the field is currently being developed in many directions, and it would therefore be difficult to list even the most important ones, we have, however, selected various generalizations or modifications of the initial model enabling the scope of graphical models to be expanded. In Section 7 we briefly describe the state of the art of Markov random fields, factor graphs, Kikuchi approximations, and credal networks. Finally, in Section 8 we present our conclusions.

We do not intend for this paper to be a complete review of probabilistic graphical models and their present research problems, as this would be impossible given the space limitations of a journal paper. Instead, we will indicate the main contributions of the early years, and will describe some present research topics which we have selected by taking into account both our knowledge and experience on the one hand, and our opinion about their relevance on the other.

2. The early years of probability in artificial intelligence

Initially, probability was not seen as an important tool for artificial intelligence and at the Darmouth conference [2] probability was hardly mentioned at all. Only *randomness* was considered to play a role in connection with creativity:

> ''A fairly attractive and yet clearly incomplete conjecture is that the difference between creative thinking and unimaginative competent thinking lies in the injection of certain randomness. The randomness must be guided by intuition to be efficient. In other words, the educated guess or the hunch include controlled randomness in otherwise orderly thinking.''

This can be considered to have been accomplished with the existing random search algorithms including the case of evolutionary computation algorithms.

It was during the construction of the expert system MYCIN [3] that it became apparent that some formalism was necessary for representing and reasoning with uncertainty [4]:

> ''It seemed clear that we needed to handle probabilistic statements in our rules and to develop a mechanism for gathering evidence for and against a hypothesis when two or more relevant rules were successfully executed.''

Although it was recognized that probability theory provides useful procedures for handling uncertainty, it was finally ruled out mainly because it was assumed to require a complete specification of conditional statement parameters which were rarely available [5]:

> ''Although conditional probability provides useful results in areas of medical decision making such as those we have mentioned, vast portions of medical experience suffer from having such little data and so much imperfect knowledge that a rigorous probabilistic analysis, the ideal standard by which to judge the rationality of a physician decision, is not possible.''

For this reason, a new formalism was created, the so-called *certainty factors*, with the aim of being able to reason with available uncertain rules, without being subject to the requirements of the Theory of Probability. This example was followed by other expert systems and some of these created their own method for handling uncertainty with rules and methods mainly based on intuition. This was the case of INTERNIST-1 [6]. Some of these suffered from important inconsistencies, mainly due to the non-distinction between absolute and updated beliefs (beliefs that are obtained under certain given observations). This was also the case of INFERNO [7]. This system was inspired by the theory of probability, but a couple of values were assigned to each event, and rules were interpreted as inequalities in conditional probabilities. However, the wrong use of information contexts can produce important inconsistencies and the system must spend a lot of effort trying to remove them. It may even be that *a propagation chain with ever increasing bounds* arises.

Although some of these systems behaved extremely well, this was due to a careful design of the knowledge base, taking care to avoid duplicities, and bearing in mind the posterior use of the system. There were also important restrictions in the way the knowledge could be used (for example in MYCIN rules could be used in one direction only). However, the blind application of certainty factors to other domains without validating their performance proved to be dangerous and subject to important flaws in the reasoning process [8].

Since the early 1980s, a group of researchers have been working on probabilistic models. Starting from the rich accumulated knowledge in statistical science, they built procedures adapted to artificial intelligence problems. Although some work had previously been carried out, we should mention the paper *Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach*[1] [9] as the origin of interest in probabilistic models in artificial intelligence. This paper demonstrated that probability could provide a sound, flexible and efficient procedure for solving complex problems of reasoning with uncertainty.

At the same time, certain non-probabilistic theories of representing imperfect knowledge were developed, among which we can distinguish fuzzy sets and possibility theory [10–12] and the theory of belief functions [13,14]. These new models provide methods for representing vague or ill-defined information such as for example ''*The temperature is high*''. They also provided the possibility of representing ignorance. Consequently, we are not forced to give a probability distribution for each variable in the problem, and we can encode situations in which the quality of information is poor,

[1] In 2000, this paper received the *AAAI Classic Paper Award*.

although we can also represent precise probability distributions with belief functions if they are available.

The coexistence of alternative models gave rise to a sour debate about the appropriateness of the different theories. The positions were hard and in some cases excluded the possibility of alternative positions. This strong view was more common in the field of probability theory, perhaps under the influence of papers showing that probability was the only possible model for representing and reasoning with uncertainty [15,16]. In this line, we can cite Cheesman [17]:

> "The aim of this paper is to show that fuzzy sets (or fuzzy logic) are unnecessary for representing and reasoning about uncertainty (including vagueness) – probability theory is all that is required."

In addition, the objective of the most recent paper by Elkan [18] is to prove that fuzzy logic is simply wrong ("*fuzzy logic cannot be used for general reasoning under uncertainty with deep knowledge*") and its success in certain applications is due to their simplicity and other external factors ("*The fact that using fuzzy logic is correlated with success does not entail that using fuzzy logic causes success*").

It is true that new theories capture a lot of attention, and some people start working on them with the idea of making fast progress (something which is more difficult in more established fields), sometimes without a deep understanding of the new models. This resulted in developments or applications which might be considered incorrect. Nevertheless, at the same time, we believe that some of these opposing positions were really maximalist and were also due to a superficial view of the condemned theories. In any case, they were very interesting debates which helped to clarify the meaning and scope of application of the models. A report of these discussions can be found in the first edited volume with selected papers of the *Workshop on Probability and Uncertainty in AI* that was held in 1985 in Los Angeles in conjunction with the IJCAI conference [19]. The largest section of the book was devoted to "*Explanation or Critique of Current Approaches to Uncertainty*".

With time, we can say that the intensity of this debate has fallen to very low levels. Nowadays, there is a tendency to evaluate models in terms of their empirical performance in solving practical problems and so each theory has found its own applications where there is a good balance between performance, efficiency, and simplicity of construction.

The struggle for establishing probability as a basic tool in artificial intelligence went beyond discussion with the proponents of other numerical uncertainty theories. It was also necessary to convince mainstream artificial intelligence (much more logical-qualitative oriented) that this quantitative approach could finally provide very competent solutions to basic artificial intelligence problems. The series of conferences on *Uncertainty in Artificial Intelligence* constituted the main forum for presenting and spreading the probabilistic approach. The first conference in 1985 was in Henrion's words: "*something of a fringe group as far as mainstream artificial intelligence research was concerned*". [20, Preface]. The title of the final panel of the Third Conference in Artificial Intelligence is also symptomatic: *Why does mainstream artificial intelligence ignore uncertainty research?* We think that there are several reasons for this: first, using probability and statistics was seen by some as a loss of generality with respect to using more universal logical languages; secondly, the human model was not followed, because initially, artificial intelligence attempted to imitate human reasoning, and it was clear that we do not think by performing thousands of numerical computations; finally, there was a tendency from the years of the great expectations to reinvent everything, without relying on the already rich existing scientific tradition. Things are different now, and since the 1990s there has been much interest in probability for dealing with uncertainty in artificial intelligence. One example of this can be seen in natural language processing where the statistical-based hidden Markov models have become the most commonly used approach [21].

3. Graphical models

The use of probability in artificial intelligence has been impelled by the development of graphical models which have become widely known and accepted after the excellent book: *Probabilistic Reasoning in Intelligent Systems* [22]. These models have provided a language for representing complex situations, and has also finally been the basis for efficient computations and the estimation of the necessary parameters from sets of observations. The use of graphs to describe statistical models has a long tradition. Perhaps the most relevant antecedents, however, can be found in the following contributions:

- The work on contingency tables by Darroch et al. [23], where undirected graphs were used to represent the relationships between a set of discrete variables and by Wermuth and Lauritzen [24] in which directed acyclic graphs were used.
- The study of conditional independence in probability by Dawid [25] which is the basis for giving meaning to the graph representation.
- The introduction of influence diagrams [26] which used directed acyclic graphs to represent complex decision problems (but without computing over these graphs).
- The 'peeling' algorithm developed by Cannings et al. [27] in the context of pedigree computations that was very similar to the propagation algorithms that are currently used in graphical models.

These developments occurred in the field of classical statistics and probability. The introduction of graphical models in artificial intelligence was due to Pearl [9]. In this paper, he was motivated by typical artificial intelligence problems (expert systems, speech recognition, language understanding, etc.) and he showed that if the knowledge about the problem was structured as a tree (in which the numerical information was given as conditional probabilities), this representation could also be used for reasoning, using rules in both directions

(prediction and diagnosis) without falling into circular inferences (*A* increases the belief in *B*, and the increase in *B* gives rise to more belief in *A*). The conclusions of this paper were simple but premonitory:

> "The paper demonstrates that the centuries-old Bayes formula still retains its potency for serving as the basic belief revising rule in large, multi-hypotheses, inference systems. It is proposed, therefore, as a standard point of departure for more sophisticated models of belief maintenance and inexact reasoning."

The term *Bayesian network* was introduced in [28] which showed that the graphical representation was not only useful as a computational device but also as a procedure for specifying complex problems by means of a family of low-order simpler relationships between small sets of variables. A Bayesian network consisted of two parts: a qualitative part (expressing the relationships between the variables with semantics based on the concept of conditional independence) and a quantitative part (the numerical values of a set of conditional probability distributions). The qualitative part, however, was considered to be more basic and primitive. In order to model a problem, we first need to provide the structure and then the numbers: "*Evidently, the notion of conditional independence is more basic than the numerical values attached to probability judgments, contrary to the picture painted in most textbooks on probability theory, where the latter is presumed to provide the criterion for testing the former.*"

The above mentioned book by Pearl [22] already contained a complete body of knowledge for Bayesian networks, in which the fundamentals were deeply justified and explained, and the basic inference algorithms were presented, and finally, with the inclusion of very convincing arguments supported by mathematical results and illustrative examples of how probability could serve as a basis for a simple, non-monotonic, coherent, and sound reasoning system. It also explained how to use this model to solve problems of diagnosis, forecasting, planning, fusion, decision, etc. Finally, he showed the relationships with other artificial intelligence formalisms, such as the theory of belief functions [13,14,29] and other qualitative logical approaches such as default logic [30] and truth maintenance systems [31].

Pearl's research attracted a lot of attention both from artificial intelligence researchers (as the work focused on solving problems from this field and the probabilistic approach was compared with traditional artificial intelligence procedures) and from statisticians (as they found new applications for their work and the fact that this new field could serve as a meeting point for researchers that were scattered inside classical statistics). Since this beginning, it has undergone a spectacular development. In the following list, we indicate what we consider to be the main achievements that impelled the use of probabilistic graphical models:

- *Inference algorithms.* Pearl's initial algorithm was limited to trees. The first efficient algorithms for general graphs were given by Lauritzen and Spiegelhalter [32] and by Shafer and Shenoy [33] who also showed that the same algorithms could be applied to computing with other representations of information by determining the basic operators (*marginalization* and *combination*) and giving a set of basic properties under which local propagation algorithms can be applied. This enabled general problems to be solved, but it was also showed that inference in Bayesian networks was #P-complete [34]. This lead to the development of approximate algorithms, mainly based on Monte Carlo simulation [35].
- *Influence diagrams.* Influence diagrams [26] were introduced as a graphical language for specifying complex decision scenarios. It was soon recognized, however, that they could also be used to develop algorithms to compute optimal policies [36,37]. Finally Cooper [38] showed the relationship between Bayesian networks and influence diagrams.
- *Learning.* One of the reasons for the success of Bayesian network models has been the development of methods for inducing a model from a raw set of observations, making them an essential tool for data mining, with clear advantages over other methodologies: the existence of a precise and intuitive semantics for the learned graphical structures. They are not, therefore, simple black boxes enabling future outcomes to be predicted, but the links of the graph can be interpreted in terms of relevance–independence relationships and sometimes we can even assign a causal meaning to them. The task of learning has been separated into two steps: determination of the structure and estimation of the parameters, with the first considered to be the main one since standard statistical methods can be applied to parameter learning. For structural learning, there was an article by Chow and Liu [39] in which an algorithm was proposed to determine a tree structure from a sample. This procedure was extended by Rebane and Pearl [40] to recover a causal polytree from data. The approach was based on the assumption that the data could be perfectly represented by a polytree and then some conditional independence tests should be carried out in order to determine the polytree. It is important to remark that while the Chow–Liu algorithm verified a global optimality condition (always recovering the best tree approximation of a multidimensional probability distribution), the Rebane-Pearl algorithm was based on the verification of a set of hypotheses and when they are not verified there is no guarantee that the result is the best approximation. This method is based on a series of independence tests being carried out and is called the *constraint-based approach*. It has its main exponent in the so-called PC algorithm [41] for general directed acyclic graphs.

Most of the present approaches to learning are based on the *score + search strategy*. This method determines the model which optimizes a score or metric function which measures how appropriate a graph is for the observational data. An entropy-based procedure (Kutato) was proposed by Herskovits and Cooper [42]. The most important contribution, however, was the proposal of the Bayesian score by Cooper and Herskovits [43]. In this paper, it was assumed that there was a prior distribution for the structures and for the

parameters given the structure. Under certain assumptions, it was possible to compute the posterior probability of a structure given a set of data, which was the value of the Bayesian score. A greedy algorithm (K2) was proposed to find a graph with maximum posterior probability. The importance of the score is due not only to the fact that it provided a global fitness measure for the structure but also to several reasons: it is possible to split its computation in such a way that small local changes in one part of the graph do not bring about a complete computation of the score, only being necessary to make certain local computations involving the modified nodes and arcs; it provided a methodology which could be applied to different problems such as discretization; the score of a graph can be interpreted in terms of probability of being the true model and this also opened the possibility of using several models for the same problem by combining them with their posterior probability [44] as weight.

- *Conditional Gaussian networks.* Most of the theory in Bayesian networks has been developed for discrete variables (multinomial case). When continuous variables are given, then the most common practice is to discretize them by dividing the range of possible values into a finite collection of intervals. Some models do exist, however, which allow the continuous variables to be treated directly. The conditional Gaussian network introduced by Lauritzen and Wermuth [45] allows continuous and discrete variables to be specified, with the restriction that a discrete variable cannot be the child of a continuous one and that the conditional distributions of the continuous variables given the discrete ones is a multi-dimensional Gaussian distribution. One method to propagate information in this graphical structure was proposed by Lauritzen [46], but it was shown to be numerically unstable and was later improved by Lauritzen and Jensen [47]. In order to learn Gaussian networks we can apply the two basic approaches (score + search or the constraint-based approach) although most work has concentrated on defining score metrics [48].
- *Applications.* One of the first real world applications of Bayesian networks showing the potentials of this tool was MUNIN which was developed at the University of Aalborg as part of a European Project (Esprit P599) in an attempt to create an expert assistant for electromyography [49]. MUNIN is able to manage relationships among more than 1000 variables by modeling a small portion of the human neuro-muscular system, and is able to efficiently compute with them. An additional result of this project was the creation of Hugin Expert A/S in 1989, a company that implemented a general tool for creating and using Bayesian networks with an intuitive and easy-to-use graphical interface [50]. Hugin and MUNIN have been the basis and inspiration for a number of successful applications and general tools which have been produced in subsequent years.

This work provided a solid basis for the fast development of a methodology for the application of graphical models to a number of different problems. There has been increasing interest in this field, which has evolved in many directions. It would be difficult even to summarize these. From our point of view, the main theoretical problem which is been addressed is the mathematical formulation of the concept of causality, providing methods to identify causalities from experimental and observational data, as well as models to compute the probabilities after interventions and the consequences of plans. Although causality is one of the most basic tools for humans to understand the outside world, it has evaded being captured in a formal model to provide methods for a systematic treatment in practice. While Pearl [51] has provided the most complete and influential interpretation of causality in terms of graphical models, there are however other approaches [52] and we are far from a unified model capturing the full meaning and behavior of causality.

In the 1990s, there was also intensive work into specialization of the general models for specific problem solving (covered in the next section) and for specific domains or types of applications. In this direction, we could mention the work on dynamic Bayesian networks [53,54]. These are models for systems which evolve over a period of time. The basic scheme is a sequence of static descriptions with each representing the state of the system at a particular time, and temporal relationships showing how each static submodel depends on the previous ones. This is a very general framework including hidden Markov models and Kalman filters as particular cases, and poses special problems for the development of algorithms for inference and learning. Another important modeling tool is the use of object-orientated Bayesian networks [55]. As in programming languages, object orientation enables the definition of generic networks fragments called classes. They can be instantiated a number of times resulting in repetitions of the defined structure, which is very useful for situations where we have copies of different network fragments sharing the same structure and conditional probabilities.

Bayesian networks and influence diagrams assume a directed acyclic graph to represent relationships between the variables of the problem. There are, however, other probabilistic graphical models in which the structure is represented by a different type of graph. For example, in Markov models [22,56] the underlying graph is undirected. A model covering directed and undirected links is the chain graph model [45]. However, the theory is not so well developed as for the particular case of Bayesian networks and they are not so widely used.

4. Problem resolution with graphical models

We assume that we have a finite set of variables $\mathbf{X}$. A Bayesian network for this set of variables is a directed acyclic graph in which there is a node for each variable in $\mathbf{X}$ and a conditional distribution of this variable given its parents. The graph represents conditional independence relationships between the variables according to the d-separation criterion [57]. Taking into account these relationships, a joint probability for the variables in $\mathbf{X}$ can be obtained by multiplying the conditional probability distributions (one for each variable given its parents).

In most cases, the variables in $\mathbf{X}$ are categorical (i.e. they take values on a finite set). The set of values in which $Y \in \mathbf{X}$ takes its values will be denoted as Ω_Y.

The basic *inference* problem is the following: we have some observed values (evidence) $\mathbf{O} = \mathbf{o}$ for a subset $\mathbf{O} \subset \mathbf{X}$ and an interest variable, $Y \in \mathbf{X}$. We want to compute the conditional probability of Y given $\mathbf{O} = \mathbf{o}$, namely

$$P(Y = y|\mathbf{O} = \mathbf{o}), \quad \forall y \in \Omega_Y$$

The observed variables are not predetermined, and any network variable can be incorporated into the evidence or play the role of interest variable. In this way, Bayesian networks can be used to predict the values of a future variable (forecasting), to determine the cause of present observations (diagnosis), or a combination of these basic procedures (e.g. when we determine the state of the world from some partial observations to predict the value of some variable in the future).

The basic *learning* problem is the following: we start with a database, $\boldsymbol{d}$, with observations for all the variables in $\mathbf{X}$. We want to induce a Bayesian network from $\boldsymbol{d}$.

In addition to these basic problems, there are other important questions which can be solved with the help of graphical models. In the following subsections, we will briefly describe the ones which we consider to be the most relevant.

4.1. Supervised classification

In recent years, there has been an important increase in the number of probabilistic graphical models for supervised classification tasks [58]. In this problem, we have a set of data $\boldsymbol{d}$ with observations for variables $\mathbf{X}$ and a class variable C. We want to build a model which is able to predict the class C from observations of variables in $\mathbf{X}$. The graphical representation of Bayesian classifiers is intuitive, allowing domain experts to understand the underlying probabilistic classification process without a deep knowledge of Bayesian classifiers.

A model hierarchy of increasing complexity could be established for Bayesian classifiers, where the naive Bayes is at the bottom of this hierarchy and a general Bayesian network is at the top. The restrictions imposed on naive Bayes, selective naive Bayes, semi-naive Bayes, tree augmented naive Bayes and k-dependence Bayesian classifiers are due to the type of relations between the predictor variables that they consider, and due to the fact that in all these paradigms the class variable is considered as the parent of the predictor variables. Despite their limitations, these Bayesian classifiers provide a set of properties that can be appreciated by domain experts. Their graphical structure facilitates interpretability and understanding, specifying the assumed conditional independence relationships. The conditional and marginal distributions of the model could be of interest for a better understanding of the uncertainty of the analyzed domain. Another interesting characteristic is that when computational time is a critical factor, these Bayesian classifiers are quickly learned from a database. Furthermore, once the Bayesian classification model has been induced, it is able to quickly obtain a prediction for an unseen example and add the knowledge of this unseen example to the model.

Naive Bayes [59,60] is the simplest Bayesian classification model. It is built on the assumption of conditional independence of the predictive variables given the class. Although this assumption is violated on numerous occasions in real domains, the paradigm still performs well in many situations [61,62]. Making this assumption, the prediction of the class for an unseen instance is simplified. Different works can be found in the literature which show the restricted capabilities of the decision surfaces related with the naive Bayes paradigm. In the case of binary variables, Minsky [60] shows that the decision surfaces are hyperplanes, while Domingos and Pazzani [61], Duda and Hart [59], and Peot [63] extend Minsky's result for a more general type of predictor variable. Several adaptations of the naive Bayes paradigm have been proposed under different circumstances: imputation of missing data [64], interval estimation [65], incremental versions when new data are coming [66], feature selection [67,68], and Bayesian approaches [69].

Due to its simplicity, the naive Bayes paradigm cannot perceive dependencies between predictive variables. The *semi-naïve Bayes* classifier [70] tries to avoid the assumptions of the classical naive Bayes by taking into account new variables. These new variables are the Cartesian product of some of the original variables. In [70] a greedy wrapper approach for building a semi-naive Bayes model where the irrelevant variables are removed and the correlated variables are joined in a Cartesian product is proposed. It starts with an empty set of variables and labels all the examples with the most common class value. Thus, until non-improvement is reached, the method selects the most accurate option at each step: either to include a new variable or to join an existing variable with a new variable. The joining is performed by means of a Cartesian product.

The *tree augmented naive Bayes* (TAN) classifier takes into account relationships between the predictive variables by means of a naive Bayes structure that is extended with a tree structure between the predictive variables. Fig. 1 shows a TAN model. The adaptation of the Chow–Liu [39] algorithm to build a TAN classifier is proposed by Friedman et al. [71]. The difference is that now the weight of the link between two variables is their conditional mutual information given the class. A wrapper greedy approach to induce a TAN structure is presented in [72]. In order to overcome the difficulties of the greedy search the use of a floating search heuristic is proposed in [73]. Extensions of the TAN paradigm include the *forest*

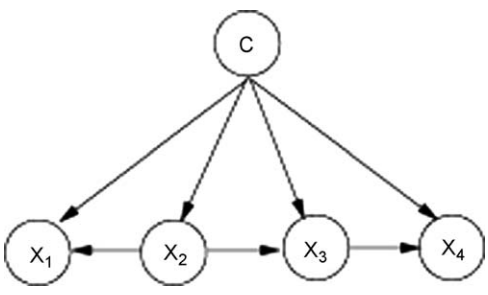

Fig. 1. Structure of a tree augmented naive Bayes model.

augmented network (FAN) algorithm [74], the *tree-augmented naive credal* classifier [75], a Bayesian approach [76], as well as an incremental version [77].

The tree augmented naive Bayes classification model is limited by the number of parents of the predictive variables. A predictive variable can have a maximum of two parents: the class and another predictive variable. The *k dependence Bayesian classifier* (*k* DB) [78] attempts to avoid this restriction by allowing a predictive variable to have up to *k* parents in addition to the class.

Kontkanen et al. [79] present an approach where *Bayesian multinets for classification* are learned from data allowing the representation of context-specific conditional independencies.

More general Bayesian classifiers can be obtained without the restriction common to the previous paradigms where the class variable was at the root of the graph. In this sense, Sierra and Larrañaga [80] propose the use of genetic algorithms for searching for the best *Markov blanket of the class variable*, using the accuracy of the model as the score for guiding the search. Although some authors have approached the supervised classification problem with algorithms that learn unrestricted Bayesian networks, the use of specific paradigms for this task seems to be more appropriate for simplicity, parsimony and computational reasons.

The learning of probabilistic classification models can be approached from either a generative or a discriminative point of view. Generative methods attempt to maximize the unconditional log-likelihood while the aim of *discriminative methods* is to maximize the conditional log-likelihood. In recent years, several approaches for discriminative Bayesian classifiers have been proposed [81–83].

4.2. Clustering

One of the main problems that arises in a great variety of fields, including pattern recognition, machine learning, and statistics, is the so-called *data clustering problem* [84]. Given some data in the form of a set of instances, $\boldsymbol{d}$, of variables $\mathbf{X}$ with an underlying (non-observed) group-structure, C, data clustering may be roughly defined as the description of the underlying group-structure when the true group membership of every instance is unknown. Each of the groups in the data at hand is called a *cluster*. In most approaches for solving data clustering problems, there are mainly two tasks to be performed: identification of the number of clusters that exist in the underlying group-structure of the data at hand (here referred to as K) and induction of a description of these K clusters. Unfortunately, most of the time K is unknown. When this is the case, there are basically two approaches for overcoming this lack of information: the use of a data preprocessing step to determine the most likely K for the current data, or the use of different values for K in the subsequent induction of the description of the clusters that exist in the given data in order to select the most convenient value a posteriori. In this section, we assume that we are somehow provided with K.

The existence of a random variable C whose entries in the database are hidden means that data clustering is usually referred to as an example of *learning from unlabeled data* or, simply, *unsupervised learning*.

Due to the lack of a priori knowledge of the mechanism that caused the instances grouped in the database, *partitional data clustering* involves the simplest definition of the description of the clusters that exist in the data. The partitional approach therefore reduces data clustering to completing every instance of the original unlabeled database with the label of the cluster whose physical process generated the instance. As every case of the database must belong to exactly one of the K existing clusters, the clusters are exhaustive and mutually exclusive, i.e. they constitute a partition of the database. A paradigmatic heuristic algorithm for solving partitional data clustering problems is the well-known *K-means algorithm* [85,86].

When the a priori knowledge of the mechanism that produced the data in $\boldsymbol{d}$ includes a parametric form (e.g. multinomial, Gaussian, etc.) of the joint probability distributions for $\boldsymbol{Y} = (\boldsymbol{X}, C)$ of the physical processes reflected in the K clusters of $\boldsymbol{d}$, we usually prefer the probabilistic or model-based approach to data clustering rather than the partitional approach. The objective of *probabilistic data clustering* is to describe the K underlying clusters of $\boldsymbol{d}$ by modeling the mechanism that generated the data in $\boldsymbol{d}$. Consequently, the only thing that needs to be learnt is the set of parameters that completely define this mechanism, i.e. the parameters of the probability distribution that determine which of the physical processes associated with the clusters of $\boldsymbol{d}$ is selected at each time as well as the parameters of the joint probability distributions for $\boldsymbol{Y}$ of these physical processes. This set of parameters is usually referred to as the model parameters. As a result, probabilistic data clustering may be redefined by finding the best set of parameters for the model of the mechanism that generated $\boldsymbol{d}$ according to the data clustering criterion.

The most classical solution to probabilistic data clustering is based on the theory of *finite mixture models* [59,87]. In this case, the mechanism that generated the instances of $\boldsymbol{d}$ is modeled as a mixture of K joint probability distributions for $\boldsymbol{Y}$ with certain proportions. Let π^g denote the probability that the physical process corresponding to the g-th cluster of $\boldsymbol{d}$ is somehow selected by the mechanism that generated an instance of $\boldsymbol{d}$ for any g. In addition, let $\boldsymbol{\theta}^g$ and $\rho(\boldsymbol{y}_l|c^g, \boldsymbol{\theta}^g)$ represent the parameters of the joint probability distribution for $\boldsymbol{Y}$ corresponding to the physical process of the g-th cluster of $\boldsymbol{d}$, and the probability or density that the l-th case of $\boldsymbol{d}$ is somehow generated given that the physical process of the g-th cluster of $\boldsymbol{d}$ is selected, respectively, for all g and l. The likelihood of $\boldsymbol{d}$ given the model parameters $\boldsymbol{\theta} = (\pi^1, \ldots, \pi^K, \boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^K)$ can then be expressed as follows:

$$L(\boldsymbol{d}|\boldsymbol{\theta}) = \prod_{l=1}^{N} \sum_{g=1}^{K} \pi^g \rho(\boldsymbol{y}_l|c^g, \boldsymbol{\theta}^g). \tag{1}$$

Since the data clustering criterion is given by Eq. (1), probabilistic data clustering merely maximizes this equation. As the set of model parameters $\boldsymbol{\theta}$ is allowed to vary freely as far as it is

consistent with probability theory, Eq. (1) is usually maximized with the help of a heuristic search strategy. The standard deterministic heuristic search strategy for this purpose is the well-recognized *expectation–maximization algorithm* [87,88]. In general terms, the expectation–maximization algorithm works by iterating between an expectation step and a maximization step until no further improvement of the data clustering criterion is found. The expectation step scores every set of model parameters in the search space by computing the expected likelihood of the complete $\boldsymbol{d}$ given that particular set of model parameters. The expectation is calculated with respect to the best set of model parameters found so far. On the other hand, the maximization step replaces the current set of model parameters with the best set among those scored in the expectation step. The expectation–maximization algorithm is known to converge to a local optimum under mild conditions. The main disadvantage of probabilistic data clustering based on unsupervised learning of finite mixture models is precisely the computational expense that the problem optimization process itself involves due to the largeness of the search space and the number of model parameters to be estimated.

The learning from data of probabilistic graphical models for clustering purposes is a bit different from the two phases (structure learning and parameter learning) when learning for discovering associations or even for supervised classification. When Bayesian networks and conditional Gaussian networks are used to solve probabilistic data clustering problems, the unsupervised model learning process is not usually decomposed into two subtasks as addressed above. Therefore, unsupervised learning of Bayesian networks and conditional Gaussian networks involves a search for the best model in the joint search space of model structures and model parameters. Consequently, the already addressed difficulty of learning Bayesian networks (BN) and conditional Gaussian networks (CGN) from complete data is aggravated when facing unlabeled data.

The factorability property of the penalized likelihood and the marginal likelihood when faced with complete data is not verified for the case of incomplete data, that is for unsupervised classification problems. Based on [89,90], in [91] two approximations of the marginal likelihood are presented. It should be noted that when dealing with incomplete data, the exact computation of this score is typically intractable [43]. However, these approximations for the marginal likelihood do not factorize into scores for families of nodes. Hence, the model structure search procedure could not take advantage of the factorability and would have to recompute the approximate score for the entire structure although only the factors of some families of nodes had changed.

The second approach, that is usually considered in order to overcome the difficulties of learning probabilistic graphical models from incomplete data in general and from unlabeled data in particular by using the marginal likelihood score to guide the model structure search, relies on the expectation–maximization algorithm [92]. This approach avoids the drawbacks of the marginal likelihood for incomplete data (i.e. intractable exact computation, inefficiency of the approximations for it, and loss of the factorability property). Among the different techniques that belong to this approach, the well-known Bayesian structural expectation–maximization algorithm [93] is probably the most representative and studied algorithm for unsupervised learning of BNs and CGNs from a Bayesian approach. Due to its good performance, this model induction algorithm has received special attention in the literature and has motivated several variants or instances of itself [94–98].

4.3. Abductive reasoning

Finding the *diagnostic explanations* is also known as *abductive inference* [99] and consists in finding the state of the world (configuration) that is the most probable given the evidence [22]. As in the basic inference problem, we have a set of observations $\mathbf{O} = \mathbf{o}$ (known as the *explanandum*), but the aim now is to obtain the best configuration of values for the explanatory variables (the *explanation*) which is consistent with the observations and which needs to be assumed to predict them. For example, we might have a patient with various observed symptoms and two possible diseases; instead of being interested in the posterior probability of each of the diseases, what we want to know is the most probable combination of possibilities (for example having the first and not having the second). The advantage of this last option is that it takes into account the possible high order relationships between the diseases (for example, if given the symptoms, they have a tendency to appear together, or one excludes the other).

Depending on what variables are considered as *explanatory*, two main abductive tasks in BNs are identified:

- *Most probable explanation* (MPE) or *total abduction*. In this case all the unobserved variables ($\mathbf{U} = \mathbf{X} - \mathbf{O}$) are included in the explanation [1]. The *best* explanation is the assignment $\mathbf{U} = \mathbf{u}^*$ which has maximum a posteriori probability given the evidence, i.e.

  $$\mathbf{u}^* = \arg \max_{\mathbf{u} \in \Omega_{\mathbf{U}}} P(\mathbf{U} = \mathbf{u} | \mathbf{O} = \mathbf{o})$$

  Searching for the best explanation has the same complexity (NP-hard; Shimony [100]) as probability propagation, in fact the best MPE can be obtained by using probability propagation algorithms but replacing summation by maximum in the marginalization operator [101]. However, as it is expected for there to be several hypotheses competing for the *explanandum*, our goal is usually to obtain the K MPEs. Nilsson [102] showed that by using the algorithm in Dawid [101] only the first three MPEs can be correctly identified, and proposed a clever method to identify the remaining $K - 3$ explanations.
- *Maximum a posteriori assignment* (MAP) or *partial abduction* [103]. The goal of this task is to alleviate the over specification problem by considering as target variables only a subset of the unobserved variables called the *explanation set* ($\mathbf{E}$). We then look for the maximum a posteriori assignment of these variables given the explanan-

dum, i.e.

$$\mathbf{e}^* = \arg\max_{\mathbf{e}} P(\mathbf{E}=\mathbf{e}|\mathbf{O}=\mathbf{o}) = \arg\max_{\mathbf{e}} \sum_{\mathbf{r}} P(\mathbf{E}=\mathbf{e},\mathbf{R}=\mathbf{r}|\mathbf{O}=\mathbf{o})$$

where $\mathbf{R}=\mathbf{X}-\mathbf{O}-\mathbf{E}$.

For example, in the problem of diseases and symptoms we may have some unobserved predisposition factors, or intermediate conditions, but we are not interested in inferring about them. They constitute the set $\mathbf{R}$.

This problem is more complex than the MPE problem because it can be NP-hard even for cases in which MPE is polynomial (e.g. polytrees) [104,105], although Park and Darwiche [104,106] have proposed exact and approximate algorithms to enlarge the class of efficiently solved cases. With respect to looking for the K best explanations, exact and approximate algorithms which combine Nilsson algorithm [102] with probability trees [107] have been proposed by de Campos et al. [108].

4.4. Decision-making

In many cases, we are not only interested in computing posterior probabilities, but also in optimizing the available decision options we have in a situation where the outputs are uncertain.

A basic situation is the troubleshooting task [109,110]: we have observed the wrong behavior of a device and we want to determine an optimal sequence of actions to fix it. There are two possible types of steps: observations and actions (repair steps), and each has an associated cost. Each action can fix the problem or fail to do so. We must determine the optimal sequence of observations and actions minimizing the expected cost of repair, under a set of observations $\mathbf{O}=\mathbf{o}$. This is an NP-hard problem in general, but in some situations it can be solved with an efficient greedy algorithm.

A general language for representing and solving complex decision problems is provided by the influence diagram (ID) model [26]. IDs are directed acyclic graphs with three types of nodes: *decision nodes*, $\mathbf{D}$ (mutually exclusive actions which the decision maker must choose between); *chance nodes*, $\mathbf{X}$ (events that the decision maker cannot control); and *utility nodes*, $\mathbf{U}$ (representing decision maker preferences). Links represent dependencies: probabilistic for links into chance nodes, informational for links into decision nodes (states for decision parents are known before the decision is taken), and functional for links into value nodes (there will be a utility function for each utility node, with a utility value for each configuration of its parents).

Direct predecessors of chance or value nodes are called *conditional* predecessors; direct predecessors of decision nodes are designated *informational* predecessors. The set of direct and indirect predecessors of X is denoted $\mathrm{pred}(X)$.

An important condition in IDs is that it is assumed to be a directed path comprising all decision nodes. This defines a total order in which decisions must be taken. Let us assume that the ordered vector of decisions is given by $D_1,\ldots,D_m$. The semantics of IDs usually assumes that the decision maker remembers previous observations and decisions (*non-forgetting assumption*). We shall therefore consider that each decision $D\in\mathbf{D}$ depends on its direct predecessors and the direct predecessors of the decisions previously taken. This set is called the *information set* for D, denoted by $\mathrm{infSet}(D)$. A *policy* for an ID prescribes an action for each decision and for every configuration of the variables on its information set. For each policy and configuration of the chance variables, the global utility is usually assumed to be the sum of the values of the utility functions of the different utility nodes.

An *optimal policy*, d^*, is a policy which maximizes the decision maker's *expected utility* for all the decision variables. To find an optimal policy will be the objective for ID evaluation algorithms. For further details, see [111]. When $\mathrm{infSet}(D)$ is very large, it may be impossible to compute or even represent the decision function for D. The representation of a policy for decision variable D is exponential in the number of variables in $\mathrm{infSet}(D)$. As the number of variables in $\mathrm{infSet}(D)$ includes all the decision variables previous to D in sequence $(D_1,\ldots,D_m)$ and all the chance predecessors of them, this set can be large for the last variable D_m. It may therefore become unfeasible even for small influence diagrams. Some computational complexity results for general influence diagrams (alternative representations of uncertainty) can be found in [112].

A solution to this important problem can be to drop the non-forgetting hypothesis by explicitly giving for each decision variable the list of variables that are known when this decision is taken. This is the case of limited memory influence diagrams (LIMID) introduced by Nilsson and Lauritzen [113], in which only a subset of $\mathrm{infSet}(D)$ is considered to define a strategy for D. If we remove some really relevant variables, then the solution is approximate in relation to what could be obtained with the non-forgetting assumption, but by doing this the problem can become solvable.

Horsch and Poole [114] also propose an approximate algorithm. It considers a reduced number of variables from $\mathrm{infSet}(D)$ for each decision, but the relevant variables are computed by the model. The approximation is obtained by building decision functions with an incremental procedure. The relevant variables are added one by one in an attempt to maximize the utility expected value.

An alternative approach has been to design Monte Carlo simulation algorithms [115–118], but although the problem is more complex than the computation of posterior conditional distributions in Bayesian networks, the research effort has been much lower.

There are special cases of influence diagrams for which specific strategies to optimize decisions can be designed. In Markov decision processes [119], we have different simple decision problems which are repeated in different time slices with an unbounded time horizon. The reward of our decision depends on the state of the world and may be uncertain. The state of the world in time i depends on the state of the world and our decision in time $i-1$. When the state of the world is not directly observable (we only have some evidence depending on

the true state of the world) we have a partially observable Markov decision process.

Although influence diagrams are important for the representation of complex decision problems under uncertainty, they also have important limitations and there have been attempts to cope with these by generalizing the initial model. From our point of view, some of the most important lines for further developments are the following:

- *Asymmetrical models*. In the basic ID model, it is assumed that if we must make a decision D, the scenarios that arise depending on the alternative we chose are symmetrical. This means that we are confronted with the same decision and observed variables (although perhaps with different selections and different uncertainty values). There are, however, simple examples where this is not the case. If we decide to perform a test, then the set of observed variables is not the same as if we do not perform the test. This can be solved by adding artificial values for the observed variables as *no-test*, and for more complex situations, artificial variables, although this makes problem specification more intricate. Models which are able to directly represent asymmetrical problems can be found in [120–123].
- *Non-sequential problems*. There are problems where the existence of a predefined order in which decisions must be made is not true. For example, we may have several tests for a disease and we must determine the tests to be applied and the best application sequence. Again, it is preferable for there to be a more expressive language and more general resolution procedures instead of making tangled transformations to the original influence diagram definition. In this sense, we can cite the work of Jensen and Vomlelova [124], where the unconstrained influence diagrams are introduced for this purpose.
- *Multiple agents*. Influence diagrams assume a unique agent. Koller and Milch [125] have extended this setting by proposing the so-called multi-agent influence diagrams (MAIDs) in which we can have several competing agents. This extends the application scope of graphical models, establishing new relationships with the traditional *game theory*.

5. Optimization with probabilistic graphical models

The use of graphical models in optimization begins by assuming that through the use of probabilistic models, useful information about the search space can be learned from a set of solutions which have already been inspected or from the problem structure. This information can be used to conduct a more effective search. Our analysis will focus on the class of optimization methods that use probabilistic graphical models to organize the search.

5.1. Estimation of distribution algorithms

Estimation of distribution algorithms (EDAs) [126–130] are evolutionary algorithms that work with a set (or population) of points. Initially, a random sample of points is generated. These points are evaluated using the objective function, and a subset of points is selected based on this evaluation. Hence, points with better function values have a higher chance of being selected. A probabilistic model of the selected solutions is then built, and a new set of points is sampled from the model. The process is iterated until the optimum has been found or another termination criterion is fulfilled. The general scheme of the EDA approach is shown in Fig. 2.

One essential assumption of these algorithms is that it is possible to build a probabilistic model of the search space that can be used to guide the search for the optimum. A key characteristic and crucial step of EDAs is the construction of this probabilistic model. If there is information available about the function (e.g. variable dependencies), this can be exploited by including parametrical and/or structural prior information in the model. Otherwise, the model is learned exclusively using the selected population. Several probabilistic models with different expressive powers and complexities can be applied. These models may differ in the order and number of the probabilistic dependencies that they represent.

Different classifications of EDAs can be used to analyze these algorithms. Regarding the way learning is done in the probability model, EDAs can be divided into two classes. One class groups the algorithms that perform a parametric learning of the probabilities, and the other one comprises those algorithms where structural learning of the model is also done. Population-based incremental learning (PBIL) [131], compact GA (cGA) [132], the univariate marginal distribution algorithm (UMDA) [129], and the factorized distribution algorithm that uses a fixed model of the interactions in all the generations (FDA) [133] all belong to the first class of algorithms. Likewise, the mutual information maximization for input clustering algorithm (MIMIC) [134], the extended compact GA (EcGA) [135] and EDAs that use Bayesian and Gaussian networks [136–141,130] belong to the second class. Another way of presenting EDAs is to classify them according to the complexity of the probabilistic models used to capture the interdependencies between the variables.

1 Set $t \Leftarrow 0$. Generate M points randomly
2 **do**
3 Evaluate the points using the fitness function
4 Select a set S of $N \leq M$ points according to a selection method
5 Calculate a probabilistic model of S
6 Generate M new points sampling from the distribution represented in the model
7 $t \Leftarrow t + 1$
8 **until** Termination criteria are met

Fig. 2. Estimation of distribution algorithms: evolutionary computation based on learning and simulation of probabilistic graphical models.

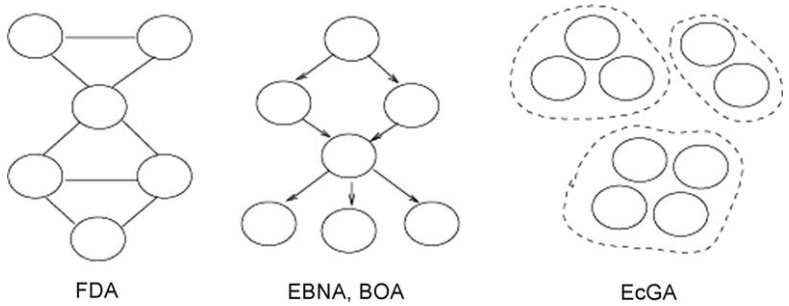

Fig. 3. Graphical representation of probability models for the proposed EDAs in combinatorial optimization with multiple dependencies (FDA, EBNA, BOA, and EcGA).

The *univariate marginal distribution algorithm* (UMDA) assumes that all variables are independent and consequently $p(\mathbf{x})$ can be factorized as follows: $p(\mathbf{x}) = \prod_{i=1}^{n} p(x_i)$.

The *tree-based estimation of distribution algorithm* (Tree-EDA) [142] uses a factorization that is based on a forest. The factorization is constructed using the algorithm introduced in [39] that calculates the maximum weight spanning the tree from the matrix of mutual information between pairs of variables. Additionally, a threshold for the mutual information values is used when calculating the maximum weight spanning tree to allow for disconnected components in the graphical structure.

Other EDA approaches in the literature propose that the joint probability distribution be factorized by statistics of order greater than 2. Fig. 3 shows different probabilistic graphical models that are included in this category. As the number of dependencies between variables is greater than in the previous categories, the complexity of the probabilistic structure as well as the computational effort for finding the structure that best suits the selected points is bigger. Therefore, these approaches require a more complex learning process. Some of the *EDA approaches based on multiple connected Bayesian networks* are as follows:

- The factorized distribution algorithm (FDA) [133] applies to additively decomposed functions for which, using the running intersection property, a factorization of the mass-probability based on residuals and separators is obtained.
- In [137] a factorization of the joint probability distribution encoded by a Bayesian network is learnt from the database containing the selected individuals in every generation. The developed algorithm is called estimation of Bayesian network algorithm (EBNA), and it uses the Bayesian information criterion (BIC) score to measure the quality of the Bayesian network structure together with greedy algorithms that perform the search in the model space.
- In [141] the authors propose an algorithm called Bayesian optimization algorithm (BOA) which uses the Bayesian Dirichlet equivalent metric to measure the goodness of every structure. A greedy search procedure is also used for this purpose. The search starts in each generation from scratch.
- The Extend compact Genetic Algorithm (EcGA) proposed in [135] is an algorithm of which the basic idea consists in factorizing the joint probability distribution as a product of marginal distribution of variable size.

Due to the stochastic nature of EDAs, random process theory would seem to provide an appropriate set of tools for describing their behavior. In particular, Markov chains constitute a proper and natural mathematical model for this purpose.

6. Some applications and software

In this section, we will review some applications of probabilistic graphical models in two challenging modeling arenas: forensic and genomics. Of course, there are many more possible fields, including medicine, meteorology, speech recognition, intelligent tutoring, gambling, monitoring, etc. [143,144]. We have only selected two examples in which the use of graphical models provides a large departure from the use of conventional techniques.

We will also mention some of the most widely used academic and commercial software for working with probabilistic graphical models.

6.1. Modeling applications

6.1.1. Forensic

Bayesian networks have been proposed as a method of formal reasoning that could assist forensic scientists in understanding the dependencies that may exist between different aspects of evidence. Since the early 1990s, both legal scholars and forensic scientists have shown an increased interest in the applicability of Bayesian networks in judicial contexts. While lawyers merely tend to be concerned with structuring cases as a whole, forensic scientists focus on the evolution of selected items of scientific evidence, such as fibres or blood.

Bayesian networks have been proposed in legal reasoning to structure aspects of complex and historically important cases. Edwards [145] provided an alternative analysis of the descriptive elements presented in the Collins case. Schum [146] worked on a probabilistic analysis of the Sacco and Vanzetti case with emphasis on the credibility and relevance of evidence given by human sources, *i.e.* testimony. Probabilistic case analysis was also proposed for the Omar Raddad [147] and O.J. Simpson [148] cases.

Interest in the probabilistic evaluation of DNA evidence has grown considerably over the last decade. Topics such as the

assessment of mixtures, consideration of error rates and effects of database selection have increased the interest in forensic statistics. DNA evidence based on fine-grained network fragments focussing on individual genes and genotypes has been evaluated [149]. When the crime stain contains material from more than one contributor, as for instance in the case of rape and other physical assaults, the resulting Bayesian networks are more complicated [150]. Graphical structures for Bayesian networks can be derived from initial pedigree representations of forensic identification problems, as for instance in a typical case of disputed paternity [151]. In some situations, the crime suspect can be selected through a search in a database [152]. A forensic scientist would also be required to explain what chance there may be to observe the findings when some alternative propositions were true, for example, that the crime stain comes from an unknown person who is unrelated to the suspect. The effect that the potential of error through a false-positive may have on the value of a reported match has been evaluated in a Bayesian framework by Thompson et al. [153].

One book containing examples of the application of Bayesian networks in forensic science is by Taroni et al. [154].

6.1.2. Genomics

The Human Genome Project has given rise to the development of several technologies (fast sequencing techniques, genotype maps, or DNA microarrays) which have provided a huge amount of data which need to be analyzed to extract relevant information (mainly, understanding of cellular processes and the relationships of genomic information with known diseases). These procedures are always noisy and subject to random variations. In this task, probabilistic graphical models play an important role. We can cite the discovery of regulatory cellular networks from measurements of gene expression levels [155–157], the development of classifiers taking gene expression data as attributes [158], non-supervised classification of genes [159], analysis of DNA sequences for motif (transcriptional regulatory regions) identification [160], or models for haplotype inference from single nucleotide polymorphisms (SNPs) data and their relationships with known diseases [160,161]. With respect to alternative models, Bayesian networks offer new possibilities: the models have a clear semantics in terms of conditional independence and probability; in some cases it is possible to discover causal relationships between the variables instead of single correlations; it is possible to include contextual information (for example two genes are coexpressed but only under certain conditions); it is possible to integrate different sources of knowledge with different kinds (for example qualitative expert knowledge, observational data, and experimental data). This is a rapidly growing research field and one which poses important challenges. A survey can be found in [162].

6.2. Software

As a result of the growing interest in Bayesian networks, many software packages have been designed to support its development. The book by Korb and Nicholson [143] contains a detailed comparison of some of them. *Elvira* [163], *BN PowerConstructor* [164], *BNT* [165], *BUGS* [166], *gr* [167], *JavaBayes* [168], and *Tetrad* [169] are some of the academic software for learning and inference with Bayesian network. The list of commercial tools include *Hugin* [50], *BayesiaLab* [170], and *Netica* [171].

Specific software for forensic identification based on probabilistic graphical models can be found in [172].

7. Some alternative models and extensions of Bayesian and Gaussian networks

Although Bayesian and Gaussian networks are the two most extended and studied probabilistic graphical paradigms, several other proposals for normative models which are able to deal with uncertainty have been presented in the literature, among them *Markov random fields*, *factor graphs* and *Kikuchi approximations*. In this section, we will also briefly review *credal networks*, which are a generalization of Bayesian networks in which the knowledge of the probability values is not precise.

Markov random fields, also known as Markov networks or undirected graphical models [56,173], have two components: a set of nodes, each of which correspond to a variable (or group of variables) and a set of undirected links each of which connect a pair of nodes.

The graphical semantics of a Markov random field is based on conditional independence properties of the graph which are determined by simple graph separation criteria. To graphically test the property that node sets A and B are conditionally independent given node set C, we should consider all paths that connect nodes in set A to nodes in set B. If all such paths pass through one or more nodes in set C, then all paths are ''blocked'' and so the conditional independence property holds. However, if there is at least one such path that is not blocked, then the conditional independence property does not hold.

The joint distribution in a Markov random field is written as a product of potential functions $\psi_C(\mathbf{x}_C)$ over the maximal cliques of the graph:

$$p(\mathbf{x}) = \frac{1}{Z}\prod_C \psi_C(\mathbf{x}_C) \tag{2}$$

where the quantity Z, is called the partition function and is a normalization constant. The connection between conditional independence and factorization for undirected graphs is stated by the Hammersley–Clifford theorem [174].

Both directed and undirected graphs allow a global function of several variables to be expressed as a product of factors over subsets of those variables. *Factor graphs* [175,176] make this decomposition explicit by introducing additional nodes for the factor themselves in addition to the nodes representing the variables. By doing this, the joint distribution over a set of variables can be written in the form of a product of factors: $p(\mathbf{x}) = \prod_s f_s(\mathbf{x}_s)$, where $\mathbf{x}_s$ denotes a configuration for a subset $\mathbf{X}_s$ of variables $\mathbf{X}$, and each factor f_s is a function defined on these configurations.

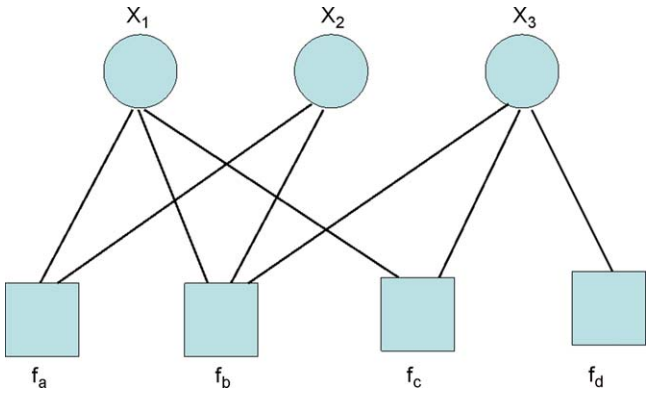

Fig. 4. Example of a factor graph with three variables and four factors.

In a factor graph, there is a node (usually depicted by a circle) for every variable in the distribution. There are also additional nodes (depicted by small squares) for each factor $f_s(\mathbf{x}_s)$ in the joint distribution. Finally, there are undirected links connecting each factor node to all of the variables nodes on which that factor depends.

The factor graph of Fig. 4 provides the following factorization of the joint probability distribution:

$$p(\mathbf{x}) = f_a(x_1, x_2) f_b(x_1, x_2, x_3) f_c(x_1, x_3) f_d(x_3) \quad (3)$$

Directed graphs represent special cases of factor graphs where the factors are local conditional distributions. Similarly, undirected graphs are a special case in which the factors are potential functions over the maximal cliques and the normalizing coefficient can be viewed as a factor defined over the empty set of variables.

Kikuchi approximations of the energy [177] are region-based decompositions of the energy that satisfy certain constraints. Basically, a region-based decomposition of a function can be seen as a function defined on the variables associated to the vertices of a graph (regions). The global function is formed by the composition of local subfunctions defined in those variables grouped in each of the regions. For instance, in the free energy approximation in physics, regions serve as the basic units to define the local energies which are combined to give the global free energy function. Region-based decompositions can be used for the approximation of other measures, for instance for calculating suitable approximations of probability distributions. In this context, an essential question is how to determine a convenient region-based decomposition that maximizes the accuracy of the approximation.

The Kikuchi approximation of a probability distribution from a clique-based decomposition of an independence graph [178] is a particular type of factorization that uses marginal distributions. The marginal distributions in the factorization are completely determined by the independence graph. Given this graph, the clique-based decomposition is formed by the cliques of the graphs and their intersections. All these cliques are called regions.

Credal networks [179] are an extension of Bayesian networks, in which we also have a directed acyclic graph, but instead of having a joint precise global probability distribution we have a closed and convex set of possible distributions. This credal set can be obtained from a conditional credal set for each variable given its parents. Credal networks represent situations in which the probability values are not precisely known, either because of lack of expert knowledge, or because they are estimated from small samples. There has been controversy regarding the importance of the precision of the probability numbers. Since the early years of graphical models, some authors have stressed the difficulty of assessing all the necessary probability values of a full Bayesian graphical model. Fertig and Breese [180] asserted:

> "One of the most difficult tasks in constructing an influence diagram is the development of conditional and marginal probabilities for each node of the network. In some instances probability information may not be readily available, and a reasoner wishes to determine what conclusions can be drawn with partial information on probabilities. In other cases, one may wish to assess the robustness of various conclusions to imprecision in the input."

In addition, Pearl [181] says:

> "We sometimes feel more comfortable assigning a range, rather than a point estimate, of uncertainty, thus expressing our ignorance, doubt, or lack of confidence about the judgement required. We may say, for example, that the possibility of the coin turning up 'heads' lies somewhere between 60% and 40%."

In our opinion, however, the problem has not received much attention, for not only was it thought to be a simple sensibility problem inside the theory of probability, but also there were certain voices which claimed that this sensibility problem was not very important. The title of the paper by Henrion et al. [182] is meaningful enough: *Why is diagnosis using belief networks insensitive to imprecision in probabilities?*

At first, there were different apparently unrelated approaches but these have since been unified under Walley's theory of imprecise probabilities [183]. The situation is a result of the following fact: if a graphical model is a representation of the independencies of a given problem, in imprecise probability the concept of independence is not unique (see Couso et al. [184] for a survey of the different alternative definitions). There can therefore be different interpretations of a graphical model with imprecise probability. Under Walley's theory, the different situations are being better understood and categorized.

The most studied situation is the one corresponding to separately specified credal sets under *strong independence* [179]. The computational problem is more complex than in the case of precise probability, with the most promising approach being the one based on the branch-and-bound technique [185]. There are, however, other important situations, such as the one arising when probabilities are estimated from a sample with the *imprecise Dirichlet model* [186]. In this model, we have interval probabilities for each event, with the length of the interval being inversely proportional to the sample size. This estimation has very good theoretical properties, but the computational

approach is even more difficult than with separately specified credal sets. The only existing algorithm is the one proposed by Zaffalon [187] for the Naive credal classifier.

One of the main differences of imprecise models against precise ones is that they do not always propose a linear ordering of the possible options. For example, in a supervised classification problem, the model will produce a *credal classifier* [188], which under a set of observations will propose a set of possible values (the non-dominated options) for the class variable, instead of a single one. The number of options will decrease as a function of the quality of available information (for example, when probabilities are estimated from a sample, when its size increases). In an extreme situation of complete ignorance, the credal classifier will not discard any possible option proposing all the values of the class variable. The imprecise model can assert: '*there is not enough information to decide*'.

8. Conclusions

Probabilistic graphical models are currently one of the most important tools for solving real problems in artificial intelligence. There are several reasons for this success: they provide a precise language for model specification; these models are based on well-founded statistics and probability theory; the language is very general and provides a common framework in which to integrate some particular models that were formerly studied in an isolated way as Markov decision processes, influence diagrams, Kalman filters, etc.; there are methods for model learning from sets of data and efficient inference procedures; the models are not simple black boxes and they can be interpreted in terms of independent relationships between the variables; it is possible to integrate heterogeneous knowledge (numerical and qualitative) and from different sources (for example learning algorithms in genomics can integrate biological information); there are methods to cope with missing data even including variables which are never observed, etc.

There are several lessons which can be learned from the history of probability in artificial intelligence. Perhaps the most important one is that it is worth investing time trying to understand the problems we are trying to solve, thinking of well-founded theories for them. Finding fast solutions is almost never a good idea (specially for hard problems). It is also important not to concentrate only on theoretical work as we may run the risk of inventing artificial problems in order to continue on with theoretical research. In the field of probabilistic graphical models, we have important theoretical open problems but at the same time, there are some really challenging applications (from genomic to web-based applications) providing new topics and on enormous scales, given the quantity of variables and data involved. They will continue to inspire the creation of new models and procedures.

Acknowledgments

The authors are very grateful to the anonymous reviewers who provided some valuable and useful suggestions. This work has been supported by the Spanish Ministry of Science and Technology under projects Algra (TIN2004-06204-C03-02), TIN2005-03824, CSD2007-00018, and by the grant number IT-242-07 from the Basque Country Government.

References

[1] J. Pearl, Probabilistic Reasoning with Intelligent Systems, Morgan & Kaufman, San Mateo, 1988.
[2] J. McCarthy, M.L. Minsky, N. Rochester, C.E. Shannon, Proposal for Darmouth summer research project on artificial intelligence, 1955.
[3] E.H. Shortliffe, B.G. Buchanan, A model of inexact reasoning in medicine, Math. Biosci. 23 (1975) 351–379.
[4] B.G. Buchanan, E.H. Shortliffe, Uncertainty and evidential support, in: B.G. Buchanan, E.H. Shortliffe (Eds.), Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project, Addison-Wesley, 1983, pp. 209–232 (Chapter 10).
[5] B.G. Buchanan, E.H. Shortliffe, A model of inexact reasoning in medicine, in: B.G. Buchanan, E.H. Shortliffe (Eds.), Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project, Addison-Wesley, 1983, pp. 233–262 (Chapter 11).
[6] R.A. Miller, H.E. Pople, J.D. Myers, Internist-1, an experimental computer-based diagnostic consultant for general internal medicine, N. Eng. J. Med. 8 (1982) 468–476.
[7] J.R. Quinlan, Inferno: a cautious approach to uncertain inference, Comput. J. 26 (1983) 255–269.
[8] E. Horvitz, D. Heckerman, The inconsistent use of measures of certainty in artificial intelligence, in: L.N. Kanal, J.F. Lemmer (Eds.), Uncertainty in Artificial Intelligence, North-Holland, 1986, pp. 137–151.
[9] J. Pearl, Reverend Bayes on inference engines: a distributed hierarchical approach, AAAI, 1982, pp. 133–136.
[10] L.A. Zadeh, Fuzzy sets as a basis for a theory of possibility, Fuzzy Sets Syst. 1 (1978) 3–28.
[11] L.A. Zadeh, A theory of approximate reasoning, in: J.E. Hayes, D. Mikulich (Eds.), Machine Intelligence, vol. 9, Elsevier, Amsterdam, 1979, pp. 149–194.
[12] D. Dubois, H. Prade, Possibility Theory, Plenum Press, New York, 1988
[13] A.P. Dempster, Upper and lower probabilities induced by a multivalued mapping, Ann. Math. Stat. 38 (1967) 325–339.
[14] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, Princeton, 1976.
[15] R.T. Cox, Probability, frequency, and reasonable expectation, Am. J. Phys. 14 (1946) 1–13.
[16] D.V. Lindley, Scoring rules and the inevitability of probability (with discussion), Int. Stat. Rev. 50 (1982) 1–26.
[17] P. Cheesman, Probabilistic versus fuzzy reasoning, in: L.N. Kanal, J.F. Lemmer (Eds.), Uncertainty in Artificial Intelligence, North-Holland, 1986, pp. 85–102.
[18] C. Elkan, The paradoxical success of fuzzy logic, in: R. Fikes, W. Lehnert (Eds.), Proceedings of the Eleventh National Conference on Artificial Intelligence, AAAI Press, Menlo Park, CA, 1993, pp. 698–703.
[19] L.N. Kanal, J.F. Lemmer, Uncertainty in Artificial Intelligence, North-Holland, Amsterdam, 1986.
[20] M. Henrion, R.D. Shachter, L.N. Kanal, J.F. Lemmer (Eds.), Uncertainty in Artificial Intelligence 5, North-Holland, Amsterdam, 1990.
[21] E. Charniak, Statistical Language Learning, MIT Press, Cambridge, 1993.
[22] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, San Mateo, California, 1988.
[23] J.N. Darroch, S.L. Lauritzen, T.P. Speed, Markov fields and log-linear interaction models for contingency tables, Ann. Stat. 8 (1980) 522–539.
[24] N. Wermuth, S.L. Lauritzen, Graphical and recursive models for contingency tables, Biometrika 70 (1983) 537–552.
[25] A.P. Dawid, Conditional independence in statistical theory (with discussion), J. R. Stat. Soc. B 41 (1979) 1–31.
[26] R.A. Howard, J.E. Matheson, Influence diagrams, in: R.A. Howard, J.E. Matheson (Eds.), The Principles and Applications of Decision

Analysis, vol. II, Strategic Decisions Group, Menlo Park, CA, 1984 pp. 719–762.
[27] C. Cannings, E.A. Thompson, M.H. Skolnick, Probability functions on complex pedigrees, Adv. Appl. Prob. 10 (1978) 26–61.
[28] J. Pearl, Bayesian networks: a model of self-activated memory for evidential reasoning, Technical Report CSD-850021, R-43, UCLA Computer Science Department, June 1985.
[29] Ph. Smets, The normative representation of quantified beliefs by belief functions, Artif. Intell. 92 (1998) 229–242.
[30] R. Reiter, A logic for default reasoning, Artif. Intell. 13 (1980) 81–132.
[31] J. Doyle, A truth maintenance system, Artif. Intell. 12 (1979) 231–272.
[32] S.L. Lauritzen, D.J. Spiegelhalter, Local computation with probabilities on graphical structures and their application to expert systems, J. R. Stat. Soc. B 50 (1988) 157–224.
[33] G. Shafer, P.P. Shenoy, Local computation in hypertrees, Working Paper N. 201, School of Business, University of Kansas, Lawrence, 1988.
[34] G.F. Cooper, The computational complexity of probabilistic inference using Bayesian belief networks, Artif. Intell. 42 (1990) 393–405.
[35] A. Cano, S. Moral, A. Salmerón, Algorithms for approximate probability propagation in Bayesian networks, in: J.A. Gámez, S. Moral, A. Salmerón (Eds.), Advances in Bayesian Networks, Springer-Verlag, Berlin, 2004, pp. 77–97.
[36] S.M. Olmsted, On representing and solving decision problems, PhD thesis, Dept. Eng. -Econ. Syst., Stanford University, Palo Alto, CA, 1983.
[37] R.D. Shachter, Evaluating influence diagrams, Oper. Res. 34 (6) (1986) 871–882.
[38] G.F. Cooper, A method for using belief networks as influence diagrams, in: Proceedings of the Workshop on Uncertainty in Artificial Intelligence, Minneapolis, Minnesota, (1988), pp. 55–63.
[39] C. Chow, C. Liu, Approximating discrete probability distributions with dependence trees, IEEE Trans. Inform. Theory 14 (1968) 462–467.
[40] G. Rebane, J. Pearl, The recovery of causal polytrees from statistical data, in: Proceedings of the Third Workshop Uncertainty in AI, Seattle, (1987), pp. 222–228.
[41] P. Spirtes, C. Glymour, R. Scheines, Causation, Prediction and Search, Springer-Verlag, Berlin, 1993.
[42] E.A. Herskovits, G.F. Cooper, Kutato: an entropy-driven system for the construction of probabilistic expert systems from databases, in: P.P. Bonissone, M. Henrion, L.N. Kanal, J.F. Lemmer (Eds.), Uncertainty in Artificial Intelligence, vol. 6, North-Holland, 1991, pp. 117–125.
[43] G.F. Cooper, E.A. Herskovits, A Bayesian method for the induction of probabilistic networks from data, Mach. Learn. 9 (1992) 309–347.
[44] N. Friedman, D. Koller, Being Bayesian about network structure, in: C. Boutilier, M. Goldszmidt (Eds.), UAI '00: Proceedings of the 16th Conference in Uncertainty in Artificial Intelligence, Morgan Kaufmann, 2000, pp. 201–210.
[45] S.L. Lauritzen, N. Wermuth, Graphical models for associations between variables, some of which are qualitative and some quantitative, Ann. Stat. 17 (1989) 31–57.
[46] S.L. Lauritzen, Propagation of probabilities, means and variances in mixed graphical association models, J. Am. Stat. Assoc. 87 (1992) 1098–1108.
[47] S.L. Lauritzen, F.V. Jensen, Stable local computation with conditional Gaussian distributions, Stat. Comput. 11 (2001) 191–203.
[48] D. Heckerman, D. Geiger, Learning Bayesian networks, Technical Report MSR-TR-95-02, Microsoft Research, Redmond, WA, December 1994. URL http://citeseer.ist.psu.edu/article/heckerman95learning.html.
[49] S. Andreassen, M. Woldbye, B. Falck, S.K. Andersen, Munin—a causal probabilistic network for interpretation of electromyographic findings, in: Proceedings of the 10th International Joint Conference on Artificial Intelligence, 1987, pp. 366–372.
[50] S.K. Andersen, K.G. Olesen, F.V. Jensen, F. Jensen, Hugin—a shell for building Bayesian belief universes for expert systems, in: G. Shafer, J. Pearl (Eds.), Readings in Uncertain Reasoning, Morgan Kaufmann, 1990, pp. 332–337.
[51] J. Pearl, Causality. Models, Reasoning, and Inference, Cambridge University Press, Cambridge, 2000.
[52] G. Shafer, The Art of Causal Conjecture, The MIT Press, Cambridge, Massachusetts, 1997.
[53] T. Dean, K. Kanazawa, A model for reasoning about persistence and causation, Comput. Intell. 5 (1989) 142–150.
[54] U. Kjærulff, A computational scheme for reasoning in dynamic probabilistic networks, in: D. Dubois, M.P. Wellman, B.D. Ambrosio, Ph. Smets (Eds.), Proceedings of the Eighth Conference in Uncertainty in Artificial Intelligence, Morgan Kaufmann, 1992, pp. 121–129.
[55] D. Koller, A. Pfeffer, Object oriented Bayesian networks, in: Proceedings of the Thirteenth Conference of Uncertainty in Artificial Intelligence, Morgan Kaufmann, 1997, pp. 302–313.
[56] R. Kindermann, J.L. Snell, Markov Random Fields and Their Applications, American Mathematical Society, 1980.
[57] J. Pearl, T.S. Verma, The logic of representing dependencies by directed graphs, AAAI, 1987, pp. 374–379.
[58] P. Larrañaga, J.A. Lozano, J.M. Peña, I. Inza, Editorial of the special issue on probabilistic graphical models in classification, Mach. Learn. 59 (2005) 211–212.
[59] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, 1973.
[60] M. Minsky, Steps toward artificial intelligence, Trans. Inst. Radio Eng. 49 (1961) 8–30.
[61] P. Domingos, M.J. Pazzani, On the optimality of the simple Bayesian classifier under zero-one loss, Mach. Learn. 29 (2–3) (1997) 103–130.
[62] D.J. Hand, K. You, Idiot's Bayes—not so stupid after all? Int. Stat. Rev. 69 (2001) 385–398.
[63] M. Peot, Geometric implications of the naive Bayes assumption, in: Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence, 1996, pp. 414–419.
[64] M. Ramoni, P. Sebastiani, Robust Bayes classifiers, Artif. Intell. 125 (2001) 209–226.
[65] V. Robles, P. Larrañaga, J.M. Peña, E. Menasalvas, M.S. Pérez, Interval estimation naïve Bayes, in: Lecture Notes in Computer Science, vol. 2810, Springer-Verlag, 2003, pp. 143–154.
[66] J. Gama, G. Castillo, Adaptive Bayes, in: Lecture Notes in Artificial Intelligence, vol. 2527, Springer-Verlag, 2002, pp. 765–774.
[67] L.E. Sucar, D.F. Gillies, D.A. Gillies, Objective probabilities in expert systems, Artif. Intell. 61 (1993) 187–208.
[68] P. Langley, S. Sage, Induction of selective Bayesian classifiers, in: Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence, 1994, pp. 399–406.
[69] D. Dash, G.F. Cooper, Exact model averaging with naïve Bayesian classifiers, in: Proceedings of the 19th International Conference on Machine Learning, 2002, pp. 91–98.
[70] M.J. Pazzani, Searching for dependencies in Bayesian classifiers, in: D. Fisher, H. Lenz (Eds.), Artificial Intelligence and Statistics V, Lecture Notes in Statistics, Springer-Verlag, 1996, pp. 239–248.
[71] N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifiers, Mach. Learn. 29 (2) (1997) 131–164.
[72] E.J. Keogh, M.J. Pazzani, Learning augmented Bayesian classifiers: a comparison of distributions-based and classification-based approaches, in: Uncertainty 99: The 7th International Workshop on Artificial Intelligence and Statistics, 1999, 225–230.
[73] F. Pernkopf, P. O'Leary, Floating search algorithms for structure learning of Bayesian network classifiers, Pattern Recogn. Lett. 24 (15) (2003) 2839–2848.
[74] P.J.F. Lucas, Restricted Bayesian network structure learning, in: Advances in Bayesian Networks, Springer-Verlag, 2004, pp. 217–234.
[75] E. Fagiuoli, M. Zaffalon, Tree-augmented naïve credal classifiers, in: Proceedings of the 8th Information Processing and Management of Uncertainty in Knowledge-Based Systems Conference, 2000, pp. 1320–1327.
[76] J. Cerquides, R. López de Mántaras, Tractable Bayesian learning of tree augmented naive Bayes classifier, in: Proceedings of the 20th International Conference on Machine Learning, 2003, pp. 75–82.
[77] J. Roure, Incremental learning of tree augmented naïve Bayes classifiers, in: Advances in Artificial Intelligence—Proceedings of the 8th Ibero-American Conference on AI, vol. 2527 of Lecture Notes in Computer Science, Springer-Verlag, 2002, pp. 32–41.

[78] M. Sahami, Learning limited dependence Bayesian classifiers, in: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, 1996, pp. 335–338.
[79] P. Kontkanen, P. Myllymäki, T. Silander, H. Tirri, P. Grünwald, On predictive distributions and Bayesian networks, Stat. Comput. 10 (2000) 39–54.
[80] B. Sierra, P. Larrañaga, Predicting the survival in malignant skin melanoma using Bayesian networks automatically induced by genetic algorithms. An empirical comparison between different approaches, Artif. Intell. Med. 14 (1–2) (1998) 215–230.
[81] R. Greiner, W. Zhou, X. Su, B. Shen, Structural extension to logistic regression: Discriminative parameter learning of belief net classifiers, Mach. Learn. (2005) 59.
[82] A.Y. Ng, M.I. Jordan, On discriminative vs. generative classifiers, in: Proceedings of the Sixteenth Advances in Neural Information Processing Systems, vol. 14, 2001.
[83] T. Roos, H. Wetting, P. Grünwald, P. Myllymäki, H. Tirri, On discriminative Bayesian network classifiers and logistic regression, Mach. Learn. (2005) 59.
[84] M.R. Anderberg, Cluster Analysis for Applications, Academic Press, 1973.
[85] E. Forgy, Cluster analysis of multivariate data: efficiency vs. interpretability of classifications, Biometrics 21 (1965) 768–769.
[86] J.B. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability I, University of California Press, 1967, pp. 281–297.
[87] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. B 39 (1) (1977) 1–38.
[88] G.J. McLachlan, T. Krishnan, The EM Algorithm and Extensions, John Wiley and Sons, 1997.
[89] P. Cheeseman, J. Stutz, Bayesian classification (AutoClass): theory and results, in: Advances in Knowledge Discovery and Data Mining, AAAI Press, 1995, pp. 153–180.
[90] R. Kass, A.E. Raftery, Bayes factors, J. Am. Stat. Assoc. 90 (1995) 773–795.
[91] D.M. Chickering, D. Heckerman, Efficient approximations for the marginal likelihood of Bayesian networks with hidden variables, Mach. Learn. 29 (1997) 181–212.
[92] S.L. Lauritzen, The EM algorithm for graphical association models with missing data, Comput. Stat. Data Anal. 19 (1995) 191–201.
[93] N. Friedman, The Bayesian structural EM algorithm, in: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers, 1998, pp. 129–138.
[94] M. Meilă, M.I. Jordan, Estimating dependency structure as a hidden variable, Neural Inform. Process. Syst. (1998) 584–590.
[95] J.M. Peña, J.A. Lozano, P. Larrañaga, Learning Bayesian networks for clustering by means of constructive induction, Pattern Recogn. Lett. 20 (11–13) (1999) 1219–1230.
[96] J.M. Peña, J.A. Lozano, P. Larrañaga, An improved Bayesian structural EM algorithm for learning Bayesian networks for clustering, Pattern Recogn. Lett. 21 (8) (2000) 779–786.
[97] J.M. Peña, J.A. Lozano, P. Larrañaga, Learning recursive Bayesian multinets for data clustering by means of constructive induction, Mach. Learn. 47 (2002) 63–89.
[98] B. Thiesson, C. Meek, D.M. Chickering, D. Heckerman, Learning mixtures of DAG models, in: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers, 1998, pp. 504–513.
[99] J.A. Gámez, Abductive inference in Bayesian networks: a review, in: J.A. Gámez, S. Moral, A. Salmerón (Eds.), Advances in Bayesian Networks, Springer-Verlag, 2004, pp. 101–120.
[100] S.E. Shimony, Finding MAPs for belief networks is NP-hard, Artif. Intell. 68 (1994) 399–410.
[101] A.P. Dawid, Applications of a general propagation algorithm for probabilistic expert systems, Stat. Comput. 2 (1992) 25–36.
[102] D. Nilsson, An efficient algorithm for finding the M most probable configurations in probabilistic expert systems, Stat. Comput. 8 (1998) 159–173.
[103] S.E. Shimony, Explanation, irrelevance and statistical independence, in: Proceedings of the National Conference in Artificial Intelligence (AAAI-91), 1991, pp. 482–487.
[104] J.D. Park, A. Darwiche, Complexity results and approximation strategies for MAP explanations, J. Artif. Intell. Res. 21 (2004) 101–133.
[105] L.M. de Campos, J.A. Gámez, S. Moral, On the problem of performing exact partial abductive inference in Bayesian belief networks using junction trees, in: B. Bouchon-Meunier (Ed.), Technologies for Constructing Intelligent Systems 2: Tools, Springer-Verlag, 2002, pp. 289–302.
[106] J.D. Park, A. Darwiche, Solving MAP exactly using systematic search., in: Proceedings of the 19th Conference in Uncertainty in Artificial Intelligence (UAI-03), 2003, pp. 459–468.
[107] A. Salmerón, A. Cano, S. Moral, Importance sampling in Bayesian networks using probability trees, Comput. Stat. Data Anal. 34 (2000) 387–413.
[108] L.M. de Campos, J.A. Gámez, S. Moral, Partial abductive inference in Bayesian networks by using probability trees, in: O. Camp (Ed.), Enterprise Information Systems V, Kluwer Academic Publishers, 2004 pp. 146–154.
[109] J. Kalagnanam, M. Henrion, A comparison of decision analysis and expert rules for sequential analysis, in: Uncertainty in Artificial Intelligence, vol. 4, North-Holland, New York, 1990, pp. 271–281.
[110] D. Heckerman, J.S. Breese, K. Rommelse, Decision-theoretic troubleshooting, Commun. ACM 38 (1995) 49–56.
[111] F.V. Jensen, Bayesian Networks and Decision Graphs, Springer-Verlag, New York, 2001.
[112] C. Pralet, G. Verfaillie, T. Schiex, Composite graphical models for reasoning about uncertainties, feasibilities, and utilities, in: 7th International CP-05 Workshop on Preferences and Soft Constraints, 2005. URL http://www.laas.fr/cpralet/praletverfschiex.ps.
[113] D. Nilsson, S.L. Lauritzen, Evaluating influence diagrams using LIMIDs, in: C. Boutilier, M. Goldszmidt (Eds.), Proceedings of the 16th Conference on Uncertainty and Artificial Intelligence, Morgan Kaufmann Publishers, 2000, pp. 436–445.
[114] M.C. Horsch, D. Poole, An anytime algorithm for decision making under uncertainty, in: Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, San Francisco, 1998, pp. 246–255.
[115] A. Jenzarli, Solving influence diagrams using Gibbs sampling, in: Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics, 1995, pp. 278–284.
[116] C. Bielza, S. Ríos-Insua M. Gómez, Influence diagrams for neonatal jaundice management, in: Proceedings of the Joint European Conference on Artificial Intelligence in Medicine and Medical Decision Making, 1999, pp. 138–142.
[117] J. Charnes, P.P. Shenoy, Multi-stage Monte Carlo method for solving influence diagrams using local computation, Manage. Sci. 50 (2004) 405–418.
[118] A. Cano, M. Gómez-Olmedo, S. Moral, A forward-backward Monte Carlo method for solving influence diagrams, Int. J. Approx. Reason. 42 (2006) 119–135.
[119] M.L. Puterman, Markov Decision Processes, John Wiley, New York, 1994.
[120] C. Bielza, P.P. Shenoy, A comparison of graphical techniques for asymmetric decision problems, Manage. Sci. 45 (1999) 1552–1569.
[121] T.D. Nielsen, F.V. Jensen, Representing and solving asymmetric Bayesian decision problems, in: C. Boutilier, M. Goldszmidt (Eds.), Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence, 2000, pp. 416–425.
[122] M. Gómez, A. Cano, Applying numerical trees to evaluate asymmetric decision problems., in: T.D. Nielsen, N.L. Zhang (Eds.), Proceedings of the 7th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 2003, pp. 196–207.
[123] F. Jensen, T.D. Nielsen, P.P. Shenoy, Sequential influence diagrams: a unified asymmetry framework, in: Proceedings of the Second European Workshop on Probabilistic Graphical Models, 2004, pp. 121–128.
[124] F.V. Jensen, M. Vomlelova, Unconstrained influence diagrams, in: Eighteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, 2002, pp. 234–241.

[125] D. Koller, B. Milch, Multi-agent influence diagrams for representing and solving games, Games Econ. Behav. 45 (1) (2003) 181–221 (full version of paper in IJCAI '03).
[126] P.A. Bosman, D. Thierens, Linkage information processing in distribution estimation algorithms, in: W. Banzhaf, J. Daida, A.E. Eiben, M.H. Garzon, V. Honavar, M. Jakiela, R.E. Smith (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference GECCO-1999, vol. I, Morgan Kaufmann, San Francisco, CA, 1999, pp. 60–67.
[127] P. Larrañaga, J.A. Lozano (Eds.), Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation, Kluwer Academic Publishers, Boston/Dordrecht/London, 2002.
[128] J.A. Lozano, P. Larrañaga, I. Inza, E. Bengoetxea (Eds.), Towards a New Evolutionary Computation: Advances on Estimation of Distribution Algorithms, Springer-Verlag, 2006.
[129] H. Mühlenbein, G. Paaß, From recombination of genes to the estimation of distributions I. Binary parameters, in: H.-M. Voigt, W. Ebeling, I. Rechenberg, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature—PPSN IV, Springer-Verlag, Berlin, 1996, , pp. 178–187, LNCS 1141.
[130] M. Pelikan, Hierarchical Bayesian Optimization Algorithm. Toward a New Generation of Evolutionary Algorithms. Studies in Fuzziness and Soft Computing, Springer, 2005.
[131] S. Baluja, Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning, Technical Report CMU-CS-94-163, Carnegie Mellon University, Pittsburgh, PA, 1994.
[132] G.R. Harik, F.G. Lobo, D.E. Goldberg, The compact genetic algorithm, IEEE Trans. Evol. Comput. 3 (4) (1999) 287–297.
[133] H. Mühlenbein, T. Mahnig, A. Ochoa, Schemata, distributions and graphical models in evolutionary optimization, J. Heuristics 5 (2) (1999) 213–247.
[134] J.S. De Bonet, C.L. Isbell, P. Viola, MIMIC: Finding optima by estimating probability densities, in: M.C. Mozer, M.I. Jordan, T. Petsche (Eds.), Advances in Neural Information Processing Systems, vol. 9, The MIT Press, Cambridge, 1997, pp. 424–430.
[135] G. Harik, Linkage learning via probabilistic modeling in the EcGA, IlliGAL Report 99010, University of Illinois at Urbana-Champaign, Illinois Genetic Algorithms Laboratory, Urbana, IL, 1999.
[136] P.A. Bosman, D. Thierens, Multi-objective optimization with diversity preserving mixture-based iterated density estimation evolutionary algorithms, Int. J. Approx. Reason. 31 (3) (2002) 259–289.
[137] R. Etxeberria, P. Larrañaga, Global optimization using Bayesian networks, in: Proceedings of the Second Symposium on Artificial Intelligence (CIMAF-99), Habana, Cuba, (1999), pp. 151–173.
[138] H. Mühlenbein, T. Mahnig, Evolutionary synthesis of Bayesian networks for optimization, in: M. Patel, V. Honavar, K. Balakrishnan (Eds.), Advances in Evolutionary Synthesis of Intelligent Agents, MIT Press, Cambridge, MA, 2001, pp. 429–455.
[139] A. Ochoa, H. Mühlenbein, M. Soto, Factorized distribution algorithms using Bayesian networks bounded complexity, in: Proceedings of the Genetic and Evolutionary Computation Conference GECCO-2000, 2000, pp. 212–215.
[140] A. Ochoa, H. Mühlenbein, M.R. Soto, A Factorized Distribution Algorithm using single connected Bayesian networks., LNCS 1917, in: M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J.J. Merelo, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature - PPSN VI 6th International Conference, Springer-Verlag, (2000), pp. 787–796.
[141] M. Pelikan, D.E. Goldberg, E. Cantú-Paz, BOA: The Bayesian optimization algorithm, in: W. Banzhaf, J. Daida, A.E. Eiben, M.H. Garzon, V. Honavar, M. Jakiela, R.E. Smith (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference GECCO-1999, vol. I, Morgan Kaufmann Publishers, San Francisco, CA, (1999), pp. 525–532.
[142] R. Santana, E. Ponce de León, A. Ochoa, The edge incident model, in: Proceedings of the Second Symposium on Artificial Intelligence (CIMAF-99), 1999, pp. 352–359.
[143] K.B. Korb, A.E. Nicholson, Bayesian Artificial Intelligence, CRC Press, 2003.
[144] O. Pourret, P. Naïm, B. Marcot (Eds.), Bayesian Belief Networks: A Practical Guide to Applications, John Wiley, 2008.
[145] W. Edwards, Influence diagrams, Bayesian imperialism, and the Collins case: an appeal to reason, Cardozo Law Rev. 13 (1991) 1025–1074.
[146] D.A. Schum, Evidential Foundations of Probabilistic Reasoning, John Wiley and Sons, 1994.
[147] T.S. Lewit, B.K. Laskey, Computational inference for evidential reasoning in support of judicial proof, Cardozo Law Rev. 22 (2001) 1691–1731.
[148] P. Thagart, Why wasnt O.J. convicted? emotional coherence and legal inference, Cogn. Emotion 17 (2003) 361–383.
[149] A.P. Dawid, J. Mortera, V.L. Pascali, D. van Boxel, Probabilistic expert systems for forensic inference from genetic markers, Scand. J. Stat. 29 (2002) 577–595.
[150] J. Mortera, A.P. Dawid, S.L. Lauritzen, Probabilistic expert systems for DNA mixture profiling, Theor. Popul. Biol. 63 (2003) 191–205.
[151] A.P. Dawid, An object-oriented Bayesian network for estimating mutation rates, in: C.M. Bishop, B.J. Frey (Eds.), Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, 2003. URL http://research.microsoft.com/conferences/aistats2003/proceedings/188.pdf.
[152] I.W. Evett, B.S. Weir, Interpreting DNA Evidence, Sinauer Associates Incorporated, 1998.
[153] W.C. Thompson, F. Taroni, C.G.G. Aitken, How the probability of a false positive affects the value of DNA evidence, J. Forensic Sci. 48 (2003) 1357–1360.
[154] F. Taroni, C. Aitken, P. Garbolino, A. Biedermann, Bayesian Networks and Probabilistic Inference in Forensic Science, Wiley, 2006.
[155] N. Friedman, M. Linial, I. Nachman, D. Pe'er, Using Bayesian network to analyze expression data, J. Comput. Biol. 7 (2000) 601–620.
[156] C. Yoo, V. Thorsson, G.F. Cooper, Discovery of causal relationships in a gene-regulation pathway from a mixture of experimental and observational microarray data, in: Proceedings of the Pac. Symp. Biocomput., 2002, pp. 498–509.
[157] N. Friedman, Inferring cellular networks using probabilistic graphical models, Science 303 (2004) 799–805.
[158] E.P. Xing, M.I. Jordan, R.M. Karp, Feature selection for high-dimensional genomic microarray data, in: Proceedings of the 18th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, CA, 2001, pp. 601–608.
[159] E. Segal, B. Taskar, A. Gasch, N. Friedman, D. Koller, Rich probabilistic models for gene expression, Bioinformatics 17 (2001) 243–252.
[160] P. Xing, Probabilistic graphical models and algorithms for genomic analysis, PhD thesis, University of California, Berkeley, 2004.
[161] P. Sebastiani, M.F. Ramoni, V. Nolan, C.T. Baldwin, M.H. Steinberg, Genetic dissection and prognostic modeling of overt stroke in sickle cell anemia, Nat. Genet. 37 (2005) 435–440.
[162] P. Sebastiani, M. Abad, M. Ramoni, Bayesian networks for genomic analysis, in: Genomic Signal Processing and Statistics, EURASIP Book Series on Signal Processing and Communications, 2004, pp. 281–320.
[163] Elvira Consortium, Elvira: an environment for probabilistic graphical models, in: J.A. Gámez, A. Salmerón (Eds.) First European Workshop in Probabilistic Graphical Models, 2002, pp. 222–230.
[164] J. Cheng, R. Greiner, Learning Bayesian belief networks classifiers: algorithms and systems. Lectures Notes in Computer Science, Springer, 2001, pp. 141–151.
[165] K. Murphy, The Bayes net toolbox for Matlab, Comput. Sci. Stat. 33 (2001) 331–350.
[166] D.J. Spiedelhalter, A. Thomas, N.G. Best, W. Gilks, Bugs: Bayesian inference using Gibbs sampling, Technical Report, MRC Biostatistics Unit, Cambridge, 1996.
[167] S.L. Lauritzen, Graphical models in R, R. News 3 (2) (2002) 39.
[168] F.G. Cozman, The Javabayes system, ISBA Bull. 7 (4) (2001) 16–21.
[169] R. Scheines, P. Spirtes, C. Glymour, C. Meek, TETRAD II: Tools for Discovery, Lawrence Erlbaum Associates, 1994.
[170] BayesiaLab, Bayesia Home Page. URL address http://www.bayesia.com/.
[171] Netica, Norsys Software Corp. Home Page. URL address http://www.norsys.com/netica.html.

[172] R.G. Cowell, Finex: a probabilistic expert system for forensic identification, Forensic Sci. Int. 134 (2003) 196–206.
[173] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[174] P. Clifford, Markov random fields in statistics, in: G.R. Grimmett, D.J.A. Welsh (Eds.), Disorder in Physical Systems. A Volume in Honour of Hohn M. Hammersley, Oxford University Press, 1990, pp. 19–32.
[175] B.J. Frey, Graphical Models for Machine Learning and Digital Communication, MIT Press, 1998.
[176] F.R. Kschischang, B.J. Frey, H.A. Loeliger, Factor graphs and the sum-product algorithm, IEEE Trans. Inform. Theory 47 (2) (2001) 498–519.
[177] R. Kikuchi, A theory of cooperative phenomena, Phys. Rev. 81 (6) (1951) 988–1003.
[178] R. Santana, Estimation of distribution algorithms with Kikuchi approximations, Evol. Comput. 13 (1) (2005) 67–97.
[179] F.G. Cozman, Credal networks, Artif. Intell. 120 (2000) 199–233.
[180] K.W. Fertig, J.S. Breese, Interval influence diagrams, in: M. Henrion, R.D. Shachter, L.N. Kanal, J.F. Lemmer (Eds.), Uncertainty in Artificial Intelligence, vol. 5, North-Holland, Amsterdam, 1990, pp. 149–161.
[181] J. Pearl, On probability intervals, Int. J. Approx. Reason. 2 (1988) 211–216.
[182] M. Henrion, M. Pradhan, B. Del Favero, K. Huang, G. Provan, P. O'Rorke, Why is diagnosis using belief networks insensitive to imprecision in probabilities? in: E. Horvitz, F. Jensen (Eds.), Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence, 1996, pp. 307–314.
[183] P. Walley, Statistical Reasoning with Imprecise Probabilities, Chapman and Hall, London, 1991.
[184] I. Couso, S. Moral, P. Walley, A survey of concepts of independence for imprecise probabilities, Risk Decis. Policy 5 (2000) 165–181.
[185] J.C.F. Rocha, F.G. Cozman, Inference in credal networks with branch-and-bound algorithms, in: Proceedings of the Third International Symposium on Imprecise Probabilities and their Applications (ISIPTA03), 2003, pp. 482–495.
[186] P. Walley, Inferences from multinomial data: learning about a bag of marbles (with discussion), J. R. Stat. Soc. B 58 (1996) 3–57.
[187] M. Zaffalon, Statistical inference of the naive credal classifier, in: G. de Cooman, T.L. Fine, T. Seidenfeld (Eds.), Proceedings of the Second International Symposium on Imprecise Probabilities and Their Applications, Shaker Publishing, 2001, pp. 384–393.
[188] M. Zaffalon, The naive credal classifier, J. Stat. Plan. Infer. 105 (2002) 5–21.