Canonical Probabilistic Models for Knowledge Engineering

Francisco J. Díez FJDIEZ@DIA.UNED.ES
Dept. Inteligencia Artificial, UNED
Juan del Rosal, 16, 28040 Madrid, Spain

Marek J. Druzdzel MAREK@SIS.PITT.EDU
Decision Systems Laboratory, School of Information Sciences and Intelligent Systems Program
University of Pittsburgh, Pittsburgh, PA 15260, USA

Abstract

The hardest task in knowledge engineering for probabilistic graphical models, such as Bayesian networks and influence diagrams, is obtaining their numerical parameters. Models based on acyclic directed graphs and composed of discrete variables, currently most common in practice, require for every variable a number of parameters that is exponential in the number of its parents in the graph, which makes elicitation from experts or learning from databases a daunting task. In this paper, we review the so called *canonical models*, whose main advantage is that they require much fewer parameters. We propose a general framework for them, based on three categories: deterministic models, ICI models, and simple canonical models. ICI models rely on the concept of *independence of causal influence* and can be subdivided into noisy and leaky. We then analyze the most common families of canonical models (the OR/MAX, the AND/MIN, and the noisy XOR), generalizing them and offering criteria for applying them in practice. We also briefly review temporal canonical models.

Contents

1. Introduction

Over the last two decades, probabilistic graphical models have become a popular tool for modeling uncertain domains. Their most prominent representatives, Bayesian networks [67] and influence diagrams [41], have found a variety of practical applications in domains such as medicine, machine diagnosis, vision, robotics, and many others. While probabilistic graphs have reduced the complexity of the representation of the joint probability distribution, Bayesian updating even in the simplest, discrete acyclic directed graphs has been shown to be NP-hard [6]. Still, the hardest part of practical fielding of this methodology turns out not to be its computational complexity but rather building sizeable practical models.

Construction of graphical probabilistic models, such as Bayesian networks and influence diagrams, requires specification of many conditional probability distributions of the form $P(y|\mathbf{x})$, where $\mathbf{X} = \{X_1, \ldots, X_n\}$ is the set of parents of a node Y in the network—see Figure 1. Most graphical models built nowadays use only discrete variables, quite likely due to the scarcity of flexible modeling tools and algorithms for Bayesian updating in the general case. Discrete joint probability distributions are usually given in the form of conditional probability tables (CPTs), which consist of a collection of discrete probability distributions of a variable conditional on its parents in the underlying directed graph. The size of CPTs of a variable Y grows exponentially with the number of parents of Y. In general, the numerical parameters are obtained from databases or assessed by human experts and, for this reason, it is usually difficult to build a CPT for a family having more than a three or four parents. In case of a database, the difficulty arises when certain configurations of the parents are not represented in the database. For instance, when $\mathbf{X}$ represents the set of diseases that may cause a certain anomaly Y, and $\mathbf{x}$ is a particular configuration corresponding to the presence of several infrequent diseases, it is quite likely that the database contains no patient for that configuration. In case the parameters are elicited from experts, the task of estimating the probability of infrequent configurations is even more daunting, because the expert may have never seen such combinations. Additional difficulties arise when the number of probabilities to be estimated is high, because of the limited time available for interaction with experts. The problem is best illustrated by the experience of one of our colleagues, who built a sizeable Bayesian network model for medical diagnosis. She reported a remark that her expert jokingly made: "Every time you come I have a headache when you leave," because of the quantity of numerical probability elicitations that she asked of the expert.[1]

One way of reducing the complexity of elicitation of numerical probabilities is to rely on so-called *canonical models*, which allow for building probability distributions from a small number of parameters. The term "canonical" is used because such models are elementary units used in the construction of more complicated models [67]. In practice, each canonical model represents a probabilistic relation of the form $P(y|\mathbf{x})$, which involves a finite number of variables, $\{Y, X_1, \ldots, X_n\}$, usually called a *family*, such that node Y is called the *child* and the X_is are called *parents*. This terminology proceeds from assuming that those variables make part of a probabilistic ADG having a directed link $X_i \to Y$ for each X_i (see Figure 1). However, these canonical models can also be embedded in other probabilistic formalisms, such as Markov networks, chain graphs, or factorized Markov decision processes.

1. Personal communication with Dr. Concha Bielza, referring to the construction of the influence diagram IctNeo [31].

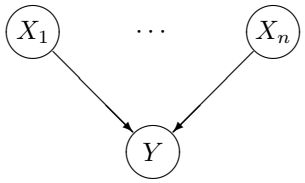


Figure 1: A family consisting of a child node Y and its parents, $Pa(Y) = \mathbf{X} = \{X_1, \ldots, X_n\}$. The qualitative information associated with this family of nodes consists of a conditional probability distribution of Y for each configuration $\mathbf{x}$: $P(y|\mathbf{x})$. If all the variables are discrete, this set of distributions can be expressed in the form of a conditional probability table (CPT).

Different canonical models may coexist in any probabilistic network. For instance, in causal Bayesian networks that model real-world domains, it is not uncommon that a significant number of families interact through OR/MAX-models, a few through AND-models and the rest of the families do not correspond to any canonical model, which implies that their CPT must be explicitly given.

Canonical models are useful not only because they simplify the construction of probabilistic models (knowledge engineering), but also because they save storage space and computation time [12, 83] and because they correspond to causal patterns that can exploited to generate user explanations [52]. Although canonical models are increasingly used in probabilistic expert systems, we believe that their advantages have not been fully explored yet. There is not enough understanding of how they work, what they express, and when they can be used. A novice knowledge engineer often has no literature guide to rely on and is left to himself or herself.

Most, although not all, variables involved in the models that we consider in this paper are discrete variables with finite numbers of values. In describing these models, we will be referring to different types of **causal interactions**. The main reason for this is clarity of our exposition: it is more natural for humans to talk about causal interactions than to talk about probabilistic interactions. This choice will greatly facilitate the task of explaining each model, defining its parameters, and offering criteria for applying them. Each model, while probabilistic rather than causal in theory, is capable of mimicking a certain pattern of behavior that we observe or assume between causes and effects in the real world. Interaction between a node and its parents, the focus of this paper, can be viewed as a causal mechanism in Simon's sense [19]. Nevertheless, it is also possible to view these models as merely probabilistic relations, and not to assign them any causal interpretation.

1.1 Overview of the paper

In this paper, we review the canonical models proposed so far in the literature, classify them in a general framework, generalize some of them, introduce new models, and offer criteria for using them in practice: what conditions or hypotheses are necessary for applying each model, how the parameters can be elicited from human experts or estimated from databases,

and how to build the probability distribution (the probability table in the case of discrete variables) given the parameters and the logical or algebraic function that define each model.

The paper begins by introducing some definitions and notation and summarizing the basic properties of Bayesian networks and their relation with causality (Sec. 2). We focus on acyclic directed graphs, such as Bayesian networks and influence diagrams, because they have been most widely applied in practice. This, we believe, is not without a reason — they have the clearest common sense interpretation and can be viewed as causal models of the underlying domains. There is a variety of probabilistic graphical models, such as Markov networks or chain graphs, but none of them has gained the popularity achieved by Bayesian networks.

Section 3 presents a general framework for canonical models based on three types. *Deterministic models* (Sec. 3.1) derive from different logical and algebraic functions and do not require any numerical parameters. *ICI models* (Sec. 3.2), which constitute most of the canonical models analyzed in this paper, are based on the assumption of *independence of causal influence*. They can be subdivided in two groups: *noisy models* (Sec. 3.2.1) and *leaky models* (Sec. 3.2.2), the latter being a generalization of the former. The proofs of the theorems that support the definition and application of leaky models are placed in Appendix A. *Simple canonical models* (Sec. 3.3) are the third kind of canonical models studied in this paper. Their main advantage is that they require fewer parameters than ICI models.

The sections that follow analyze three families of models: OR/MAX models (Sec. 4), AND/MIN models (Sec. 5), and the XOR model (Sec. 6), and give criteria for applying them in knowledge engineering. The duality of AND and OR models is discussed in Appendix C.

We also show, in Section 7, how canonical models can be extended to temporal domains. Section 8 offers bibliographical notes about the models themselves (Sec. 8.1) about how they have been applied for solving real-world problems (Sec. 8.2) and, about acquiring the numerical probabilities (Sec. 8.3). The conclusions of this paper are summarized in Section 9.

2. Preliminaries

2.1 Notation

We will use capital letters to represent variables and lower case letters to represent their values. For instance, v will represent a possible value of variable V. In the same way, $\mathbf{V}$ will denote a set of variables $\{V_1, \ldots, V_n\}$, and $\mathbf{v}$ a certain n-tuple $(v_1, \ldots, v_n)$, where v_i represents a value taken on by variable V_i. The number of variables that variable X can take on is represented by n_X.

There are binary variables of which one outcome represents the presence of something (in diagnostic problems, for example, the presence of an anomaly or disease) or a positive result in a test, and the other outcome represents the absence of the same entity or a negative result in the test. These binary variables of type *present/absent*, *positive/negative* or *true/false* are called *Boolean variables*. Their first value will be denoted by 1 or $+v$ and the second by 0 or $\neg v$. As some of the models analyzed in this paper require an ordering of the values of the variables, we define that $+v > \neg v$.

We define functions I_+ and $I_\neg$, which map configurations of Boolean variables onto subsets of indices (non-negative integers), as follows:

$$I_+(\mathbf{v}) = \{i \,|\, V_i \text{ takes the value } +v_i \text{ in } \mathbf{v}\} \tag{1}$$
$$I_\neg(\mathbf{v}) = \{i \,|\, V_i \text{ takes the value } \neg v_i \text{ in } \mathbf{v}\} \,. \tag{2}$$

For example, $I_+(+v_1, \neg v_2, +v_3) = \{1, 3\}$ and $I_\neg(+v_1, \neg v_2, +v_3) = \{2\}$. For every configuration $\mathbf{v}$, we have $I_+(\mathbf{v}) \cap I_\neg(\mathbf{v}) = \varnothing$ and $I_+(\mathbf{v}) \cup I_\neg(\mathbf{v}) = \{1, \ldots, n\}$.

Given a graph, we say that X is a *parent* of Y if there is a directed link $X \rightarrow Y$ in the graph. We will use the notation $\mathrm{Pa}(Y)$ for the parents of node Y and $\mathrm{pa}(X)$ for a configuration of them. The *ancestors* of a node are its parents and the ancestors of its parents. The definitions of *child* and *descendant* are reciprocal of the former.

2.2 Systems, models, variables, and probability distributions

Pieces of the real world that can reasonably be studied in isolation from the rest of the world, are often called *systems*. Systems can be natural (e.g., the human body) or artificial (e.g., a car engine), can be relatively simple (e.g., a pendulum) or extremely complex (e.g., the human nervous system). Although systems are always interlocked with the rest of the world, one can make a strong philosophical argument that they usually consist of strongly interconnected elements, but that their connections with the outside world are relatively weak [76]. This property allows them to be successfully studied in isolation from the rest of the world.

Abstractions of systems, used in science or everyday thinking, are often called *models*. There is a large variety in the complexity and rigor of models: there are informal mental models, simple black-box models, and large mathematical models of complex systems involving hundreds or thousands of variables. A common property of models is that they are simplification of reality. It could, in fact, be argued that models that are not simplification are useless, as they do not offer any advantage over reality. A basic component of models are *variables*, which are entities that can assume values. A model is, in fact, a specification of how individual variables are interconnected and how they interact with one another.

There is a variety of formal methods for representing models, such as logical knowledge bases, systems of simultaneous equations, or logical constraints enhanced with probability distributions. Variables can be continuous or discrete. The latter can be binary or logical or assume multiple possible values. How variables are interrelated and how the value of one variable is determined by the values of other variables is typically described by mathematical equations or functions. It is often the case that although something is known about the qualitative and statistical properties of a system's components, the exact functional form of the system's interactions is unknown. In this case, we often resort to specifying these components by means of their *joint probability distribution*, which expresses the probability of each combination of their values. The joint probability distribution allows for deriving the impact of a subset of variables on other variables through the mechanism of probabilistic conditioning. We can compute the probability distribution of any variable (or a group of variables) of interest conditional on the values assumed by other variables.

2.3 Bayesian networks and influence diagrams

Bayesian network models, details of which will be introduced in this section, represent all interactions among a system's variables by means of probability distributions and, therefore, supply a convenient way to model such cases.

Formally, Bayesian networks are acyclic directed graphs in which nodes represent random variables and arcs represent direct probabilistic dependencies among them. A Bayesian network encodes the joint probability distribution over a set of variables $\{X_1, X_2, \ldots, X_n\}$, where n is finite, and decomposes it into a product of conditional probability distributions over each variable given its parents in the graph. In case of nodes with no parents, we use their prior probability distribution. The joint probability distribution over $\{X_1, X_2, \ldots, X_n\}$ can be obtained by taking the product of all of these prior and conditional probability distributions:

$$P(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} P(x_i|\mathrm{pa}(X_i)) \ . \tag{3}$$

Figure 2 shows a highly simplified example Bayesian network modeling causes of HIV virus infection and AIDS.[2] The variables in this model are: *HIV infection* (H), sexual *Intercourse* (I), *Blood transfusion* (T), *Needle sharing* (N), *Mosquito bite* (M), and *AIDS* (A). For the sake of simplicity, we assumed that each of these variables is binary. For example, H has two outcomes, $+h$ and $\neg h$, representing "HIV infection present" and "HIV infection absent," respectively.

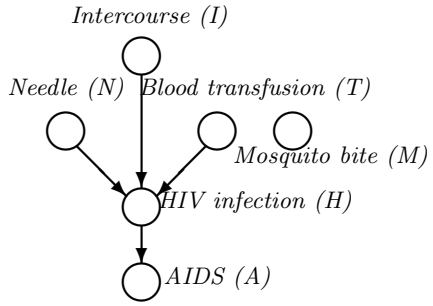


Figure 2: An example belief network for HIV infection.

A direct arc between N and H denotes the fact that whether or not an individual shares needles will impact the likelihood of her contracting the HIV virus. Similarly, an arc from H to A denotes that HIV infection influences the likelihood of developing AIDS.

Lack of directed arcs is also a way of expressing knowledge, notably assertions of (conditional) independence. For instance, lack of a directed arcs between N, I, T, and A encodes the knowledge that needle sharing, sexual intercourse, and blood transfusion can influence the chance of developing AIDS, A, only indirectly through an HIV infection, H. These causal assertions can be translated into statements of conditional independence: A is independent of N, I, and T given H. In mathematical notation,

$$P(a|h) = P(a|h,n) = P(a|h,i) = P(a|h,t) = P(a|h,n,i,t) \ .$$

2. The network is a modified version of a network presented in [20].

Structural independencies, i.e., independencies that are expressed by the structure of the network, are captured by so called Markov condition, which states that a node (here A) is independent of its non-descendants (here N, I, and T) given its parents (here H).

Similarly, the absence of arc $I \rightarrow N$ means that the individual's decision to engage in a sexual intercourse will not influence her chances of sharing needles. The absence of any links between mosquito bite M and the remainder of the variables means that M is independent of the other variables. In fact, M would typically be considered irrelevant to the problem of HIV infection and we added it to the model only for the sake of presentation.

Independence properties, such as those listed above, imply that

$$P(n, i, t, m, h, a) = P(n)\, P(i)\, P(t)\, P(m)\, P(h|n, i, t)\, P(a|h)\ ,$$

i.e., that the joint probability distribution over the graph nodes can be factored into the product of the conditional probabilities of each node given its parents in the graph. Please note that this expression is just an instance of Equation 3.

The assignment of values to observed variables is usually called *evidence*. The most important type of reasoning in a probabilistic system based on Bayesian networks is known as *belief updating* or *evidence propagation*, which amounts to computing the probability distribution over the variables of interest given the evidence. In the example model of Figure 2, the variable of interest could be H and the focus of computation could be the posterior probability distribution over H given the observed values of N, I, and T, i.e., $P(h|n, i, t)$.

Bayesian networks enhanced with nodes representing decision alternatives and utility functions are known as *influence diagrams* [41]. Influence diagrams allow for computing the expected utility of each of the decision alternatives that they model explicitly and, hence, identify the alternative that is optimal in the sense of yielding the highest expected utility. As canonical models are specified around random variables only, we will not cover influence diagrams in the remainder of this paper.

2.4 Causality and network structure

The mathematical formalism of BNs is based on factorization of the joint probability distribution of all variables in the model. Since this factorization is usually not unique, many equivalent models can be used to represent the same system. Models that represent probabilistic independencies explicitly in their graphical structure are strongly preferred. Such models minimize the number of arcs in the graph, which, in turn, increases clarity and offers computational advantages. Individual arcs can be oriented in any direction (this may have implications on the direction of other arcs in the graph) and this direction can be reversed by means of Bayes theorem.

Historically, graphical probabilistic models, such as Bayesian networks were developed to represent a subjective view of a system elicited from a decision maker or a domain expert [41]. During the elicitation process, decision makers are usually encouraged to specify variables that are directly relevant probabilistically (or causally) to a variable and influence that variable directly. These variables neighbor one another in the graph and a directed arc is drawn between them. Often, the direction of this arc reflects the direction of causal influence, as perceived by the decision maker. Sometimes, the direction of the arc reflects simply the direction in which the elicitation of conditional probabilities is easier.

There is little doubt to us that the notion of causality is critical in graphical probabilistic models. There is strong evidence that humans are not indifferent to causal relations and often give causal interpretation to conditional probabilities in the process of eliciting conditional probability distributions [80]. We have found in practice that it is very helpful to use a causal framework for modeling the interactions among the variables. There are several reasons for this. The foremost is that a causal graph is usually the easiest for an expert or a user to understand and conceptualize. It usually ensures satisfaction of the Markov condition, which ties conditional probabilistic independence with the structure of the graph. Testing for conditional independence is generally easier when the graph is causal. Henrion [40] gives an appealing practical example when a little reflection on the causal structure of the domain helps a domain expert to refine the model. Discovery of the fact that an early version of a model violates conditional independence of variables (a consequence of the Markov condition) leads the expert to realize that there is an additional intermediate node in the causal structure of the system and subsequently to refine the model. The probabilistic consequences of the causal structure, in terms of the pattern of dependences, are so strong that an expert seeking to fulfill the Markov condition, in fact, often ends up looking for the right causal model of the domain. Even those holding the strict "probabilistic influence" view admit that experts often construct influence diagrams that correspond to their causal models of the system [73]. Causal graphs also facilitate interactions among multiple experts. Causal connections and physiological mechanisms that underlie an engineering, physical, chemical, biological, or disease process are part of engineering, scientific, or medical training and provide a common language among the experts participating in the session. We have observed that experts rarely disagree about the model structure. If they do, a brief discussion of the mechanisms involved usually leads to a consensus. Finally, when the direction of arcs coincides with the direction of causality, it is usually (although not always!) easier to obtain probability judgments. For example, medical textbooks typically report conditional probabilities in the causal direction, such as the sensitivity and specificity parameters of medical tests.

The same can be said about the user interfaces to decision support systems: having a model that represents causal interactions aids in explaining the reasoning based on that model. Experiments with rule-based expert systems, such as Mycin, have indicated that diagnostic rules alone are not sufficient for generating understandable explanations and that at some level a model incorporating the causal structure of the domain is needed [4, 88]. Explanation of reasoning in Bayesian networks typically relies on causal patterns among the network variables [52].

One way of formalizing causality, due to Simon [75], is based on the concept of a causal mechanism. The notion of a mechanism can be operationalized by providing a procedure for determining whether the mechanism is present and active or not. Sometimes a mechanism is visible and tangible. One can, for example, expose the clutch of a car and even touch the plates by which the car's engine is coupled with the wheels. One can even provide an empirical demonstration of the role of this mechanism by starting the engine and depressing the clutch pedal. Often, especially in systems studied in social sciences, a mechanism is not as transparent. Instead, one often has other clues or well-developed and empirically tested theories of interactions in the system that are based on elementary laws like "no action at a distance" or "no action without communication" [77, page 52]. Mechanisms may be

identified entirely on the basis of a theory or consist of principles derived from observations, knowledge of legal and institutional rules restricting the system (such as tax schedules, prices, or pollution controls), technological knowledge, physical, chemical, or social laws. They may, alternatively, be formed on a dual basis: a theory supported by systematically collected data for the relevant variables. Causal mechanisms and causality are fundamental in human reasoning. A system is not understood until we understand the causal mechanisms that it is composed of. Because causality is so important in how humans store and organize information, it plays a fundamental role in knowledge elicitation.

3. General framework

3.1 Deterministic models

The simplest class of canonical models consists of deterministic relations among variables. In this case, the value taken on by Y is a function of the values of the X_is: $y = f(x_1, \ldots, x_n)$, and the CPT is given by

$$P(y|\mathbf{x}) = \begin{cases} 1 & \text{if } y = f(\mathbf{x}) \\ 0 & \text{otherwise .} \end{cases} \tag{4}$$

Therefore, the main advantage of the deterministic canonical models is that they do not require any numerical parameters, because the CPT in this case can be derived from the definition of function f.

Table 1 shows some examples of functions that have been proposed for canonical models and Table 2 shows the CPTs for some of the canonical models based on the logical functions defined in Table 1, assuming two parents for the OR, AND, and XOR models. Please note that those functions have the following properties:

- Logical functions apply to Boolean variables.[3] Algebraic functions apply to continuous variables and also to discrete ordinal variables, when their values are associated with succeeding integers. For instance, we may have the association {*absent*=0, *mild*=1, *moderate*=2, *severe*=3}, or {*decreased*=−1, *normal*=0, *increased*=1}. In particular, when the functions INV, MAX, MIN, and discrete-average are applied to discrete ordinal variables having all the same domain, the domain of Y is the same as that of the X_is.

- The NOT (negation), MINUS, and INV (invert) functions have only one argument, which implies that the corresponding models admit only one parent. The other functions admit two or more arguments and, consequently, the corresponding canonical models admit two or more parents.

- For all of them, except for NOT, MINUS, and INV, $f(x_1) = x_1$.

- All the functions in Table 1 are commutative (except for the linear combination when the a_is are different), which implies that the order of the parents in the corresponding canonical models is irrelevant.[4]

3. The number of Boolean functions of n arguments is 2^{2^n}; 8 of them are commutative, and 6 are both conmutative and associative. For a deeper analysis of their properties, see [25, 45, 89].
4. Sometimes commutative functions are said to be *symmetric*.

Type of function	Type of variables	Name	Definition
logical	Boolean	NOT	$y \Longleftrightarrow \neg x$
		OR	$y \Longleftrightarrow x_1 \vee \ldots \vee x_n$
		AND	$y \Longleftrightarrow x_1 \wedge \ldots \wedge x_n$
		XOR	$y \Longleftrightarrow \mathrm{card}(I_+(\mathbf{x})) = 1$
		r-out-of-n	$y \Longleftrightarrow \mathrm{card}(I_+(\mathbf{x})) = r$
		threshold	$y \Longleftrightarrow \mathrm{card}(I_+(\mathbf{x})) \geq r$
algebraic	ordinal	MINUS	$y = -x$
		INV	$y = x_{\max} - x$
		MAX	$y = \max(x_1, \ldots, x_n)$
		MIN	$y = \min(x_1, \ldots, x_n)$
		ADD	$y = x_1 + \ldots + x_n$
		average	$y = \frac{1}{n}(x_1 + \ldots + x_n)$
		discrete average	$y = \left\lceil \frac{1}{n}(x_1 + \ldots + x_n) \right\rceil$
		linear combination	$y = a_0 + a_1 x_1 + \ldots + a_n x_n$

Table 1: Some of the functions most commonly used in canonical models.

- The functions AND, OR, MAX, MIN, and ADD are associative,[5] in the sense that

$$\forall n, \forall x_1, \ldots, \forall x_n, \forall i, 1 \leq i < n, \quad f(x_1, \ldots, x_n) = f(x_1, \ldots, x_i, f(x_{i+1}, \ldots, x_n)) \; . \quad (5)$$

  The XOR (exclusive OR), exactly-r, threshold, average, and discrete average are not associative.[6] The linear combination is associative only in the particular case of an ADD function, i.e., when $a_0 = 0$ and $a_i = 1$ for $1 \leq i \leq n$ (and, of course, the trivial case in which all the a_is are zero). Leaky models are based on associative functions—see Section 3.2.2.

- When the algebraic functions are applied to Boolean variables, by representing the *present*, *positive*, or *true* value with 1 and the *absent*, *negative*, or *false* value with 0, INV becomes a NOT, MAX and discrete-average become an OR, and MIN becomes an AND. Therefore, these algebraic functions can be viewed as extensions of the logical functions to multi-valued variables.

eg because min(1,0,1) is 0 and min(1,1,1) is 1 so it gives 0 if one is 0

5. The concept of associativity is closely related to that of *decomposability* [36]. A function f is said to be *decomposable* if there exist $n-1$ two-argument functions $\{g_i\}$ such that

$$f((x_1, \ldots, x_n) = g_1(x_1, g_2(x_2, \cdots g_{n-2}(x_{n-2}, g(x_{n-1}, x_n)) \cdots)) \; ,$$

   It is usual in practice that all the gs are the same. Decomposable commutative functions are said to be *multiply decomposable*.

6. The XOR function is not associative because $f(true, true, true) = false$, while $f(true, f(true, true)) = true$. Since the XOR is a particular case of the exactly-r function, with $r = 1$, exactly-r is not associative, either. The threshold function is not associative because when $r = 2$ we have $f(true, true, false) = true$ while $f(true, f(true, false)) = false$. The average function is not associative because $f(0, 2, 2) = 4/3$ while $f(0, f(2, 2)) = 1$. The discrete average is not associative, either, because $f(2, 1, 0) = 1$, while $f(2, f(1, 0)) = 2$.

- The threshold function [85], which returns *true* when at least r of its arguments are *true*, includes as extreme cases the functions OR ($r = 1$) and AND ($r = n$). Recent research has shown that in some cases a noisy threshold model with $1 < r < n$ can be more accurate than a noisy OR or a noisy AND [45, 85].

For an in-depth analysis of the properties of Boolean functions and the canonical models based on them, see [44, 58, 86].

Function	CPT		
NOT	$P(+y\|x)$	$+x$	$\neg x$
		0	1
OR	$P(+y\|x_1, x_2)$	$+x_1$	$\neg x_1$
	$+x_2$	1	1
	$\neg x_2$	1	0
AND	$P(+y\|x_1, x_2)$	$+x_1$	$\neg x_1$
	$+x_2$	1	0
	$\neg x_2$	0	0
XOR	$P(+y\|x_1, x_2)$	$+x_1$	$\neg x_1$
	$+x_2$	0	1
	$\neg x_2$	1	0

Table 2: Conditional probability table (CPT) for some of the deterministic models induced by the logical functions in Table 1.

3.2 ICI models

Deterministic relationships are not very common in practice, as typical interactions in the real world are uncertain. In this section, we discuss the general framework for a particular kind of indeterministic models based on the assumption of *independence of causal influence* (ICI). We analyze two kind of models: noisy and leaky, the former being a particular case of the latter.

3.2.1 Noisy ICI models

Noisy models are built from deterministic models (see the above section) by introducing n auxiliary variables $\{Z_1, \ldots, Z_n\}$, as shown in Figure 3, such that Y is a deterministic function of the Z_is and the value of each Z_i depends probabilistically on X_i, as captured by

the CPT $P(z_i|x_i)$.[7] In some models, such as the noisy OR/MAX and the noisy AND/MIN, the Z_is may have a causal interpretation. However, we can just see them as auxiliary variables that are used for deriving the equations and are not part of the model. The conditional probability $P(y|\mathbf{x})$ is obtained by marginalizing out the Z_is:

$$P(y|\mathbf{x}) = \sum_{\mathbf{z}} P(y|\mathbf{z}) \cdot P(\mathbf{z}|\mathbf{x}) \ . \tag{6}$$

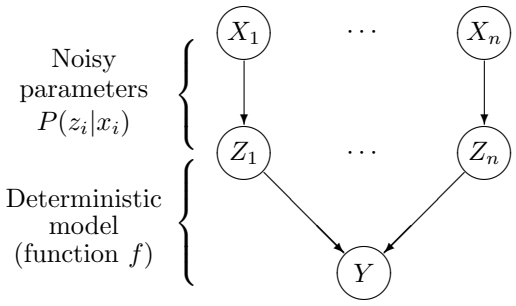


Figure 3: Auxiliary structure for the derivation of a noisy model. The assumption of *independence of causal influence* is reflected in the absence of links of the form $X_i \rightarrow Z_j$ or $Z_i \rightarrow Z_j$ with $i \neq j$.

Independence of causal influence (ICI) means that there are no interactions among the causal mechanisms (and the inhibitors) by which the X_is affect the value of Y (see also Footnote 21). Given the graph in Figure 3, this property is equivalent to the absence of links $X_i \rightarrow Z_j$ and $Z_i \rightarrow Z_j$ for all $i \neq j$, which means that

$$P(\mathbf{z}|\mathbf{x}) = \prod_i P(z_i|x_i) \ , \tag{7}$$

This factorization, together with Equations 4 and 6, leads to

$$P(y|\mathbf{x}) = \sum_{\mathbf{z}|f(\mathbf{z})=y} \prod_i P(z_i|x_i) \ . \tag{8}$$

This equation is valid for all noisy ICI models. However, we will see later that for some models, i.e., some particular fs, it can take other forms that are computationally more efficient.

To summarize, a noisy model is characterized by:

- **The domains of the variables, Y, X_is, and Z_is.** In the noisy OR, AND, and XOR models, all nodes represent Boolean variables. The MAX and MIN models only require that Y is an ordinal variable and that the domain of every Z_i is the same as that of Y.

7. The Z_is that appear in this and the subsequent models are auxiliary variables that help in explaining the models but do not make part of the resulting model, which is characterized only by the name of the function, the parameters of each link, and the leaky parameters.

- **The function f.** It is reasonable to require, following [97], that the function used in a ICI model be commutative and associative.
- **Constraints on the values of $P(z_i|x_i)$**, which follow from causal assumptions, as we shall show in Sections 4 and 5. For instance, the restrictions for the noisy OR model (Sec. 4.1.1) are

  1. $P(+z_i|\neg x_i) = 0$, and
  2. $0 < P(+z_i| + x_i) \leq 1$.

  Similar constraints apply to the AND, MAX, and MIN models. The constraint for the feeding-lines model in Section 4.3.2 is given by Equation 45. These restrictions are usually expressed in terms of *inhibitors*.[8]

Please note that each parameter $P(z_i|x_i)$ of a canonical model is associated with a particular link $X_i \rightarrow Y$, while every parameter $P(y|\mathbf{x})$ in a CPT corresponds to a certain configuration $\mathbf{x}$ made up by all the parents of Y, and cannot be associated with any particular link. This property, stemming from the ICI assumption, entails two advantages from the point of view of knowledge engineering. The first is a significant reduction in the number of parameters required to specify a model, from $O(\exp(n))$ in a general model to $O(n)$ in a canonical model. This can amount to a substantial reduction of the elicitation effort. For example, a binary node with 10 binary parents, will have a CPT consisting of $2^{11} = 2,048$ numerical parameters. Adding one more node doubles this number to $2^{12} = 4,096$ parameters. In contrast, a noisy OR model would require only 10 and 11 parameters, respectively. The second advantage is that the parameters in canonical models lend themselves to fairly intuitive interpretations, which facilities the task of eliciting them from human experts. In summary, canonical models not only require fewer parameters than ordinary CPTs, but also their parameters are more intuitive and easier to estimate.

Although the notion of ICI has been described many times in the literature, usually under the term "causal independence," the definition is not always identical. For example, Zhang and Poole [97] require that the domain of every Z_i be the same as that of Y (a condition that does not necessarily hold for the feeding-lines model in Section 4.3) and that the base function f is decomposable in terms of "a commutative and associative binary operator" (which does not hold for the XOR model in Section 6). Our concept of ICI basically reduces to the absence of links $X_i \rightarrow Z_j$ and $Z_i \rightarrow Z_j$ with $i \neq j$, i.e., to Equation 7.

3.2.2 Leaky ICI models

In practical applications, it is either not feasible or not desirable to model all variables influencing a certain node Y. In this case, we can assume that there is a large Bayesian network that properly represents the real-world domain defined over a set of variables, $\mathbf{V}'$, but we only include in our reduced model some of the variables, $\mathbf{V} \subset \mathbf{V}'$. The rest of the variables, $\mathbf{V}_I = \mathbf{V}' \backslash \mathbf{V}$, are not explicit in the model — the index I stands for "implicit." Figure 4 shows an example network with a node Y having both explicit and implicit parents. We have $\mathbf{V} = \{U_1, U_2, X_1, X_2, U_5, Y, U_7\}$. The implicit nodes, $\mathbf{V}_I$, are enclosed in the dashed pentagon.

8. Inhibitors were introduced by Pearl [67] and have also been used in [36, 40, 78].

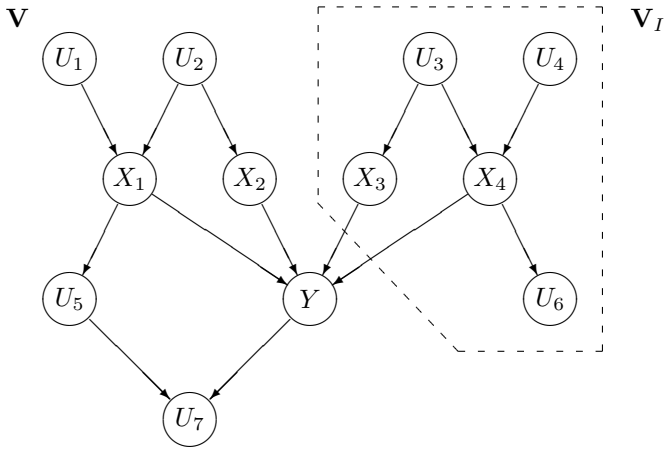


Figure 4: Using a leaky model, a large network can be simplified by removing the nodes in $\mathbf{V}_I$ (implicit nodes), as stated in Theorem 1. Furthermore, Theorem 3 states that if the CPT for Y in the large network was given by a noisy model defined on an associative function f, its CPT in the small network will be given by a leaky model defined on the same function.

Let us consider a node Y in $\mathbf{V}$ whose parents in the reduced model are $\mathbf{X} = \mathrm{Pa}(Y) \cap \mathbf{V}$. In our example, $\mathbf{X} = \{X_1, X_2\}$. The following theorem holds (the proof can be found in Appendix A).

Theorem 1 (Marginal probability for the small network) *Let $\mathcal{B}$ be a Bayesian network whose variables are $\mathbf{V}'$, let $\mathbf{V}$ be a subset of $\mathbf{V}'$ (the variables explicit in the small network), and Y a variable in $\mathbf{V}$. We introduce the following definitions:*

- $\mathbf{V}_I = \mathbf{V}' \backslash \mathbf{V}$ *(implicit variables)*
- $\mathbf{X}' = Pa(Y)$ *(parents of Y)*
- $\mathbf{X} = \mathbf{X}' \cap \mathbf{V}$ *(explicit parents of Y)*
- $\mathbf{X}_I = \mathbf{X}' \backslash \mathbf{X} = \mathbf{X}' \cap \mathbf{V}_I$ *(implicit parents of Y)*

If no node in $\mathbf{V}_I$ is a parent of any node in $\mathbf{V}$ —except for the fact that the nodes in $\mathbf{X}_I$ are parents of Y— and no node in $\mathbf{V}$ is a parent of any node in $\mathbf{V}_I$, then

$$P(\mathbf{v}) = \left(\prod_{i | V_i \in \mathbf{V} \backslash \{Y\}} P(v_i | pa(V_i)) \right) \cdot P(y|\mathbf{x}) \; . \tag{9}$$

The theorem essentially states that if no node in $\mathbf{V}$ (except for Y) has a parent in $\mathbf{V}_I$, and no node in $\mathbf{V}_I$ has a parent in $\mathbf{V}$, then it is possible to build a reduced Bayesian

network for the variables in $\mathbf{V}$. Equation 9 shows that $P(\mathbf{v})$ can be factored according to the graph of the small network, in which the parents of Y are $\mathbf{X}$. Please note that, even though it refers to the small network, all the probabilities involved are the same as those of the large network. Therefore, any marginal or conditional probability obtained from the small network is the same as the one we would obtain from the large network.

Definition 2 (Leak parent) *Under the conditions of the previous theorem, the* leak parent *is a random variable, Z_L, such that*

$$\mathrm{dom}(Z_L) = \mathrm{range}(f(\mathbf{z}_I)) \tag{10}$$

and whose probability distribution given any configuration $\mathbf{x}_I$ of $\mathbf{X}_I$ is

$$P(z_L|\mathbf{x}_I) = \sum_{\mathbf{z}_I|f(\mathbf{z}_I)=z_L} \prod_{i|X_i\in\mathbf{X}_I} P(z_i|x_i)\ . \tag{11}$$

Clearly $P(z_L|\mathbf{x}_I)$ is a probability distribution because it is non-negative for every z_L and

$$\sum_{z_L} P(z_L|\mathbf{x}_I) = \sum_{\mathbf{z}_I} \prod_{i|X_i\in\mathbf{X}_I} P(z_i|x_i) = \prod_{i|X_i\in\mathbf{X}_I} \sum_{z_i} P(z_i|x_i) = 1\ .$$

The probability $P(z_L)$ is given by

$$P(z_L) = \sum_{\mathbf{x}_I} P(z_L|\mathbf{x}_I)P(\mathbf{x}_I) = \sum_{\mathbf{z}_I|f(\mathbf{z}_I)=z_L} \sum_{\mathbf{x}_I} \left(\prod_{i|X_i\in\mathbf{X}_I} P(z_i|x_i) \right) P(\mathbf{x}_I) \tag{12}$$

and can be computed on a subnetwork containing only the implicit nodes, $\mathbf{V}_I$, and variable Z_L. The CPT of Z_L is given by a noisy model—please note that Equation 11 is just Eq. 8 applied to that subnetwork. Furthermore, given that in such computation the nodes in $\mathbf{V}_I$ that are not descendants of any node in $\mathbf{X}_I$ are barren nodes, it suffices that the subnetwork contains Z_L, the nodes in $\mathbf{X}_I$, and their ancestors. Thus, the subnetwork for the example in Figure 4 would be that in Figure 5.

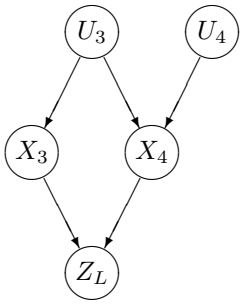


Figure 5: Subnetwork for computing Z_L from the network in Figure 4.

The next theorem, whose proof can also be found in Appendix A, offers a way for computing $P(y|\mathbf{x})$, i.e., the CPT for Y in the small network, when the CPT for Y in the large network, $P(y|\mathbf{x}')$, was given by a noisy model based on and associative function f. The resulting $P(y|\mathbf{x})$ is then said to be given by a leaky model based on function f.

Theorem 3 (Probability table for leaky models) *Under the conditions of the previous theorem, if the CPT $P(y|\mathbf{x}')$ in the large network stems from a noisy model given by parameters $P(z_i|x_i)$ and an associative function f, then $P(y|\mathbf{x})$ —obtained from the large network and used as the CPT for Y in the small network— is given by*

$$P(y|\mathbf{x}) = \sum_{\mathbf{z}} \prod_{i|X_i \in \mathbf{X}} P(z_i|x_i) \sum_{z_L | f(\mathbf{z}, z_L) = y} P(z_L) \; . \tag{13}$$

This equation can be taken as the definition of leaky models, in the same way as Equation 8 is the definition of noisy models. The parameters $P(z_i|x_i)$ in a leaky model (small network) are the same as those in the noisy model (large network), and $P(z_L)$ is a vectorial parameter, typically referred to as the "leak probability," which summarizes the influence of $\mathbf{V}_I$ on Y. It can be interpreted as the probability of a hidden parent of Y, that we have called the "leak parent," Z_L (see Figure 6). In some models, such as the leaky OR, this hidden parent may have a causal interpretation.

In principle $P(z_L)$ can be computed from the large network by applying Equation 12, as was done in [70]. However, in most practical applications, the large network is never built, and the leak probability parameter is either elicited from human experts (for instance, as in [13]) or estimated from a database (for instance, as in [65]) — see also Section 4.2.1.

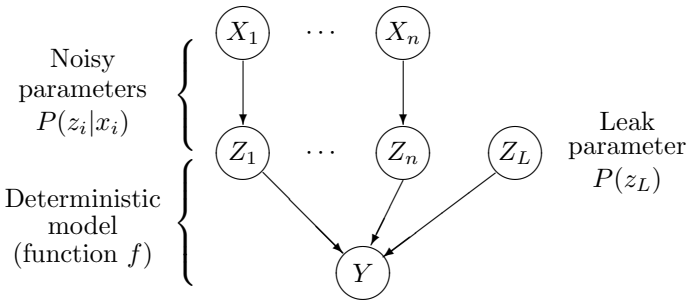


Figure 6: Internal structure of a leaky model. Variable Z_L summarizes the effect of $\mathbf{V}_I$, the parents of Y not explicitly represented in the model.

3.2.3 Probabilistic ICI models

The framework for probabilistic ICI (pICI) models is similar to that in Figure 3, the main difference being that the relation between Y and the Z_is is not deterministic, as in noisy models, but probabilistic:

$$P(y|\mathbf{z}) = f'(y, \mathbf{z}) \tag{14}$$

There are two differences between the function f used in a noisy model (see Eq. 4) and the function f' used in a pICI model. First, the domain of f is $\mathrm{dom}(\mathbf{Z})$, i.e., the set of configurations of $\mathbf{Z}$, and its range is the set of values of Y, $\mathrm{dom}(Y)$; in contrast, the domain of $f'(\mathbf{z})$ is $\mathrm{dom}(Y) \times \mathrm{dom}(\mathbf{Z})$ and its range is the interval $[0, 1]$. Second, f' must satisfy the restriction $\sum_y f'(y, \mathbf{z}) = 1$ for every configuration $\mathbf{z}$, while f is unrestricted.

Once again, the resulting model is obtained by summing out the auxiliary variables:

$$P(y|\mathbf{x}) = \sum_{\mathbf{z}} P(y|\mathbf{z})P(\mathbf{z}|\mathbf{x}) = \sum_{\mathbf{z}} f'(y,\mathbf{z}) \prod_i P(z_i|x_i) \tag{15}$$

Please note the similarity and differences of this equation with (8).

If we make $P(z_i|x_i) = \delta_{x_i,z_i}$ [Kronecker's delta] in the case of discrete variables and $P(z_i|x_i) = \delta(x_i - z_i)$ [Dirac's delta] in the case of continuous variables, we have $P(y|\mathbf{x}) = f'(y,\mathbf{x})$. As a consequence, pICI models are in principle as general as CPTs, because any CPT can be used as the f' function.

However, pICI are of interest when they are based on decomposable functions (see Footnote 5), because they can lead to faster inference and to more accurate learning from small databases. An example of them is the *pICI average model* [95, 96], based on the following function:

$$P(y|\mathbf{z}) = f'(y,\mathbf{z}) = \frac{1}{n} \text{ card}(\{Z_i \mid Z_i = y\}) \; . \tag{16}$$

3.3 Simple canonical models

ICI models need a probability table $P(z_i|x_i)$ for each link $X_i \rightarrow Y$, which means that the number of parameters necessary for building the CPT is proportional to the number of parents. This is a significant improvement with respect to the general case, which requires an exponential number of parameters. Nevertheless, in some cases there is not enough causal knowledge nor enough data to build an ICI model. In such cases, it may be advisable to apply a simple canonical model (SCM), whose internal structure, illustrated in Figure 7, contains an auxiliary variable Z such that there is a deterministic relation between the Xs and Z, given by a certain function f, and a probabilistic relation between Z and Y, given by a probability table, $P(y|z)$. The number of independent parameters is $(n_Z \; -1) \times n_Y$, regardless of the number of parents. If both Y and z are binary variables, the model only requires two parameters. In general SCMs require much fewer parameters than ICI models.

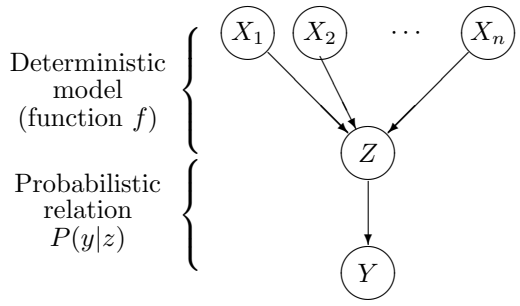


Figure 7: Internal structure of simple canonical models.

The difference between ICI models and SCMs can be clearly appreciated by comparing Figures 3 and 7: each ICI model has n intermediate variables, Z_i, while a SCM has only one. In ICI models, there a probabilistic relation between the Xs and Y, and a deterministic

relation between the Zs and Y, while in a SCM the organization of relations is just the opposite.

In Section 5.1.3, we will study the simple AND as an example of SCMs.

4. OR/MAX models

In Section 3.1, we defined the deterministic OR, MAX, and discrete-average models, and showed that the deterministic MAX and the discrete-average models reduce to the deterministic OR when applied to Boolean variables. In this section we analyze in detail the ICI versions of those models: the noisy and leaky OR/MAX, and the feeding-lines model, which is the noisy version of the discrete-average model. In all these models, the parents $\mathbf{X} = \mathrm{Pa}(Y)$ are usually interpreted as causes capable of producing Y. In the OR/MAX models, each Z_i represents the fact that X_i has produced a certain value of Y, while in the feeding-lines model Z_i represents the effective contribution of X_i to Y (see Section 4.3.2).

4.1 Definition of the OR/MAX models

4.1.1 Noisy OR

The causal interpretation of the noisy OR is that each X_i represents a cause of Y and each Z_i indicates whether X_i has produced Y. The term "noisy" refers to the possibility that some of the causes fail to produce the effect even when they are present. Then, $\neg z_i$ means that X_i has not produced Y, either because X_i was absent or because a certain inhibitor I_i has prevented X_i from producing Y. If we denote by q_i the probability that the inhibitor I_i is active, then the probability c_i that X_i produces Y when it is present is

$$c_i = P(+z_i|+x_i) = 1 - q_i \ . \tag{17}$$

In practice, we require that $c_i > 0$, because if c_i were zero, then X_i would not be a possible cause of Y and should not be included among its parents. A purely probabilistic version of this causal argument is that, according to Equation 19, $c_i = 0$ implies that $P(y|\mathbf{x})$ is the same for $+x_i$ and $\neg x_i$ and, hence, link $X_i \to Y$ is unnecessary and should be removed.

Naturally, when X_i is absent, it cannot cause Y, i.e.,

$$P(+z_i|\neg x_i) = 0 \ . \tag{18}$$

$P(z_i\|x_i)$	$+x_i$	$\neg x_i$
$+z_i$	c_i	0
$\neg z_i$	$1 - c_i$	1

Table 3: Parameters of the noisy OR for link $X_i \to Y$.

We can obtain the CPT from Equation 8 by taking into account that $f_{\mathrm{OR}}(\mathbf{z}) = \neg y$ only for the configuration $(\neg z_1, \ldots, \neg z_n)$. Therefore,

$$P(\neg y|\mathbf{x}) = \prod_{i=1}^{n} P(\neg z_i|x_i) = \prod_{i \in I+(\mathbf{x})} P(\neg z_i|+x_i) \cdot \prod_{i \in I\neg(\mathbf{x})} P(\neg z_i|\neg x_i)$$

(please recall that I_+ and $I_\neg$ were defined by Equations 1 and 2, respectively) and replacing the factors on the right-hand side with the parameters introduced in Equations 17 and 18, we arrive at

$$P(\neg y|\mathbf{x}) = \prod_{i \in I_+(\mathbf{x})} q_i = \prod_{i \in I_+(\mathbf{x})} (1 - c_i) \; . \tag{19}$$

In case of two causes X_1 and X_2 of a common effect Y, this equation leads to Table 4. Please note that when $c_1 = c_2 = 1$, this table becomes the CPT for the deterministic OR model (Table 2). In fact, the deterministic OR is a particular case of the noisy OR, in which $c_i = 1$ for all i.

$P(+y\|x_1, x_2)$	$+x_1$	$\neg x_1$
$+x_2$	$c_1 + (1 - c_1) \cdot c_2$	c_2
$\neg x_2$	c_1	0

Table 4: CPT for a noisy OR with two parents.

Equation 19 implies that if all the causes are absent, then Y is absent, i.e.,

$$P(\neg y|\neg x_1, \ldots, \neg x_n) = 1 \; . \tag{20}$$

Similarly, when X_i is present and all the other causes of Y are absent, then

$$P(+y| + x_i, \neg x_{j\,(\forall j, j \neq i)}) = c_i \; , \tag{21}$$

which is coherent with the definition of c_i as the probability that X_i causes Y (cf. Equation 17).

4.1.2 Leaky OR

As discussed in Section 3.2.2, it is generally infeasible in practice to explicitly include all possible causes of an effect, and for this reason, we need a leaky version of the OR model. Since the range of the OR function is $\{\mathit{true}, \mathit{false}\}$, variable Z_L is also a Boolean variable (see Equation 10), and the leak probability $P(z_L)$ is expressed in terms of one parameter c_L,

$$\begin{cases} P(+z_L) = c_L \\ P(\neg z_L) = 1 - c_L \; . \end{cases}$$

The CPT for Y in the leaky model can be obtained from Equation 13 by taking into account that $f_{\mathrm{OR}}(\mathbf{z}, z_L) = \neg y$ only for the configuration $(\neg z_1, \ldots, \neg z_n)$ and the value $\neg z_L$. Therefore,

$$P(\neg y|\mathbf{x}) = P(\neg z_L) \cdot \prod_{i=1}^{n} P(\neg z_i|x_i) = (1 - c_L) \cdot \prod_{i \in I_+(\mathbf{x})} (1 - c_i) \; . \tag{22}$$

Table 5 shows the form of the CPT for $n = 2$. Please note that when $c_L = 0$, this table is equivalent to the CPT for the noisy OR model (Table 4).

$P(+y\|x_1, x_2)$	$+x_1$	$\neg x_1$
$+x_2$	$1-(1-c_1)\cdot(1-c_2)\cdot(1-c_L)$	$c_2+(1-c_2)\cdot c_L$
$\neg x_2$	$c_1+(1-c_1)\cdot c_L$	c_L

Table 5: CPT for a leaky OR with two parents.

When all explicit causes are absent, the probability of Y being present is the leak probability:

$$P(+y|\neg x_1, \ldots, \neg x_n) = c_L \ , \tag{23}$$

which means that Z_L can be interpreted as the variable that indicates whether the implicit causes produced Y or not.

The noisy OR is a particular case of the leaky OR in which $c_L = 0$. In a proper leaky OR, $c_L > 0$. We also require that $c_L < 1$, because if c_L were 1 then $P(+y|\mathbf{x}) = 1$ for all $\mathbf{x}$, i.e., Y would always be present, regardless of the values assumed by the X_is. In that case, links $X_i \rightarrow Y$ would be unnecessary and should be removed.

4.1.3 Recursive noisy OR

We have seen that in the noisy OR all the values of the CPT are obtained from the c_i parameters under the assumption of independence of causal influence (ICI). The recursive noisy OR (RNOR) [56] was proposed to relax the ICI assumption by allowing that the human expert gives a value not only for the c_is, but also for some of the probabilities in which several causes are present. When the value for a certain two-cause parameter $c_{i,j}$ has not been given by the expert, it is computed from c_i and c_j as if it were a standard noisy OR:[9]

$$c_{i,j} = 1-(1-c_i)(1-c_j) = c_i + (1-c_i)\cdot c_j \ . \tag{24}$$

The value of $c_{i,j}$ obtained from this expression represents, by definition, the *lack of synergy* between X_i and X_j. When the value provided by the human expert is higher than the one obtained from this equation, we say that there is *synergy* between them. When the expert-estimated value is smaller, we say that X_i and X_j *interfere*. In any case, the axioms of the RNOR require that the expert-estimated value be higher than $\max(c_i, c_j)$.

Similarly, if the value of the three-cause parameter $c_{i,j,k}$ is not explicitly given by the expert, it is computed as follows:

$$c_{i,j,k} = 1 - \frac{(1-c_{i,j})(1-c_{j,k})(1-c_{i,k})}{(1-c_i)(1-c_j)(1-c_j)} \ . \tag{25}$$

Please note that if all the $c_{i,j}$s were computed from the c_is, then $c_{i,j,k}$ would have the same value as in a noisy OR (cf. Eq. 19). However, if some of the $c_{i,j}$s were given by the expert, the value of $c_{i,j,k}$ resulting from Equation 25 will in general be different from the its noisy OR counterpart, i.e., the one based on Equation 19.

This model is called *recursive* because the n-cause parameters, if not given by the expert, are computed from the $(n-1)$-cause and $(n-2)$-cause parameters by applying a

9. In this section, the subindices of c indicate the causes that are present. For instance, when $n = 4$ we have $c_{1,3} = P(+y| + x_1, \neg x_2, +x_3, \neg x_4)$. This notation is coherent with the interpretation of the c_is as the one-cause parameters—see Equation 21.

generalization of Equation 25, namely Equation 4 in [56]. The RNOR includes the noisy OR as a particular case because if only the one-cause parameters —the c_is— are given by the expert then the resulting CPT is the same as for the noisy OR.

A drawback of this model is that it may be asymmetric when $n > 3$. In fact, even though Equations 24 and 25 are symmetric, when c involves four or more causes, the order of the subindices becomes relevant: the value of $c_{i,j,k,l}$ is based on $c_{i,j}$, $c_{j,k}$, $c_{k,l}$, and $c_{l,i}$, while the value of $c_{i,k,j,l}$ is based on $c_{i,k}$, $c_{k,j}$, $c_{j,l}$, and $c_{l,i}$, which may be different if some of the two-cause parameters were given by the expert. No causal argument can justify this difference and the authors of this model do not give any criterion for determining the right order of the parents.

Another limitation of this model is that, as mentioned above, the axioms of the RNOR prohibit that a c parameter involving m causes be smaller than any parameter involving a subset of those causes. A violation of this requirement of monotonicity might make higher-order parameters be outside the $[0, 1]$ interval. The next model provides a remedy to this limitation.

4.1.4 Inhibited recursive noisy OR

A possibility of modelling inhibition consists in applying the *inhibited RNOR* [50], a non-ICI model that computes, on the one hand, the probability that Y is caused and, on the other, the probability that it is inhibited. The knowledge engineer will declare that some subsets of parents of Y are causes of Y, while other subsets are inhibitors of Y, each with a certain probability. For example, $c_{j,k}$ is the probability that X_j and X_k together cause Y, while $i_{j,k,l}$ is the probability that X_j, X_k, and X_l together inhibit Y.

For a configuration $\mathbf{x}$, $P_c(y|\mathbf{x})$ gives the probability that Y was caused, and $P_i(y|\mathbf{x})$ gives the probability that Y has been inhibited. Each of these CPTs is derived as a recursive noisy OR, the former based on the c parameters and the latter on the is. There is also a leak probability, c_L, representing the probability that other causes not explicit in the model produce Y. The effect is present when it has been caused and not inhibited:

$$\begin{aligned} P(+y|\mathbf{x}) &= P(Y \text{ caused by } \mathbf{x}) \cdot P(Y \text{ not inhibited by } \mathbf{x}) \\ &= [P_c(+y|\mathbf{x}) + (1 - P_c(+y|\mathbf{x})) \cdot c_L] \cdot [1 - P_i(+y|\mathbf{x})] \end{aligned}$$

The RNOR is a particular case of the inhibited recursive noisy OR having no inhibitors and no leak probability. The leaky OR is a particular case of the inhibited in which only the one-cause parameters are explicitly given, i.e., having no inhibitors and no interactions between causes.

Please note that this model contains two types of inhibitors. On the one hand, $c_i < 1$ means that there is a certain (implicit) inhibitor for cause X_i, and $c_{i,j} < 1$ implies that there is an inhibitor for the combination of X_i and X_j. Besides this implicit local inhibitors, there are other inhibitors explicitly given by the i parameters, which have a global scope, i.e., they inhibit Y independently of how it has been produced.

4.1.5 Noisy MAX

The noisy MAX model was developed with the intention to extend the noisy OR model to multi-valued variables. In the noisy MAX, each Z_i represents the value of Y produced by

X_i. The resulting value produced by the individual X_is is $y = f_{\text{MAX}}(\mathbf{z})$. Therefore, Y and the Z_is must share the same domain. Each Z_i represents the fact that X_i has raised the value of Y to a certain value, and the actual value of Y is the maximum of the Z_is.

The parameters for a link $X_i \rightarrow Y$ are

$$c_{z_i}^{x_i} = P(z_i|x_i) \tag{26}$$

or, equivalently,

$$c_{y}^{x_i} = P(Z_i = y|x_i) \tag{27}$$

Each $c_y^{x_i}$ can be understood as the probability that X_i, when taking the value x_i, raises the value of Y to y.

Please note that this noisy MAX only requires that Y is an ordinal variable. It does not impose any condition on the domains of the X_is or on the values of the $c_y^{x_i}$s. Therefore, this noisy MAX is more general than the *causal MAX* described in the next section, and in turn the latter is more general than the *graded noisy MAX* proposed in [10, 40].

CPT for the noisy MAX In order to obtain the CPT for the noisy MAX, we first compute $P(Y \leq y|\mathbf{x})$ for all values y and all configurations $\mathbf{x}$ by applying Equation 8 and taking into account that $f_{\text{MAX}}(\mathbf{z}) = \max(z_1, \ldots, z_n)$, which implies that $f_{\text{MAX}}(\mathbf{z}) \leq y$ if and only if $z_i \leq y$ for all i. Therefore,

$$P(Y \leq y|\mathbf{x}) = \sum_{\mathbf{z}|f_{\text{MAX}}(\mathbf{z})\leq y} \prod_i c_{z_i}^{x_i} = \sum_{z_1 \leq y} \cdots \sum_{z_n \leq y} \prod_i c_{z_i}^{x_i} = \prod_i \left(\sum_{z_i \leq y} c_{z_i}^{x_i} \right) . \tag{28}$$

If we define accumulative parameters,

$$C_y^{x_i} = \sum_{z_i \leq y} c_{z_i}^{x_i} , \tag{29}$$

the previous equation becomes

$$P(Y \leq y|\mathbf{x}) = \prod_i C_y^{x_i} . \tag{30}$$

Each value of the CPT can be obtained as follows:

$$P(y|\mathbf{x}) = \begin{cases} P(Y \leq y|\mathbf{x}) - P(Y \leq y-1|\mathbf{x}) & \text{for } y \neq y_{\min} \\ P(Y \leq y|\mathbf{x}) & \text{for } y = y_{\min} \end{cases} . \tag{31}$$

4.1.6 Causal noisy MAX

The causal noisy MAX is a particular case of the noisy MAX in which Y represents an anomaly and the X_is are the anomalies than may cause Y. Variable Y is a *graded variable* [10], i.e., an ordinal variable whose *neutral state*, denoted by $\neg y$, is the minimum of Y and represents the absence of anomaly, and higher states of Y represent more severe degrees of anomaly. In contrast, the X_is need not be ordinal: it suffices that each X_i has a neutral

state, $\neg x_i$, which may be different from the minimum of X_i, as shown in the example in the next section.[10]

In this ICI model, each Z_i represents the degree of Y caused by X_i. Because of the definition of neutral state,

$$c_{\neg y}^{\neg x_i} = P(Z_i = \neg y | \neg x_i) = 1 \ . \tag{32}$$

It follows from Equation 8 that anomaly Y is absent when all its parents are in their neutral states, i.e.,

$$P(\neg y | \neg x_1, \ldots, \neg x_n) = 1 \ . \tag{33}$$

It also follows from Equation 8 that

$$P(y | x_i, \neg x_{j\,(\forall j,\, j \neq i)}) = P(Z_i = y | x_i) = c_y^{x_i} \ , \tag{34}$$

which means that when X_i takes the value x_i and the other causes of Y are in their neutral states, the probability that Y takes the value y is $c_y^{x_i}$. Therefore, $c_y^{x_i}$ represents the capability of X_i to raise the state of Y to a certain value independently of the states of the other causes of Y (which may happen to raise the value of Y to a higher value).

When a X_is is ordinal, especially when it is graded, typically higher values of X_i tend to cause higher values of Y, i.e.,

$$x_i < x_i' \Longrightarrow \forall y, P(Z_i \geq y | x_i) \leq P(Z_i \geq y | x_i') \ . \tag{35}$$

(In the example in the next section X_2 is a graded variable and satisfies this condition.) This is equivalent to

$$x_i < x_i' \Longrightarrow \forall y, \sum_{z_i = y}^{y_{\max}} c_{z_i}^{x_i} \leq \sum_{z_i = y}^{y_{\max}} c_{z_i}^{x_i'} \ . \tag{36}$$

In this case, the link $X_i \rightarrow Y$ represents a positive influence, as defined in [51, 54, 90].

The causal noisy MAX defined in this section is more general than the noisy MAX introduced in [10, 39], which may be called *graded noisy MAX* because it required that all the variables were graded. The noisy OR is a particular case of the graded noisy MAX, and consequently, a particular case of the causal noisy MAX. Therefore, the similarity of Equations 20 and 33 and Equations 21 and 34 is not surprising.

4.1.7 Leaky MAX and causal leaky MAX

Following the framework exposed in Section 3.2.2, it is possible to build a leaky MAX in which the interpretation of the auxiliary variable Z_L is that some causes, not explicit in the model, have raised the value of Y to y, with probability $P(Z_L = y)$. In practice, $\mathrm{dom}(Z_L) = \mathrm{dom}(Y)$, which implies that the leaky MAX requires n_Y leak parameters, $c_y^L = P(Z_L = y)$, that can be computed from Equation 12 or estimated from databases or from human experts. The number of independent parameters is $n_Y - 1$, because their sum must be 1.

10. The term *neutral state* is synonimous the *distinguished state* proposed by Heckerman and Breese [36], who used the expression *amechanistic property* to refer to the fact that $\neg x_i$ does not take Y out of its normality state $\neg y$ (see Equations 32 and 33). It is also synonymous with the term *normality state* used in [51, 54]

In the causal version of the leaky MAX, Y and its parents satisfy the same conditions as in the causal noisy MAX, and each parameter c_y^L represents the probability that $Y = y$ when the causes explicit in the model are in their neutral states:

$$c_y^L = P(y|\neg x_1, \ldots, \neg x_n) \ , \tag{37}$$

(please note the similarity with Equation 23), in accordance with the assertion that $Z_L = y$ represents the fact that the causes of Y implicit in the model have raised its value to y.

Example: causal leaky MAX. A certain disease Y may be caused by the excess or deficiency of substance X_1 in the patient's blood. The same disease can also be due to a certain anomaly X_2. We may represent this problem by choosing the following domains for these variables:

$$\begin{aligned} \mathrm{dom}(Y) &= \{absent, mild, moderate, severe\} \\ \mathrm{dom}(X_1) &= \{decreased, normal, increased\} \\ \mathrm{dom}(X_2) &= \{absent, present\} \end{aligned}$$

Their neutral values are $\neg y$ =*absent*, $\neg x_1$ =*normal*, and $\neg x_2$ =*absent*, respectively. X_1 is not a graded variable because its minimum, "*decreased*," does not represent the absence of anomaly. Figure 8.(a) shows the parameters for this model in Elvira.[11]

Figure 8.(b) represents the CPT computed from these parameters. We can see that when X_1 and X_2 are in their neutral states (5th column), the probability of Y is the leak probability shown in Figure 8.(a), which agrees with Equation 37. ■

The noisy MAX is a special case of the leaky MAX in which $c^L_{y_{\min}} = 1$ and the other c_y^Ls are zero, which implies that $Y > y_{\min}$ only when some of the X_i (the explicit causes, we might say) has raised its value. Clearly, the causal noisy MAX is a particular case of the causal leaky MAX under the same condition: $c^L_{\neg y} = 1$ implies that Y takes its minimum value, $\neg y$, when its parent are all in their neutral states.

The leaky OR is a special case of the causal leaky MAX in which Y and their parents are all Boolean variables, $c^L_{+y} = c_L$ and $c^L_{\neg y} = 1 - c_L$.

CPT for the causal leaky MAX Similarly to the case of the noisy MAX, we will first compute $P(Y \leq y|\mathbf{x})$ for all the values y and all the configurations $\mathbf{x}$ by applying Equation 13, as follows:

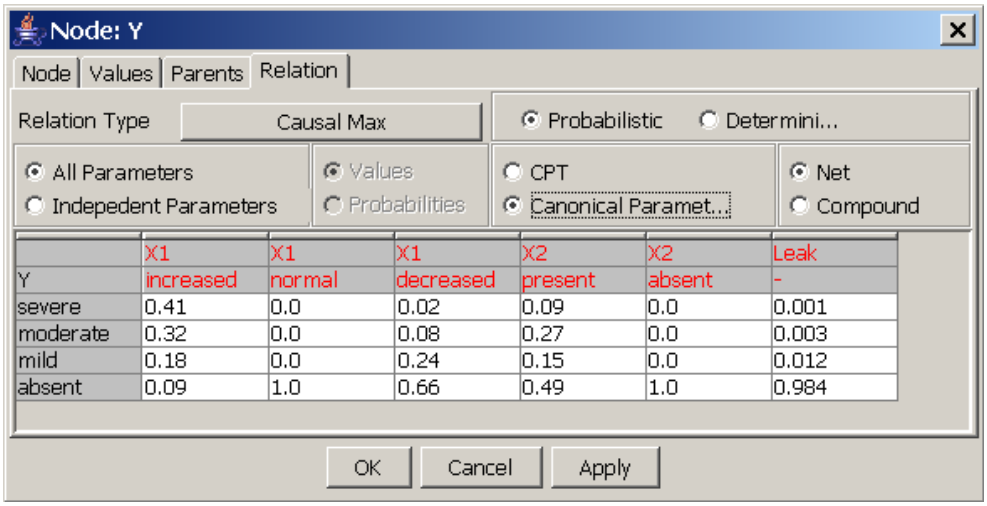
$$P(Y \leq y|\mathbf{x}) = \sum_{\mathbf{z}} \prod_{i|X_i \in \mathbf{X}} P(z_i|x_i) \sum_{z_L | f_{\mathrm{MAX}}(\mathbf{z}, z_L) \leq y} P(z_L) = \left(\sum_{z_L \leq y} c^L_{z_L} \right) \cdot \prod_i \left(\sum_{z_i \leq y} c^{x_i}_{z_i} \right) \ . \tag{38}$$

If we define an accumulative vectorial parameter,

$$C_y^L = \sum_{z_L \leq y} c^L_{z_L} \ , \tag{39}$$

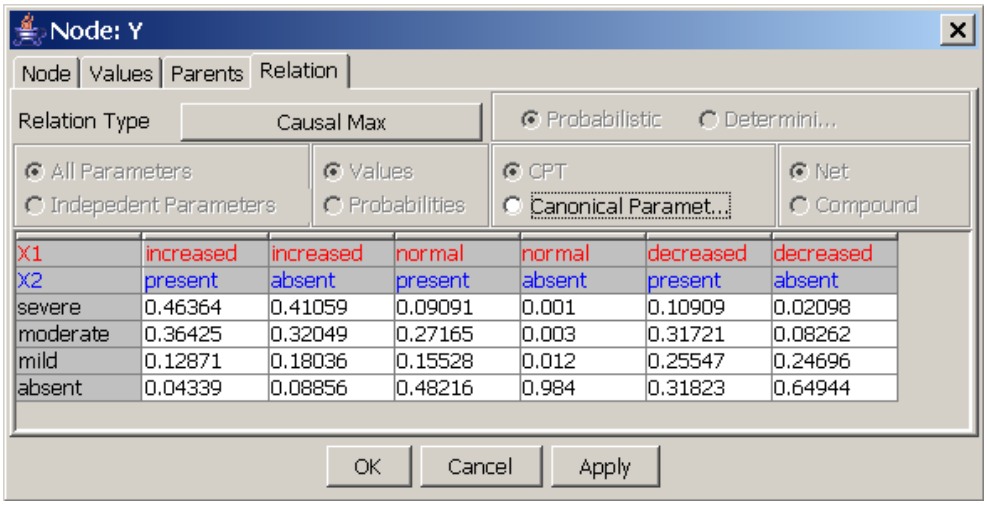
11. Elvira [24] is a public software tool for Bayesian networks and influence diagrams developed by several Spanish universities—see `http://www.ia.uned.es/~elvira`. Similar tables might be obtained with GeNIe, another software tool developed by the second author's lab at the University of Pittsburgh—see `http://genie.sis.pitt.edu`.

a)



	X1	X1	X1	X2	X2	Leak
Y	increased	normal	decreased	present	absent	-
severe	0.41	0.0	0.02	0.09	0.0	0.001
moderate	0.32	0.0	0.08	0.27	0.0	0.003
mild	0.18	0.0	0.24	0.15	0.0	0.012
absent	0.09	1.0	0.66	0.49	1.0	0.984

b)



X1	increased	increased	normal	normal	decreased	decreased
X2	present	absent	present	absent	present	absent
severe	0.46364	0.41059	0.09091	0.001	0.10909	0.02098
moderate	0.36425	0.32049	0.27165	0.003	0.31721	0.08262
mild	0.12871	0.18036	0.15528	0.012	0.25547	0.24696
absent	0.04339	0.08856	0.48216	0.984	0.31823	0.64944

Figure 8: A causal MAX in Elvira [24]. (*a*) Canonical parameters. (*b*) Conditional probability table.

the previous equation becomes

$$P(Y \leq y|\mathbf{x}) = C_y^L \cdot \prod_i C_y^{x_i} \ . \tag{40}$$

Each probability $P(y|\mathbf{x})$ can be obtained from Equation 31, as shown in the following example.

Example. According with Equations 29 and 40, the C parameters for the causal leaky MAX shown in Figure 8.(*a*) are as follows:

$C_y^{x_1}$	X_1		
	incr	*norm*	*decr*
Y=*sev*	1.00	1.00	1.00
Y=*mod*	0.59	1.00	0.98
Y=*mild*	0.27	1.00	0.90
Y=*abs*	0.09	1.00	0.66

$C_y^{x_2}$	X_2	
	pres	*abs*
Y=*sev*	1.00	1.00
Y=*mod*	0.91	1.00
Y=*mild*	0.64	1.00
Y=*abs*	0.49	1.00

C_y^L	
Y=*sev*	1.000
Y=*mod*	0.999
Y=*mild*	0.996
Y=*abs*	0.984

All the probabilities in Figure 8.(b) can be computed by applying Equations 31 and 40. For instance, the probability at the top right corner is

$$\begin{aligned}
P(Y \leq sev\,|X_1 = incr, X_2 = pres) &= 1 \times 1 \times 1 = 1 \\
P(Y \leq mod|X_1 = incr, X_2 = pres) &= 0.59 \times 0.91 \times 0.999 = 0.53636 \\
P(Y = sev\,|X_1 = incr, X_2 = pres) &= P(Y \leq sev\,|X_1 = incr, X_2 = pres) \\
&\quad - P(Y \leq mod|X_1 = incr, X_2 = pres) \\
&= 1 - 0.53636 = 0.46364\ .
\end{aligned}$$

■

4.2 Knowledge engineering for the OR/MAX models

4.2.1 Net parameters vs. compound parameters

In the noisy OR, c_i is the probability of $+y$ when X_i is present and the other X_js are absent (cf. Equation 21). However, in the leaky OR, the probability of Y when X_i is present and the other *explicit* causes are absent is not c_i, but rather p_i, where

$$p_i = P(+y|+x_i, \neg x_{j\,(\forall j,\, j\neq i)}) = c_i + c_L - c_i \cdot c_L\ . \tag{41}$$

We call c_i a net parameter because it measures the net effect of X_i, and we call p_i a compound parameter because it reflects the fact that Y may be due to either X_i, with probability c_i, or to the *implicit* causes, with probability c_L.

Since $c_L > 0$ in the proper leaky OR, we have $p_i > c_i$. We also have

$$1 - c_i = \frac{1 - p_i}{1 - c_L}\ . \tag{42}$$

Henrion's [40] definition of the leaky OR is based on the compound parameters, p_i. In fact, Henrion's equation for the probability table derived from the parameters of the noisy OR gate was

$$P(+y|\mathbf{x}) = 1 - (1 - c_L) \cdot \prod_{i \in I_+(\mathbf{x})} \frac{1 - p_i}{1 - c_L}\ , \tag{43}$$

which is equivalent to Equation 22.

Which parameters are more appropriate from the point of view of knowledge engineering? It depends on the source of knowledge. When the probabilities are obtained from a **database**, c_L can be estimated as the proportion of cases in which Y is present among

those in which all the causes of Y stored in the database are absent (cf. Equation 23). If $c_L > 0$, then there must be other causes of Y, not explicitly recorded in the database. The proportion of cases in which Y is present among those in which X_i is present and the explicit causes X_j take on the value absent is an estimate of the compound parameter p_i (cf. Equation 41). In this situation, it is impossible to estimate the net parameter c_i directly from the database, because, as mentioned above, the implicit causes are not recorded in the database and they may always be present.

In contrast, when the probabilities are estimated by **human experts**, the question that the knowledge engineer should ask in order to obtain the net parameter c_i is: "What is the probability that X_i produces Y if all other possible causes of Y are absent?," while the question for eliciting the compound parameter p_i would be: "What is the probability that Y is present when X_i is present and none of the other causes that we are considering explicitly in our model are present?" The answer to the first question can be based on an analysis of the causal mechanism $X_i \rightarrow Y$ and its possible inhibitors (cf. Section 4.1.1), while the second question calls on the "statistical" data stored in the expert's memory. Recent work on the elicitation of probabilities for a medical expert system has shown that it seems easier for human experts to give the net parameters [64]. A controlled study of this problem has confirmed this observation [93]. This is consistent with our conjecture that human estimation of probabilities relies on the knowledge of causal mechanisms and not only on observed frequencies.

In summary, the frequencies that we observe in a database correspond to Henrion's compound parameters, p_i. In this case, we recommend to convert them into the corresponding c_is (cf. Equation 42), because Equation 22 is slightly more efficient than Equation 43. In contrast, when the probabilities are estimated by a human expert, we recommend focusing directly on the net parameters, c_i, because the corresponding questions will be simpler and more intuitive. Fortunately, if c_L is small — as it typically shall be, because an accurate model must explicitly include all the frequent causes of each anomaly — the difference between each c_i and its corresponding p_i will be smaller than the error of the subjective estimate, and the knowledge engineer does not need to worry too much about which parameters she is eliciting from the expert.

4.2.2 Criteria for applying the OR model

In order to apply the OR model in practice, the knowledge engineer has to verify the following criteria:

1. Are the variables involved in the family all Boolean?

   For example, if one of the parent variables is *Sex*, the OR model cannot be applied, because it is not possible to identify one of its values (*male* or *female*) with the cause of an anomaly and the other with its absence. If *Sex* appears among the parents of a certain family, generally it behaves as a risk factor or as a precondition for another variable, and in this case the right choice may be the AND model (see Section 5).

2. Is there a causal mechanism for each parent X_i, such that X_i is able to cause Y in the absence of the other anomalies?

Please remember that in the OR model variable Z_i (Figure 3) represents the fact that the effect Y has been produced by X_i. When the effect cannot be caused by individual mechanisms, but it is necessary that some of the causes co-occur to produce the effect, the OR model cannot be applied.

3. What is the nature of the causal mechanisms involved?

   If some of these causal mechanisms are not deterministic, the noisy OR must be used instead of the deterministic OR. If there are other causes, not explicit in the model but capable of producing Y, the leaky OR must be used instead of the noisy OR.

4. Are the causal mechanisms independent?

   This is, in general, the most difficult condition to establish, since in many cases our knowledge of the domain is not precise enough to ascertain that the causal mechanisms and their inhibitors do not interact with one another. In practice, unless there are known interactions, we assume that this condition holds and the noisy/leaky OR can be applied. If this assumption does not hold, then we should use a non-ICI model, such as the above-mentioned *RNOR* or *inhibited RNOR* models.

   An interesting attempt to extend canonical models to cases in which interactions happen was made by Lemmer and Gossink [56]. They extend the plain noisy OR model by allowing the expert to specify interaction effect between chosen groups of causes and derive the CPTs from these specifications.

5. In case of a leaky OR, are the implicit causes of Y and their ancestor causes (called $\mathbf{V}_I$ in Section 3.2.2) independent of all the explicit variables in the model?

   Typically, we assume that this condition holds unless there is evidence against it.

If all the above conditions hold, we can proceed with obtaining the numerical parameters, either from a database or from an expert. In the latter case, if a cause X_i almost always produces the effect, we recommend to ask the expert for an estimate of $P(\neg z_i| + x_i) = 1 - c_i$, i.e., the probability that an inhibitor prevents X from producing Y, because in our experience it is easier for a human expert to estimate whether q_i is 0.01 or 0.001 (the latter is 10 times bigger than the former) than to assess whether c_i is 0.99 or 0.999 (the difference being less than 1%). Otherwise, we recommend to estimate $c_i = P(+z_i| + x_i)$ directly. In case of a leaky noisy OR, the knowledge engineer must decide if the knowledge elicitation will be based on the net parameters or on the compound parameters (see Section 4.2.1), although when the leak probability is small, the difference is almost irrelevant.

4.2.3 Criteria for applying the noisy MAX

In general, the criteria for using the MAX model are the same as those for the OR model.

We should note that, although in the development of practical models, application of the deterministic OR is not uncommon, we have never encountered examples in which the deterministic MAX would be appropriate, except for when the parents of Y represent different subtypes of Y. For instance, in a Bayesian network that we developed for echocardiography [13], the node *Mitral-regurgitation* took on values {*absent*, *mild*, *moderate*, *severe*} and had

two parents, *Acute-mitral-regurgitation* and *Chronic-mitral-regurgitation*, whose interaction was modeled by a deterministic MAX.

The noisy/leaky causal MAX is the candidate model when there are several causes X_i that can produce an effect Y with various degrees of severity. The conditions that must be satisfied are:

1. Is Y a graded variable, i.e., an ordinal variable whose minimum corresponds to the absence of an anomaly?

   One case in which this condition does not hold is when the values of Y are, for instance, {*decreased*, *normal*, *increased*}, because the normal value is not the minimum. Such variable could be, however, one of the parents of a noisy MAX gate, as shown in the example in Section 4.1.7.

2. Does each link $X_i \rightarrow Y$ corresponds to a distinct causal mechanism?

3. Are the mechanisms causally independent of each other, or are there interactions among them?

   If this condition did not hold, i.e., if there were interactions among the causal mechanisms, we should use a non-ICI model similar to the *RNOR* or the *inhibitory RNOR* capable of dealing with non-binary variables. Unfortunately, to our knowledge no such model has been developed up to date. Fortunately, in our experience in building probabilistic graphical models we have never needed them.

When the above criteria are satisfied, we must estimate the parameters of the noisy/leaky MAX from a database or elicit them from an expert. As in the case of the leaky OR, there are two ways of eliciting the parameters of each link in a MAX model. The parameters $c_y^{x_i}$ that we have used in the description of the causal noisy MAX are *net* (in the sense introduced in Section 4.2.1) and correspond to the question: "What is the probability that $Y = y$ when $X_i = x_i$ and *all the other causes of* Y are absent?"

In contrast, we could ask the expert: "What is the probability that $Y = y$ when $X_i = x_i$ and *the other causes of* Y *that we are considering in our model* are absent?" The answer would correspond to the $p_y^{x_i}$ parameters, defined as follows,

$$p_{z_i}^{x_i} = P(y|x_i, \neg x_{j\,(\forall j,\, j\neq i)}) \,, \tag{44}$$

which are the equivalent of the compound parameters that Henrion used in his description of the leaky OR. The relation between the $p_{z_i}^{x_i}$s and the $c_{z_i}^{x_i}$s is given in Appendix B. However, as mentioned above, in our experience, the first question seems more intuitive and was easier to answer for the experts that we have worked with. It also seems to yield parameters of higher accuracy [93].

4.3 Feeding-lines model

4.3.1 Definition of the model

The feeding-lines model proposed by Srinivas [78] follows the pattern of noisy models described in Section 3.2.1. Its causal interpretation corresponds to the case of several lines

feeding a certain device, for instance, several wires transmitting signals to an electronic component. It is a noisy model because it admits the possibility that some lines fail. Let X_i represent the input of line i and Z_i its output. If there is no failure, the output will be the same as the input, and for this reason each Z_i must take values in the same domain as its corresponding X_i. If there is a failure, the output will be z_i with probability q_{z_i}. Therefore, the probability of a failure is $q_{total} = \sum_{z_i} q_{z_i}$, and the conditional probability of Z_i is[12]

$$P(z_i|x_i) = \begin{cases} 1 - \sum_{z'_i \neq x_i} q_{z'_i} & \text{if } z_i = x_i \\ q_{z_i} & \text{if } z_i \neq x_i \end{cases} , \tag{45}$$

which simply reflects the fact that the value of Z_i will be the same as that of X_i unless a failure in the line leads Z_i to a different state.

The behavior of the device, whose inputs are the Z_is and whose output is Y, is modeled by a certain function, $y = f(\mathbf{z})$. Implicit in this model is the assumption of a deterministic device.[13] In principle, f can be any discrete function. As an example, Srinivas [78] proposed the function

$$y = f(x_1, \ldots, x_n) = y_{\min} + \left\lceil (y_{\max} - y_{\min}) \, \frac{1}{n} \sum_i \frac{x_i - x_{i\,\min}}{x_{i\,\max} - x_{i\,\min}} \right\rceil . \tag{46}$$

When $x_i = x_{i\,\min}$, the value of the ith term in the sum on i is 0. In contrast, when $x_i = x_{i\,\max}$, the i-th term contributes 1 to the sum. Therefore, $\frac{1}{n}\sum_i \frac{x_i - x_{i\,\min}}{x_{i\,\max} - x_{i\,\min}}$ is the average of the normalized values of the X_is, and this average is then mapped from $[0, 1]$ onto the set $\{y_{\min}, \ldots, y_{\max}\}$. When $y_{\max} = x_{i\,\max}$ and $y_{\min} = x_{i\,\min}$ for all i, this function simplifies into the discrete average function shown in Table 1.

It is easy to check that

$$y = f(\mathbf{x}) = y_{\min} \iff \forall i,\ x_i = x_{i\,\min} .$$

When Y and the X_is are all Boolean, then $Y = 0$ if and only if all the Z_is take the value 0, and $Y = 1$ otherwise, which means that the relation between Y and the Z_is is given by a deterministic OR, since the discrete average function simplifies into the OR function in case of Boolean variables (cf. Section 3.1). If $q_{+x_i} = 0$, Srinivas' model becomes a noisy OR in which $c_i = 1 - q_{\neg x_i}$.

The CPT for this model must be obtained directly from Equation 8 as, in general, there is no way of computing it more efficiently.

4.3.2 Comparison with the noisy OR/MAX

Both the noisy MAX and the feeding-lines model are noisy ICI models (cf. Section 3.2) that generalize the noisy OR to non-binary variables. Nevertheless, their semantics are quite different. In the noisy MAX model, Z_i represents the probability that a cause X_i has

12. Actually, Srinivas [78] used a slightly different notation. Each parameter q_{z_i} in our description corresponds to a certain $P_i^{inh}(I_i(x_i))$ in his paper (Z_i has the same domain as X_i).

13. Srinivas explained that this restriction can be relaxed by adding an AND model with two inputs, one being the output of the ideal device (without failure) and the other representing the possibility of a failure of the device; the output of this AND would be the output of the real device. An easier solution would be to use the simple AND described in Section 5.1.3.

changed the value of Y from its minimum to a certain value y. For this reason, the domain of Z_i is the same as that of Y. In contrast, each Z_i in the feeding-lines model represents the output of line i. If there is no failure, the output value, z_i, is the same as the output value, x_i, and for this reason Z_i has the same domain as X_i, which may be different from that of Y.

Another difference is that the feeding-lines model can use any function $f : X_1 \times X_2 \times \ldots \times X_n \mapsto Y$ (the function in Equation 46 is only one of the many possibilities), while in the noisy MAX the function is always the maximum.

Finally, in the feeding-lines model there are m_{X_i} parameters q_{x_i} for each link $X_i \rightarrow Y$, while the noisy MAX has $m_Y \cdot m_{X_i}$ parameters $c_y^{x_i}$ for each link, $(m_Y - 1) \cdot m_{X_i}$ of which are independent. This is, however, a minor difference, since the feeding-lines model could be modified by imposing different restrictions on the CPTs $P(z_i|x_i)$.

The feeding-lines model would be particularly suitable for diagnosis and reliability analysis of networks, electric circuits, and other systems consisting of different components connected by noisy lines. However, to our knowledge, it has never been used for building real-world applications.

5. AND/MIN models

In Section 3.1, we defined the deterministic AND and MIN, and mentioned that the deterministic MIN reduces to the deterministic AND when applied to Boolean variables. In this section, we analyze the noisy and leaky versions of the noisy AND, the simple AND, and the noisy MIN. We also discuss the application of those models to the construction of real-world applications.

5.1 Definition of the AND/MIN models

5.1.1 Noisy AND

The parents of a noisy AND can be interpreted as the conditions necessary for Y to be true. In the most general version of this model, each condition can be inhibited or substituted. If q_i is the probability that the ith inhibitor is active when condition X_i is fulfilled, then $c_i = 1 - q_i$ (see Table 6) If there is no inhibitor for X_i, then $c_i = 1$. Similarly, s_i is the probability that the ith substitute replaces X_i when this condition is not met. If there is no substitute for X_i, then $s_i = 0$. In general, $c_i \cong 1$ and $s_i \cong 0$.

$P(z_i\|x_i)$	$+x_i$	$\neg x_i$
$+z_i$	c_i	s_i
$\neg z_i$	$1 - c_i$	$1 - s_i$

Table 6: Parameters of the noisy AND for the link $X_i \rightarrow Y$.

In analogy with the derivation of the CPT for the noisy OR (Eq. 19), the CPT for the noisy AND can be obtained from Equation 8 by taking into account that $f_{\text{AND}}(\mathbf{z}) = +y$

only for the configuration $(+z_1, \ldots, +z_n)$. Therefore,

$$P(+y|\mathbf{x}) = \prod_i P(+z_i|x) = \prod_{i \in I_+(\mathbf{x})} c_i \prod_{j \in I_\neg(\mathbf{x})} s_j \ , \tag{47}$$

i.e., each X_i present ($X_i = +x_i$) contributes a factor c_i to this product, and each X_j absent ($X_j = \neg x_j$) contributes a factor s_j. If a certain condition X_k is absent and has no substitute ($s_k = 0$), then $P(+y|\mathbf{x}) = 0$, regardless of the values taken on by the other variables.

In particular, if there are no inhibitors, then $c_i = 1$ for all i and

$$P(+y|\mathbf{x}) = \prod_{j \in I\neg(\mathbf{x})} s_j \ ,$$

i.e., the probability of Y is computed by taking into account the possibility of finding substitutes for the absent conditions.

Similarly, if there are no substitutes, then $s_i = 0$ for all i and

$$P(+y|\mathbf{x}) = \begin{cases} \prod_i c_i & \text{if } \forall i,\ X_i = +x_i \\ 0 & \text{otherwise} \end{cases} \ ,$$

which means that Y only occurs when all the conditions are fulfilled and none have been inhibited. This particular case was presented as the definition of the noisy AND in [29, 57].

Overspecification of the noisy AND. Given a noisy AND, in general we can obtain an equivalent model by replacing c_1 with $k \cdot c_1$, s_1 with $k \cdot s_1$, c_2 with c_2/k and s_2 with s_2/k, provided that the new parameters are all between 0 and 1. This means that the noisy AND is overspecified in the sense that a CPT can be parametrized in several different ways, i.e., there is in general a whole family of noisy ANDs representing the same CPT. In Section 5.2.1 we will discuss the implications of this property for knowledge engineering.

5.1.2 Leaky AND

As mentioned above, the leaky AND is obtained from the standard noisy AND by adding an implicit inhibitor that — with probability q_L — may prevent the occurrence of Y even when all the conditions explicit in the model are fulfilled. Therefore, the CPT for the leaky AND is

$$P(+y|\mathbf{x}) = (1 - q_L) \cdot \left(\prod_{i \in I_+(\mathbf{x})} c_i \right) \cdot \left(\prod_{j \in I_\neg(\mathbf{x})} s_j \right) \ , \tag{48}$$

which is a generalization of Equation 47. (The proof of this equation is similar to that of Eq. 22.)

The *noisy AND* is a particular case of the leaky AND with $q_L = 0$. The *semi-deterministic AND* model [5], which admits the possibility that Y is false even when all its conditions are fulfilled, with probability q, is a particular case of the leaky AND, in which $c_i = 1$ and $s_i = 0$ for all i, and $q_L = q$.

Please note that, as a consequence of the overspecification of the noisy AND mentioned in the previous section, the leak parameter can be absorbed into the parameters of any link,

for instance by replacing c_1 and s_1 with $(1 - q_L) \cdot c_1$ and $(1 - q_L) \cdot s_1$. Therefore, a leaky AND is always equivalent to a noisy AND. However, from the point of view of knowledge engineering, in some cases it may be convenient to use a leaky AND instead of an equivalent noisy AND, as discussed in Section 5.2.1.

In the leaky AND, q_L represents the effect of a global inhibitor that can make Y to be false even when all the conditions have been fulfilled. However, the leaky AND does not admit a global substitute that allows Y to be true when some or all the conditions fail: according to Equation 48, it suffices that one condition fails ($j \in I_{\neg}(\mathbf{x})$) and is not substituted ($s_j = 0$) to make Y fail. This property is a consequence of the internal structure of the leaky AND, shown in Figure 6: if f is the AND function and one of the Z_is is false, then Y is false, independently of the value of Z_L and the other Z_js. For this reason, if we need to represent an AND-like model in which Y can be true even when some of its conditions fail, we need either a leaky model based on a different f or a non-ICI model, such as the one shown in the next section.

5.1.3 Simple AND

It may happen in practice that, for a certain family, the expert only knows that Y may be false even when all the conditions are fulfilled, and/or Y can be true when some conditions fail, but he/she is not able to assess the parameters c_i and s_i associated with the inhibitor and substitute of each single condition X_i. In this case, it may be useful to apply a different probabilistic extension of the deterministic AND, namely the simple AND (see Section 3.3, especially Figure 7), which only requires two parameters: c, representing the probability that Y occurs when all the conditions X_i are satisfied, and s, the probability that Y occurs when some of its conditions fail, as shown in Table 7.

$P(y\|z)$	$+z$	$\neg z$
$+y$	c	s
$\neg y$	$1 - c$	$1 - s$

Table 7: Parameters for the simple AND. When $c \cong 1$ and $s \cong 0$ this model approaches the deterministic AND.

The CPT for this model is

$$P(+y|\mathbf{x}) = \begin{cases} c & \text{for } \mathbf{x} = (+x_1, \ldots, +x_n) \\ s & \text{otherwise .} \end{cases}$$

Clearly, the deterministic AND is a particular case of the simple AND in which $c = 1$ and $s = 0$. The *semi-deterministic AND* model [5], introduced in the previous section as a particular case of the leaky AND, is also a particular case of the simple AND, in which $c = 1 - q$ and $s = 0$.

However, the leaky AND and the simple AND are very different: the former is an ICI model, whose internal structure is shown in Figure 3, while the latter is an SCM (see Sec. 3.3 and Figure 7). Secondly, the noisy AND with n parents requires $2n$ parameters and the leaky AND $2n + 1$, while the simple AND requires only two, regardless of the number of

parents. Finally, the main difference between both models is that the simple AND admits a global substitute, represented by parameter s, which allows Y to be true even if all the conditions $\{X_i\}$ are false, while the noisy AND excludes that possibility.[14]

5.1.4 MIN models

The noisy MIN only differs from the generalized noisy MAX described in Section 4.1.5 in that, obviously, the underlying function f is a min instead of a max. As in the noisy MAX model, the only restriction is that Y is an ordinal variable. Then $Z_i = y$ represents the fact that X_i has led Y to take on the value y and the parameter $c_y^{x_i} = P(Z_i = y|x_i)$ represents its probability. The resulting value of Y will be the minimum of the individual values produced by each X_i.

If we want to make sure that higher values of X_i lead to higher values of Y for some particular link $X_i \rightarrow Y$, the parameters of that link must satisfy the condition given by Equations 35 and 36, as in the case of the noisy MAX.

Finally, we might have a leaky MIN, analogous to the leaky MAX.

The CPT for the noisy MIN the leaky MIN can be obtained from Equations 28 and 31 by replacing "$\leq$" with "$\geq$", "-1" with "$+1$", and "$y_{\min}$" with "$y_{\max}$".

5.2 Knowledge engineering for the AND/MIN models

5.2.1 Criteria for applying the noisy AND

When describing the OR model, we spoke about "causes that may produce Y," while when describing the AND models, we spoke of "conditions for Y." This, however, should not imply that an interaction expressed as a set of conditions should be modeled by an AND model instead of an OR model. The AND model represents a conjunction, while the OR represents a disjunction of premises. Therefore, the OR must be used when each of the conditions is *sufficient* to guarantee Y, while the AND must be used only to represent the interaction among a set of *necessary* conditions. (As mentioned in Section 3.1, when r conditions must be met simultaneously for Y to occur, the appropriate model should be based on the threshold function, defined in Table 1.)

Apart from this, the noisy AND is more flexible than the noisy OR in several ways. First, the parents of a noisy OR must all be Boolean, but the noisy AND can accommodate any binary variables: in fact, the role of $+x_i$ and $\neg x_i$ in Table 6 is symmetric, while in Table 3 it is not. For this reason, a non-Boolean variable such as the sex of the patient, which could not be one of the parents of an OR model, can perfectly be one the conditions of an AND model. For instance, *Sex=female* is a condition for being pregnant or for suffering from breast cancer, in the same way as *Sex=male* is a condition for suffering from prostate

14. If we need to model a problem in which there are several conditions X_i for Y, each having different inhibitors and substitutes, and there is also a global substitute, both the leaky AND and the simple AND would be inadequate. A solution might be to combine both of them in a three-layered model: the first level would be given by the individual relations $X_i \rightarrow Z_i$, with parameters $c_i = P(+x_i| + z_i)$ and $s_i = P(+x_i|\neg z_i)$; the second level would be a deteministic AND relation between the Z_is and Z, and the third level would be a probabilistic relation between Z and Y, with only one parameter $s = P(+y|\neg z)$, since $P(+y| + z) = 1$. The number of parameters would then be $2n + 1$, inheriting the above-mentioned overspecification of the noisy AND. However, we have found no situtation requiring such a sophisticated model, neither in our experience nor in the literature.

diseases. The following example will illustrate some of the issues that play a role in building noisy AND models.

Example. Let us show by means of an example how to model the probability of *Pregnancy* (P) by using a noisy AND.[15] We may select three variables: *Sex* (S), *Intercourse* (I), and *Contraceptives* (C)—see Figure 9. If the values of the variable *Sex* (S) are $+s =$ *female* and $\neg s =$ *male*,[16] parameter c_1 represents the probability that a woman can become pregnant provided that the rest of the conditions are fulfilled. At the first sight, we might think that $c_1 = 1$, but we must take into account that not all women are fertile, because of their age or for any other reason. In an adult population, c_1 would be close to 1, but in a general population with little girls and elderly women, it would be lower. Parameter s_1 would represent the probability of a male getting pregnant; obviously, $s_1 = 0$.

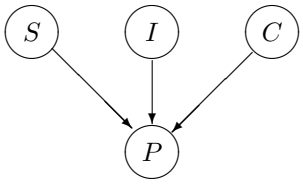


Figure 9: An example involving a noisy AND model. The nodes represent *Sex* (male or female), *Intercourse*, *Contraceptives*, and *Pregnancy*.

Since I represents *Intercourse*, parameter c_2 would indicate the probability of pregnancy when having sexual intercourse, provided that the other conditions, such as being a fertile female, etc., are fulfilled. We may take $c_2 \approx 1$; in that way $1 - c_2$ represents the effect of the inhibitors associated with I, such as the fact the male can be infertile. Parameter s_2 represents the probability of pregnancy when not having intercourse. If we exclude the possibility of in vitro fertilization, which would be a substitute for I, then $s_2 = 0$.

If the values of C are $+c =$ *contraceptives* and $\neg c =$ *no contraceptives*, c_3 and s_3 will be the probabilities of getting pregnant when using / not using contraceptives, respectively. Therefore, $c_3 \approx 0$ (for instance, $c_3 = 0.02$) and $s_3 = 1$.[17]

These three conditions are not sufficient for the onset of a pregnancy. Even when a fertile female has sexual intercourse without contraceptives and none of the anomalous inhibitors mentioned above appears, pregnancy does not always occur, for many reasons —

15. The simple AND discussed in Section 5.1.3 is inappropriate in this case, because s would represent the probability of getting pregnant when *any* of the conditions fails. When the 'not-contraceptives' condition fails, i.e., when a woman is taking contraceptives, the probability of pregnancy is small but positive, say $s = 0.01$. Therefore, the resulting model would predict that the probability of pregnancy for a woman not having intercourse would be 0.01, and worse even, the probability of a man to be pregnant would also be 0.01.
16. Even though S is not a Boolean variable, we use the notation $+s =$ *female* to represents the fulfillment of the first contition necessary for pregnancy.
17. This case is an exception to the rule, mentioned in the first paragraph of Section 5.1.1, that in general $c_i \cong 1$ and $s_i \cong 0$. The reason is that in this example the second parent is *Contraceptives*, while the condition for *Pregnancy* is "no contraceptives".

for example, because the intercourse took place not within the fertile period of the woman's monthly cycle, because the embryo does not implant in the uterus, etc. It is then necessary to add a leak parameter, q_L, representing the probability that some conditions (inhibitors), not modeled explicitly, prevent pregnancy.

Finally, we must check the hypothesis of independence of causal influence (cf. Sec. 3.2), i.e., whether the inhibitory and substitutionary mechanisms, including those represented by the leak factor, are a-priori independent and they do not interact with one another. In our example, it seems reasonable to assume that this assumption is correct. (One might object by noticing, for example, that the contraceptive cannot fail when the male is infertile, and so the inhibitors of C and the leaky parameters, but in fact, the above model just says that the contraceptive can fail independently of the male fertility. It just happens that a failure with an infertile male does not yield to a pregnancy, which is also predicted by this model.)
■

Please note that in the AND model each parent X_i represents one "external" condition that can be observed, while Z_i represents the "true" or "internal" condition. Thus, in our example, S (i.e., X_1) represents the condition of being a woman, while Z_1 represents the condition of being a fertile woman; I (i.e., X_2) represents the condition of having an intercourse, while Z_2 represents the condition of having intercourse with a fertile male; and C (X_3) represents the use of contraceptives, while Z_3 indicates whether the contraceptive is effective in a particular case. Given this definition of the Z_is, in the general population, $c_1 = P(\textit{fertile-woman}|\textit{woman}) \approx 0.5$.

Alternatively, we may redefine Z_1 so that it represents not only the fact that the woman is fertile, but also that she is in a fertile day of the monthly cycle. In this new model, $c_1 = P(\textit{woman-on-fertile-day}\ |\textit{woman}) \approx 0.1$. Therefore, in the first model, the probability of being in an infertile day was part of the leak parameter q_L, while in the new model this probability is contained in c_1. Please remember that in Section 5.1.2 we mentioned the possibility of absorbing the leak parameter, which represents a global inhibitor, into the parameters of some link. We also mentioned in Section 5.1.1 that the noisy AND is overspecified, i.e., that in general a CPT can be parametrized in several ways. Therefore, different definitions of the Zs lead to different parametrizations and all of them may be correct. However, it is very important that, in the process of parameter elicitation, the knowledge engineer and the human expert determine as precisely as possible the meaning of each Z_i, because an imprecise definition of the Z_is may result in neglecting or double-counting some inhibitory mechanisms.

5.2.2 Criteria for applying the noisy MIN

The application of the MIN model when Y is not binary is based on the assumption that this variable remains in its maximum unless it is lowered by some of the factors X_i. Therefore, the question asked of experts when eliciting parameters $c_y^{x_i}$ would be: "What is the probability that factor X_i lowers Y to its value y when no other factor has lowered it?"

In our experience in building expert systems, we have never encountered an example of a MIN gate with a multi-valued Y, but it does not mean that this model cannot be useful to other knowledge engineers in the future. However, there are several examples in which the noisy MIN can be used as a generalization of the noisy AND when Y is binary and

some of its parents are multi-valued. It would be the equivalent of adding more columns in Table 6, one for each value of X_i.

Example. Let us assume that we wish to refine the example in the previous section by taking into account the effectiveness of the different contraceptives. Then, the domain of variable C would have a value x_3 for each method that we want to consider, including *no-contraceptives*, and each parameter $c_{+y}^{x_3}$ would represent the probability of getting pregnant when using method x_3 (provided that the other conditions are fulfilled). ■

In this use of the noisy MIN model, the number of parameters for a link $X_i \rightarrow Y$ is the same as the number of values of X_i, because $c_{\neg y}^{x_i} = 1 - c_{+y}^{x_i}$. For each $c_{+y}^{x_i}$, the question that the knowledge engineer should ask the expert is: "What is the probability that Y is true when $X_i = x_i$ and all the other conditions hold?"

6. XOR models

6.1 Properties of XOR models

The *deterministic XOR* model was introduced in Section 3.1—see Tables 1 and 2. It would be possible to define a *noisy XOR* by following the scheme given in Section 3.2.1. In the case of a two-parent family, the noisy XOR would be given by Table 8. In principle, it is not possible to define a *leaky XOR* following the scheme given in Section 3.2.2, because the XOR function is not associative (cf. Sec. 3.1).[18]

$P(+y\|x_1, x_2)$	$+x_1$	$\neg x_1$
$+x_2$	$c_1 \cdot (1 - c_2) + (1 - c_1) \cdot c_2$	$s_1 \cdot (1 - c_2) + (1 - s_1) \cdot c_2$
$\neg x_2$	$c_1 \cdot (1 - s_2) + (1 - c_1) \cdot s_2$	$s_1 \cdot (1 - s_2) + (1 - s_1) \cdot s_2$

Table 8: CPT for a noisy XOR with two parents, where $c_i = P(+z_i| + x_i)$ and $s_i = P(+z_i|\neg x_i)$.

Examples of the application of the XOR are very rare in practice. Besides the XOR-based classifiers discussed in the next subsection, the only example of which we are aware is the following case, mentioned by Jurgelenaite and Lucas [44]: a bacterial infection may be treated with either bactericidal drugs, such as penicillin, or bacteriostatic drugs, such as chlortetracyclin, but the join administration of both has virtually no effect, because they are based on antagonic causal mechanisms. However, in this case the direct specification of the CPT would need 4 parameters, exactly the number required by an XOR, and the parameters of the CPT are much more intuitive than those of the XOR. For this reason,

18. It would be possible to define a quasi-leaky XOR by introducing a three-valued auxiliary variable Z such that z_0 means "no parent of Y takes the value *true*," z_1 = "exactly one parent of Y takes the value *true*," and z_2 = "at least two parents of Y take the value *true*." Then, $P(+y|z_1) = 1$ and $P(+y|z_0) = P(+y|z_2) = 0$. The function relating Z and the X_i would be associative and, consequently, might be governed by a leaky model.

    We can not see any utility of this leaky XOR for knowledge engineering, but still the idea of introducing a three-valued Z can serve to speed up the propagation of evidence by applying *parent divorcing* [63] or the so-called *temporal decomposition* [34, 35].

even in this example, it would counterproductive to try to build the CPT by means of an XOR.

If the case of a family with many parents, a noisy XOR would have the advantage of reducing the number of parameters from 2^n to $2n$, but it is difficult for us to imagine a real-world situation in which there are several possible causes of an effect Y such that each one separately can produce Y but the concurrence of two or more causes prevents the effect. If that hypothetical situation occur, perhaps it would be possible to find an interpretation for the auxiliary variables $\{Z_i\}$ and for the parameters of the noisy XOR. However, in the absence of a qualitative model of causal influences it is virtually impossible to give an interpretation to the Z_is and to the parameters of the model and, consequently, it is not possible to pose meaningful questions to a human expert or to establish a correspondence between the frequencies stored in a database and the parameters of the XOR.

In summary, we have discussed here the noisy XOR for the sake of completeness, not because we think it can be useful in practice.

Nevertheless, occasionally we see our students improperly applying the XOR, because of the following argument: "In this domain, the concurrence of causes is so improbable that in practice every effect has only one cause; for this reason we use the XOR, which leads to the diagnosis of only one cause for each effect. The OR does not look appropriate because it might diagnose two or more causes." This is a serious modeling error and the reasoning on which it is based is flawed: if the prior probability of each of the causes is very small, the OR model will not diagnose the presence of two of them unless there is enough evidence for it, a situation that, even though rare, may occur, and would not be properly tackled by an XOR model.

6.2 XOR-based classifiers

Although the noisy XOR is not useful in practice, its deterministic version can be applied in classification problems as follows. Let us consider an artificial vision system trying to determine whether the object in a certain image is a hammer, a screwdriver, or a pair of pliers (we assume that there is no other possibility). The probabilities of finding each one of them are respectively p_1, p_2 and p_3, and $p_1+p_2+p_3 = 1$. A naïve knowledge engineer might build a tree, with a root Y and three children, X_1, X_2 and X_3 (see Figure 10). Each X_i may have several children, representing the findings that characterize X_i. For example, F_1 might represent the presence of a wooden handle, which characterizes a hammer. The value $+y$ means that there is an object in the image, $+x_1$ means that the object is a hammer, $+x_2$ that it is a screwdriver, and $+x_3$ that it is a pair of pliers. The prior probability $P(+y)$ does not matter because we will know with certainty if there is an object in the image or not. The probabilities for link $Y \rightarrow X_i$ are $P(+x_i|+y) = p_i$ and $P(+x_i|\neg y) = 0$. When there is an object in the image, this model correctly gives the a priori probability that it is a hammer, a screwdriver of a pair of pliers. Nevertheless, it incorrectly asserts that the object can be at the same time a hammer and a screwdriver, with probability $P(+x_1 \wedge +x_2|+y) = p_1 \cdot p_2 > 0$. Furthermore, $P(x_2|+y, x_1) = P(x_2|+y)$. This means that a certain finding contributing evidence for or against the identification of the object as a hammer does not necessarily modify the probability that the object is a screwdriver.

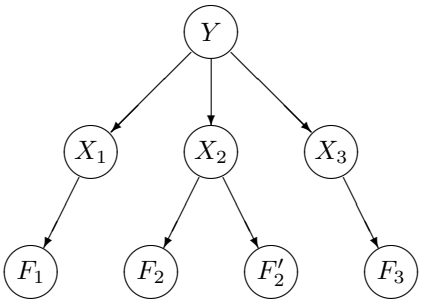


Figure 10: A wrong graph for the classification problem.

There are at least two ways to properly represent this problem. The first is the naïve Bayes classifier [23], whose graph is a tree having a root node with one value per class. The fact that the values of a variable are exclusive and exhaustive guarantees that each object is assigned to exactly one class. All the findings F_i are children of the root node (see Figure 11). The problem is that each link requires a CPT that grows linearly with the number of objects to be identified. An even more serious problem is the lack of modularity, in the sense that the addition of a new object to the list (for instance, a set of scissors), would oblige to rebuild all the conditional probabilities $P(F_i|x)$ in the model.

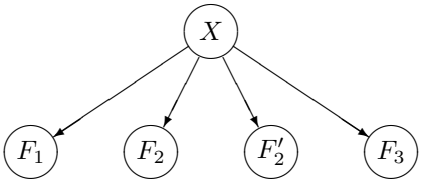


Figure 11: The naive-Bayes classifier.

An alternative model for classification based on the deterministic XOR can be built by including a Boolean variable X_i for each class, as shown in Figure 12. The prior probability of X_i is set to

$$P(+x_i) = \frac{p_i}{1+p_i} \,, \tag{49}$$

whereby the values of the p_is must add to one, i.e., $\sum_i p_i = 1$. Then we add a node Y, with n parents $\{X_1, \ldots, X_n\}$, interacting through a deterministic XOR and set the value

of Y to $+y$ — see Figure 12. This model correctly returns that $P(+x_i| + y) = p_i$ and $P(+x_i \wedge +x_j| + y) = 0$ for $i \neq j$.[19]

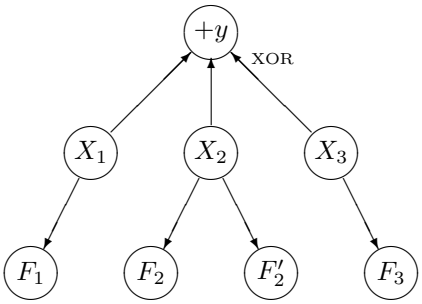


Figure 12: A classifier based on the XOR model.

When a node F_j represents a specific feature of one of the objects, X_i, the graph must contain a link $X_i \rightarrow F_j$. If F_j does not provide information for discriminating the other objects, i.e., if $P(+f_j| + x_k) = P(+f_j| + x_l)$ for every k and l different from i, then X_i will be the only parent of F_j. For example, if the probability of detecting a wooden handle in the image (by error) is the same for the screwdriver (X_2) and for the pliers (X_3), then it suffices to have one link pointing to F_1, namely $X_1 \rightarrow F_1$. If a node F_j represents a feature of both X_i and X_k, these two nodes should be the parents of F_j, interacting by means of a noisy OR.

The marginal and conditional probability distributions given by this model are correct only when $Y = +y$, not when the value of Y is $\neg y$ or when it is unspecified. This is not a problem in practice, since this model is intended for classifying an object when the object has been observed.

The naïve-Bayes model is more adequate when the classifier is learned from a database. When the classifier is build and maintained manually, however, the alternative XOR-classifier might be more adequate. Given that classifiers are usually built from data, the naïve-Bayes model seems more appropriate in general, but it might happen that in some

19. The proof of the first of these equations is as follows:

$$\begin{aligned}
P(+x_1| + y) &= P(+x_1, \neg x_2, \neg x_3| + y) = \\
&= \frac{P(+y| + x_1, \neg x_2, \neg x_3)P(+x_1)P(\neg x_2)P(\neg x_3)}{\sum_{x_1}\sum_{x_2}\sum_{x_3} P(y|x_1, x_2, x_3)P(x_1)P(x_2)P(x_3)} \\
&= \frac{P(+x_1)P(\neg x_2)P(\neg x_3)}{P(+x_1)P(\neg x_2)P(\neg x_3) + P(\neg x_1)P(+x_2)P(\neg x_3) + P(+x_1)P(\neg x_2)P(+x_3)} \\
&= \frac{P(+x_1)/P(\neg x_1)}{P(+x_1)/P(\neg x_1) + P(+x_2)/P(\neg x_2) + P(+x_3)/P(\neg x_3)} \ .
\end{aligned}$$

From Equation 49 we have that $P(+x_i)/P(\neg x_i) = p_i$ and, therefore,

$$P(+x_1| + y) = \frac{p_1}{p_1 + p_2 + p_3} = p_1 \ .$$

The second equation follows immediately from the fact that in the XOR $P(+y| + x_i \wedge +x_j) = 0$ for $i \neq j$.

problems of knowledge engineering or learning, the XOR-classifier were more appropriate [cite the work by E. Sucar].

7. Temporal canonical models

7.1 Introduction: event networks

There are two main kinds of temporal probabilistic models. *Periodic models* consist in discretizing time and creating an instance of each random or decision variable for each point in time. Dynamic Bayesian networks [9], which include Markov chains as a special case, and dynamic LIMIDs [55, 84], which include Markov decision processes and dynamic influence diagrams, are examples of periodic probabilistic models. In this context "dynamic" is virtually a synonym for "periodic."

The second class of temporal probabilistic models is formed by *event networks*, which take a very different approach: each node or variable represents an event and each value of that variable represents the time in which the event occurs. There are two kinds of event networks: temporal-nodes Bayesian networks (TNBNs) [2] and networks of probabilistic events in discrete time (NPEDTs) [30]. Both of them use discrete variables. In the former, each value of a variable represents a time interval, whose length may differ from those of other values (other intervals) of the same variable and those of other variables. On the contrary, in an NPEDT, each value represents a point in time. The interval between consecutive points is always the same, and all the variables have the same domain, $\{t_1, t_2, \ldots, t_N,$ never$\}$; $v[t]$ means that event V occurs at time t, and v[never] that it does not occur.

In an NPEDT, the CPT for a node X without parents can be given either explicitly or by means of a function; for example, when the hazard rate is constant, the probability of X may be given by the exponential decay function:

$$P(x\,[t]) = (1-k)^t \cdot k\ . \tag{50}$$

In contrast, the CPTs for the nodes with parents are usually assumed to be *time invariant*:

$$P(y\,[t_Y + \triangle t] \mid x_1\,[t_1 + \triangle t], \ldots, x_n\,[t_n + \triangle t]) = P(y\,[t_Y] \mid x_1\,[t_1], \ldots, x_n\,[t_n])\ . \tag{51}$$

Periodic models are best suited for modeling and controlling dynamic processes, while event networks are more appropriate for diagnosing which events have happened and when, and for predicting when some events will occur (see [30]). A more detailed comparison of TNBNs vs. NPEDTs can be found in [28].

7.2 Temporal canonical models for NPEDTs

Two families of models have been proposed for NPEDTs: OR and AND [30].

The *temporal deterministic OR* is based on the assumption that Y occurs as soon as it has been caused by one of its parents. Therefore, if X_i produces Y at time t_i and X_j at time t_j, then Y occurs at time $\min(t_i, t_j)$. We define that for all t, $\min(t,$ never$) = t$. Consequently, the function f used for building a temporal deterministic OR is min.[20]

20. If we ordered the time from the future to the past, the deterministic OR and its probabilistic counterparts would be based on a max function rather than on min, as explained in [30].

The *temporal noisy OR* follows the framework described in Section 3.2.1 (see especially Figure 3). The interaction between Y and the Z_is is given by a temporal deterministic OR, i.e., by a min function. The interpretation of the auxiliary variables Z_i is straightforward: $z_i[t]$ means that X_i has produced Y at time t. The parameters $c_i[t, t']$ represent the probability that X_i occurring at time t produces Y at time t'. An obvious restriction for these parameters is that $t > t'$ implies $c_i[t, t'] = 0$, because the effect can not occur before the cause. An additional restriction in the case of time invariance is that $c_i[t + \triangle t, t' + \triangle t] = c_i[t, t']$. These restrictions simplify the elicitation of the parameters, as it suffices to give the probabilities $c_i[0, t]$ for all the values of t; in practice, either $c_i[0, t]$ is positive only for a few values of t or it can be approximated by an exponential decay, as in Equation 50.

The *temporal leaky OR* admits the possibility that causes not explicit in the model produce Y with a probability $c_L[t]$, which in the case of time invariance is a constant (even though it may be different for other leaky models in the same network).

Similarly, it is possible to define the deterministic, noisy, and leaky version of *temporal AND models*, all of them based on the max function, which implies that Y can not occur until all its parents have occurred (or have been substituted). In these models, a parent taking on the value "never" is sufficient to prevent the occurrence of Y.

It would also be possible to define temporal models based on a threshold function, such that Y occurs as soon m of its parents cause it. When $m = 1$ we have the temporal noisy OR, and when $m = n$, the temporal noisy AND.

8. Bibliographical notes

8.1 Models

The term "canonical model" in the context of probabilistic graphical models was introduced by Pearl [67].

The NOT model was first used by Heckerman and Breese [36]. The ADD model, sometimes called "noisy addition" or "noisy SUM," and the r-out-of-n function (cf. Section 3.1) were introduced by Heckerman [34, 36]. The threshold function was used in [45, 85].

An analysis of the properties of Boolean functions of n variables can be found in [25, 45, 89].

The *independence of causal influence* (ICI), was studied by Heckerman and Breese [34, 36] under the name "causal independence." The term ICI was proposed by Laskey.[21] Some authors occasionally call it "independence of causal interaction". Our framework for ICI models (Section 3.2) is an extension of the model proposed by Srinivas [78]. The analysis of leaky models (Section 3.2.2) is a contribution of this paper, although the idea of computing

21. In a message posted to the UAI mailing list on October 9, 1996, Kathryn Laskey wrote: "I am lobbying for a change in terminology from 'causal independence' to 'independence of causal influence' (ICI). [...] In my experience, beginning modelers often have trouble understanding the difference between:
—marginal independence of A and B (independence of two effects of a cause); and
—A and B independently acting as causal influences of C (as in a noisy-OR or noisy adder model).
Using the term 'causal independence' for the latter seems to exacerbate this confusion, while using the term 'independence of causal influence' seems to help clear it up."
David Heckerman replied: "I like the new name, as it is a more accurate description of the independence assumption."

the leak probability from a larger Bayesian network was used in [70]. Simple canonical models are a contribution of this paper.

The noisy OR was proposed by Good [32]. It was also studied by Kim and Pearl [49, 66], who coined the term "noisy OR," and Peng and Reggia [68]. Cozman [7, 8] has shown that the noisy OR is the only (binary) model that satisfies a set of desirable properties. This justifies why the noisy OR and its extension, the causal MAX, which satisfies basically the same properties, have been used in practice much more than any other model.

Henrion [39] introduced the leaky OR, Lemmer [56] the recursive noisy OR (RNOR), and Kuter et al. [50] the inhibitory RNOR.

Henrion [39] also introduced the first noisy MAX model, which we have called here *graded noisy MAX*. Díez [10] proposed the terms "noisy MAX" and "noisy MIN," formalized these models, and introduced the leaky MAX. The causal noisy MAX defined in Section 4.1.6 is slightly more general than the graded noisy MAX in [10, 39], because it does not require that the parents are graded variables. The noisy MAX described in Section 4.1.5 is even more general because it neither requires that Y is a graded variable nor imposes any condition on the parameters $c_y^{x_i}$. The distinction between Díez's *net parameters*, c_is, and Henrion's *compound parameters*, p_is (cf. Sec. 4.2.1) is a contribution of this paper.

The feeding-lines model (Section 4.3) was proposed by Srinivas [78], although the name "feeding-lines" has been introduced in this paper.

The noisy AND presented in Section 5.1.1, which admits both inhibitors and substitutes, is a generalization of the noisy AND described by Galán and Díez [29] and studied by Lucas [57], which only admits inhibitors. The simple AND is a contribution of this paper. The semi-deterministic AND, which is a special case of both the leaky AND and the simple AND, was first proposed and applied by Conati et al. [5].

Lucas, Jurgelenaite and van Gerven [58, 44, 86] have studied the quantitative and qualitative properties of several noisy models based on different Boolean functions.

Temporal canonical models were proposed by Galán and Díez [29, 30].

A significant amount of research has been done on how the specific properties of the OR/MAX models can be used to generate qualitative explanations in expert systems [1, 11, 15, 18, 51, 66, 91]. Nevertheless, we have the feeling that there are still many possibilities of explanation in canonical models to be discovered. See also [52] for a review of explanation methods in Bayesian networks.

Several authors have analyzed the computational properties of canonical models, especially the OR/MAX models, in order to achieve more compact representations of CPTs and more efficient algorithms (both in time and space) for the computation of probabilities. References up to 2003 can be found in [12]; see also [83].

In this paper, focused on knowledge engineering, we have not reviewed statistical distributions that have been used for building probabilistic graphical models, especially in the context of learning models with continuous variables. Some of those distributions are: linear regression, generalized linear regression, logistic regression, Gaussian, conditional Gaussian, mixtures of Gaussians, mixtures of truncated exponentials, Gamma, etc. The reader interested in this topic can found numerous references by searching the Internet with each of these terms and "Bayesian networks."

8.2 Applications

Medicine is the field where canonical models have been more widely used. Habbema [33], Heckerman [38] and Shwe et al. [74] used the noisy OR for building models containing various diseases. The noisy MAX model, including the noisy OR as a particular case, was applied to medical domains, such as Díez et al.'s DIAVAL [11, 13], Pradhan et al.'s CPCS project [70], Oniśko et al.'s HEPAR II [64, 65], Kappen and Neijt's Promedas [48], Lacave and Díez's Prostanet [53], Gómez et al.'s IctNeo [31], etc. The DXpress tool, by Knowledge Industries, which includes the noisy OR as a modeling tool, was used for building three medical Bayesian networks for dementia, headaches, and sleep disorders, respectively.[22] Lucas [59] used the noisy AND in a model for managing infectious diseases at the ICU.

The noisy OR/MAX was also applied to troubleshooting of mechanical devices (printers, copiers, automobiles, and turbines) by Heckerman et al. [37]. Conati et al. [5] and Vomlel [87] have used canonical models for modelling students in intelligent tutoring systems.

Temporal canonical models were applied by Galán et al. to medicine [27] and to power plant diagnosis [28].

Zagorecki and Druzdzel [94] have studied the problem of how common noisy MAX models are in practice. They proposed an algorithm that fits a noisy MAX distribution to an existing CPT and then applied it to nodes of three sizeable existing Bayesian network models. They found that conditional probability distributions in as many as 50% of the nodes with two or more parents can be reasonably approximated by noisy MAX models. In another study [93], they compared the speed and the accuracy of elicitation of CPTs to elicitation of noisy OR distributions following both net and compound parameterizations (cf. Section 4.2.1). They found that the net parameterizations led to slightly more precise estimates than the other two parameterizations within a shorter amount of time.

The RNOR and the inhibitory RNOR were implemented by Kuter et al. [50] in CAT, a propietary tool used for interactively planning military operations.

Rish et al. [72] used the deterministic XOR when building Bayesian networks for decoding messages transmitted through noisy channels.

XOR classifiers (Sec. 6) were introduced in a previous version of this paper. Sucar [give references] used them for ???.

The above-mention DXpress (by Knowledge Industries), GeNIe [16], and MSBN [46] were the first general-purpose tools that supported canonical models for the construction of Bayesian networks and influence diagrams. To our knowledge, currently the most complete treatment of canonical models if offered by Elvira [24].

Neal [62], Friedman and Goldszmidt [26], Meek and Heckerman [60], and Oniśko et al. [65] used the noisy OR to improve the learning of probabilities. The group led by Peter Lucas has used the noisy threshold model for learning probabilistic models aimed at diagnosing and treating ventilator-associated pneumonia [45] and predicting carcinoid heart disease [85]. This group has also used other noisy ICI models for learning CPTs [42, 43].

8.3 Acquiring the numerical probabilities

In this section we will mention a few references that, even though not specific for canonical models, may help knowledge engineers to obtain the required probabilistic parameters.

22. See `http://www.kic.com/dxpress.htm`.

- Druzdzel and van der Gaag [20] discussed how to elicit the probabilities by combining qualitative and quantitative information.

- They were also the editors of a special issue of the *IEEE Transactions on Knowledge and Data Engineering* [21], based on an IJCAI'95 workshop entitled "Building Probabilistic Networks: Where Do the Numbers Come From?" [22].

- The book by Morgan and Henrion [61, chapters 6 and 7] discusses some techniques for "calibrating the experts" and some elicitation protocols.

- Van der Gaag, Renooij et al. [71, 81, 82] proposed several techniques for efficiently eliciting many probabilities.

- There is also a vast literature on transforming verbal expressions of probability, such as "very often," "almost always," etc., into numerical statements—see for instance [14, 92] and references therein.

- Druzdzel and Díez [17] showed that combining probability estimates from different sources, such as epidemiological studies, databases, and subjective estimates, may lead to significant errors, and gave criteria for avoiding them.

- Additionally, knowledge engineers should be aware of the psychological biases that affect the elicitation of probabilities; a lot of work has been done in this field after the pioneering works of Tversky and Kahneman [47, 79]. See, for instance, [3, 69].

- Finally, a sensitivity analysis might help to determine if some probabilities should be assessed more carefully as they have a higher influence on the results returned by the model. It is possible to find many sensitivity analysis methods by searching the Internet with the terms "sensitivity analysis" and either "Bayesian networks," "Markov networks," or "influence diagrams."

9. Conclusions

We argued that the hardest part in the process of building Bayesian networks and influence diagrams for real-world problems is obtaining their numerical parameters. The straightforward approach that involves elicitation of an exponential number of numerical probabilities composing a conditional probability table (CPT), is impractical for nodes having more than three or four parents. The so called *canonical models*, whose role can be compared to that of parametric conditional probability distributions, can significantly reduce the number of probability estimates required to quantify a conditional probability distribution. After studying some functions that can be used for building deterministic models, we proposed a general framework for models, with a special emphasis on ICI models, i.e., those based on the assumption of *independence of causal influence*. We then analyzed the most common families of canonical models, the noisy OR/MAX, the noisy AND/MIN, and the noisy XOR, generalizing them and offering criteria for applying these models in practice. We also briefly reviewed extending canonical models to temporal domains. Most of the non-deterministic models studied in this paper are based on the ICI assumption, with the only exception

of the RNOR and inhibitory RNOR models (Secs. 4.1.3 and 4.1.4) and simple AND (cf. Sec. 5.1.3).

The original contributions of this paper are:

- A general framework for canonical models, based on three categories: deterministic models, ICI models (which can be subdivided into noisy and leaky models), and simple canonical models (SCMs). The analysis of leaky models and the definition of SCMs have not been published previously. This framework does not only provide a unifying view of the models published so far, but also a guide for building user-tailored models.
- A review of the models proposed in the literature, with comprehensive bibliographical references.
- A generalization of some of the existing models, such as the noisy MAX and the noisy AND.
- A detailed analysis of OR/MAX, AND/MIN, and XOR models; in particular, XOR classifiers are a contribution of this paper.
- A guide for applying those models in practice, i.e., criteria for deciding which model can be applied in a certain situation and detailed recommendations for obtaining the numerical parameters. We have also discussed some errors that novice knowledge engineers are prone to make and how to prevent them.

Acknowledgments

This paper has benefited from comments by Rasa Jurgelenaite, Marcel van Gerven, and Adam Zagorecki.

F. J. Díez was supported by the Spanish Ministry of Education and Science under projects TIC–2001–2973–C05 and TIN–2006–11152. M. Druzdzel was supported by the Air Force Office of Scientific Research grants F49620–03–1–0187 and FA9550–06–1–0243, and by Intel Research. Our collaboration was enhanced by travel support from NATO Collaborative Linkage Grant number PST.CLG.976167. (However, the first author would prefer that public funds devoted to scientific research were not administered by military organizations.)

Appendices

Appendix A: Proofs of the theorems for leaky models

Proof of Theorem 1. The join probability for the large network is:

$$P(\mathbf{v}') = \prod_{V_i \in \mathbf{V} \setminus \{Y\}} P(v_i|\mathrm{pa}(v_i)) \cdot P(y|\mathbf{x}, \mathbf{x}_I) \cdot \prod_{V_j \in \mathbf{V}_I} P(v_j|\mathrm{pa}(v_j)) \; .$$

Given that $\mathbf{V} \cap \mathbf{V}_I = \varnothing$ and the only node in $\mathbf{V}$ having parents in $\mathbf{V}_I$ is Y, we have,

$$P(\mathbf{v}) = \sum_{\mathbf{v}_I} P(\mathbf{v}') = \prod_{V_i \in \mathbf{V} \setminus \{Y\}} P(v_i|\mathrm{pa}(v_i)) \cdot \sum_{\mathbf{v}_I} \left(P(y|\mathbf{x}, \mathbf{x}_I) \cdot \prod_{V_j \in \mathbf{V}_I} P(v_j|\mathrm{pa}(v_j)) \right) \; .$$

On the other hand, as no node in $\mathbf{V}_I$ has parents in $\mathbf{V}$, we have that

$$P(\mathbf{v}_I) = \sum_{\mathbf{v}} P(\mathbf{v}') = \left(\sum_{\mathbf{v}} \prod_{V_i \in \mathbf{V}} P(v_i|\mathrm{pa}(v_i)) \right) \prod_{V_j \in \mathbf{V}_I} P(v_j|\mathrm{pa}(v_j)) \; .$$

In the computation of $P(\mathbf{v}_I)$ all the nodes in $\mathbf{V}$ are barren nodes, which implies that

$$P(\mathbf{v}_I) = \prod_{V_j \in \mathbf{V}_I} P(v_j|\mathrm{pa}(v_j)) \; .$$

Therefore,

$$P(\mathbf{v}) = \prod_{V_i \in \mathbf{V} \backslash \{Y\}} P(v_i|\mathrm{pa}(v_i)) \cdot \sum_{\mathbf{v}_I} P(y|\mathbf{x}, \mathbf{x}_I) \cdot P(\mathbf{v}_I) \; . \tag{52}$$

We define $\mathbf{R}_I = \mathbf{V}_I \backslash \mathbf{X}_I$; i.e., $\mathbf{R}_I$ contains the implicit nodes that are not parents of Y. This leads to

$$\begin{aligned} \sum_{\mathbf{v}_I} P(y|\mathbf{x}, \mathbf{x}_I) \cdot P(\mathbf{v}_I) &= \sum_{\mathbf{x}_I} P(y|\mathbf{x}, \mathbf{x}_I) \cdot \sum_{\mathbf{r}_I} P(\mathbf{v}_I) \\ &= \sum_{\mathbf{x}_I} P(y|\mathbf{x}, \mathbf{x}_I) \cdot P(\mathbf{x}_I) \; . \end{aligned}$$

The subsets $\mathbf{X}$ and $\mathbf{X}_I$ have no common ancestor. For this reason, $P(\mathbf{x}_I) = P(\mathbf{x}_I|\mathbf{x})$ and

$$\sum_{\mathbf{v}_I} P(y|\mathbf{x}, \mathbf{x}_I) \cdot P(\mathbf{v}_I) = \sum_{\mathbf{x}_I} P(y|\mathbf{x}, \mathbf{x}_I) \cdot P(\mathbf{x}_I|\mathbf{x}) = \sum_{\mathbf{x}_I} P(y, \mathbf{x}_I|\mathbf{x}) = P(y|\mathbf{x}) \; .$$

The substitution of this result into Equation 52 proves the theorem. ∎

Proof of Theorem 3. The CPT for Y in the large network (noisy model) is, according with Equation 8,

$$\begin{aligned} P(y|\mathbf{x}, \mathbf{x}_I) &= \sum_{\mathbf{z}} \sum_{\mathbf{z}_I | f(\mathbf{z}, \mathbf{z}_I) = y} \prod_{i | X_i \in \mathbf{X}} P(z_i|x_i) \prod_{i | X_i \in \mathbf{X}_I} P(z_i|x_i) \\ &= \sum_{\mathbf{z}} \prod_{i | X_i \in \mathbf{X}} P(z_i|x_i) \sum_{\mathbf{z}_I | f(\mathbf{z}, \mathbf{z}_I) = y} \prod_{i | X_i \in \mathbf{X}_I} P(z_i|x_i) \; . \end{aligned}$$

Since f is associative,

$$f(\mathbf{z}, \mathbf{z}_I) = y \;\; \Leftrightarrow \;\; f(\mathbf{z}, f(\mathbf{z}_I)) = y \;\; \Leftrightarrow \;\; \exists \, z_L \mid f(\mathbf{z}_I) = z_L \wedge f(\mathbf{z}, z_L) = y \; .$$

Please note that $z_L \in \mathrm{range}(f(\mathbf{z}_I))$, in accordance with Equation 10. Consequently,

$$\sum_{\mathbf{z}_I | f(\mathbf{z}, \mathbf{z}_I) = y} \equiv \sum_{z_L | f(\mathbf{z}, z_L) = y} \sum_{\mathbf{z}_I | f(\mathbf{z}_I) = z_L}$$

and

$$P(y|\mathbf{x}, \mathbf{x}_I) = \sum_{\mathbf{z}} \prod_{i | X_i \in \mathbf{X}} P(z_i|x_i) \sum_{z_L | f(\mathbf{z}, z_L) = y} \underbrace{\sum_{\mathbf{z}_I | f(\mathbf{z}_I) = z_L} \prod_{i | X_i \in \mathbf{X}_I} P(z_i|x_i)} \; .$$

Because of Definition 2 (Eq. 11),

$$P(y|\mathbf{x},\mathbf{x}_I) = \sum_{\mathbf{z}} \prod_{i|X_i\in\mathbf{X}} P(z_i|x_i) \sum_{z_L|f(\mathbf{z},z_L)=y} P(z_L|\mathbf{x}_I) \ .$$

The CPT for Y in the small network (leaky model) is

$$\begin{aligned} P(y|\mathbf{x}) &= \sum_{\mathbf{x}_I} P(y|\mathbf{x},\mathbf{x}_I)P(\mathbf{x}_I|\mathbf{x}) = \sum_{\mathbf{x}_I}\sum_{\mathbf{z}} \prod_{i|X_i\in\mathbf{X}} P(z_i|x_i) \sum_{z_L|f(\mathbf{z},z_L)=y} P(z_L|\mathbf{x}_I)P(\mathbf{x}_I) \\ &= \sum_{\mathbf{z}} \prod_{i|X_i\in\mathbf{X}} P(z_i|x_i) \sum_{z_L|f(\mathbf{z},z_L)=y} \sum_{\mathbf{x}_I} P(z_L|\mathbf{x}_I)P(\mathbf{x}_I) \\ &= \sum_{\mathbf{z}} \prod_{i|X_i\in\mathbf{X}} P(z_i|x_i) \sum_{z_L|f(\mathbf{z},z_L)=y} P(z_L) \ . \end{aligned}$$

■

Appendix B: Conversion between the $p_y^{x_i}$s and the $c_y^{x_i}$s

Let us analyze now the relation between the $p_y^{x_i}$ parameters introduced in Equation 44 and the $c_y^{x_i}$s used in the definition of the noisy/leaky MAX (Eq. 27).

In analogy with Equation 29, we can define accumulative parameters for the $p_y^{x_i}$s, as follows:

$$P_y^{x_i} = P(Y \le y|x_i, \neg x_{j\,(\forall j,\, j\neq i)}) = \sum_{z_i\le y} p_{z_i}^{x_i} \ . \tag{53}$$

Because of Equation 40,

$$P_y^{x_i} = C_y^L \cdot C_y^{x_i} \cdot \prod_{j\neq i} C_y^{\neg xj} \ ,$$

and because of Equations 29 and 32,

$$C_y^{\neg xj} = \sum_{y'\le y} c_{y'}^{\neg xj} = \underbrace{c_{\neg y}^{\neg xj}}_{1} + \sum_{y'|\neg y<y'\le y} \underbrace{c_{y'}^{\neg xj}}_{0} = 1 \ .$$

Therefore,

$$P_y^{x_i} = C_y^L \cdot C_y^{x_i} \ . \tag{54}$$

We also have, from the definition of the Cs and the Ps (Eqs. 29 and 53) that

$$c_y^{x_i} = \left(\sum_{z_i\le y} c_{z_i}^{x_i}\right) - \left(\sum_{z_i\le y-1} c_{z_i}^{x_i}\right) = C_y^{x_i} - C_{y-1}^{x_i} \tag{55}$$

$$p_y^{x_i} = \left(\sum_{z_i\le y} p_{z_i}^{x_i}\right) - \left(\sum_{z_i\le y-1} p_{z_i}^{x_i}\right) = P_y^{x_i} - P_{y-1}^{x_i} \ . \tag{56}$$

In summary, the conversion from the $p_y^{x_i}$s to the $c_y^{x_i}$s, and vice versa, can be done by applying the equations indicated in parentheses in the following diagram:

$$p_y^{x_i} \underset{\text{Eq. 56}}{\overset{\text{Eq. 53}}{\rightleftarrows}} P_y^{x_i} \underset{\text{Eq. 54}}{\overset{\text{Eq. 54}}{\rightleftarrows}} C_y^{x_i} \underset{\text{Eq. 29}}{\overset{\text{Eq. 55}}{\rightleftarrows}} c_y^{x_i} \ .$$

This process is a generalization of Equation 42.

Appendix C: AND/OR duality

According to the De Morgan's laws, $x_1 \wedge \ldots \wedge x_n$ is equivalent to $\neg(\neg x_1 \vee \ldots \vee \neg x_n)$. Therefore, we can try to define the AND models by using the schema in Figure 13: if the relation among Y' and the X_i's is given by a deterministic OR, the relation between Y and Y' is given by a NOT (negation) and each relation between X_i' and X_i is also given by a NOT, then the relation between Y and the X_is corresponds to a deterministic AND. In a similar way, we can obtain the deterministic OR from the deterministic AND. There is a perfect duality between both models.

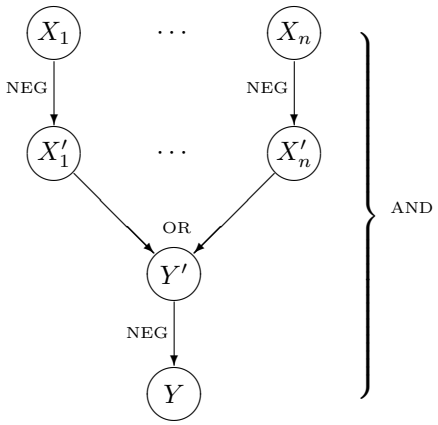


Figure 13: AND/OR duality: The deterministic AND can be obtained from the deterministic OR.

However, when applying the above schema to the noisy OR, we do not obtain the noisy AND as presented in this paper. The main reason is that the noisy OR has only one parameter per link (cf. Section 4.1.1), while the noisy AND has two (cf. Sec. 5.1.1). Coming down to the details, if the noisy OR in Figure 13 has a parameter c_i' for each link $X_i' \rightarrow Y'$, then Equation 19 tells us that

$$P(\neg y'|\mathbf{x}') = \prod_{i \in I_+(\mathbf{x}')} (1 - c_i')$$

and, after applying the negations, we obtain

$$P(+y|\mathbf{x}) = \prod_{i \in I_\neg(\mathbf{x})} (1 - c_i') \ .$$

The comparison of this result with Equation 47 shows that Y and $\mathbf{X}$ interact through a noisy AND such that $s_i = 1 - c'_i$ and $c_i = 0$, i.e., a noisy AND with substitutes (i.e., a failing condition may be replaced by a substitute condition) but without inhibitors.

Conversely, if we apply the schema in Figure 13) to the noisy AND, we do not obtain a noisy OR but a more general model having two parameters per link. Let c'_i and s'_i be the parameters for link $X'_i \to Y'$ in the noisy AND; the CPT for the resulting OR-like model is

$$P(\neg y|\mathbf{x}) = \prod_{i \in I_+(\mathbf{x})} s'_i \prod_{i \in I_\neg(\mathbf{x})} c'_i \,,$$

or, defining $c_i = 1 - s'_i$ and $s_i = 1 - c'_i$,

$$P(\neg y|\mathbf{x}) = \prod_{i \in I_+(\mathbf{x})} (1 - c_i) \prod_{i \in I_\neg(\mathbf{x})} (1 - s_i) \,. \tag{57}$$

According to this equation, c_i represents the probability that cause X_i produces the effect Y when X_i is present and the other causes of Y are absent (as in the standard noisy OR—see Eq. 19), while s_i represents the probability that a substitutive cause associated with X_i produces Y when X_i is absent. Apparently this is a leak probability, but in the leaky OR c_L is associated with the implicit causes, while in this generalized version of the noisy OR (the dual of the noisy AND) c_i is associated with one of the explicit causes, X_i and, therefore, the substitutive cause represented by c_i can only replace X_i, not the other causes of Y.

However, the idea of an implicit cause that can only substitute one of the causes of Y and not the others contradicts our intuitive understanding of causality, and for this reason in the definition of the noisy OR we do not allow this kind of substitutive causes, i.e., we require that $s_i = 0$. Consequently, there is not a perfect duality between the noisy OR and the noisy AND.

Finally, if we apply the scheme in Figure 13 with a MAX model instead of the OR and the INV model (defined in Section 3.1) instead of the NOT, we obtain the MIN model. In the same way, the MAX can be obtained from the MIN by the same procedure. In this case, there is a perfect duality between the two models.

References

[1] M. Agosta. Conditional inter-causally independent node distributions, a property of Noisy-OR models. In *Proceedings of the 7th Conference on Uncertainty in Artificial Intelligence (UAI'91)*, pages 9–16, Los Angeles, CA, 1991. Morgan Kaufmann, San Mateo, CA.

[2] G. Arroyo-Figueroa and L. E. Sucar. A temporal Bayesian network for diagnosis and prediction. In *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence (UAI'99)*, pages 13–20, Stockholm, Sweden, 1999. Morgan Kaufmann, San Francisco, CA.

[3] J. Baron. *Thinking and Deciding.* Cambridge University Press, Cambridge, UK, third edition, 2000.

[4] W. J. Clancey. The epistemology of rule-based expert systems — A framework for explanation. *Artificial Intelligence*, 20:215–251, 1983.

[5] C. Conati, A. S. Gertner, K. VanLehn, and M. J. Druzdzel. On-line student modeling for coached problem solving using Bayesian networks. In *Proceedings of the Sixth International Conference on User Modeling (UM'97)*, pages 231–242. Springer, Vienna, Austria, Chia Laguna, Italy, 1997.

[6] G. F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42:393–405, 1990.

[7] F. G. Cozman. Axiomatizing Noisy-OR. Technical Report BT/PMR/0409, Escola Politécnica, Universidade de São Paulo, Brazil, 2004.

[8] F. G. Cozman. Axiomatizing Noisy-OR. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-04)*, pages 979–980, Valencia, Spain, 2006. IOS Press, Amsterdam, The Netherlands. (A very short version of [7]).

[9] T. Dean and K. Kanazawa. A model for reasoning about persistence and causation. *Computational Intelligence*, 5:142–150, 1989.

[10] F. J. Díez. Parameter adjustment in Bayes networks. The generalized noisy OR–gate. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, pages 99–105, Washington, D.C., 1993. Morgan Kaufmann, San Mateo, CA.

[11] F. J. Díez. *Sistema Experto Bayesiano para Ecocardiografía*. PhD thesis, Dpto. Informática y Automática, UNED, Madrid, 1994. In Spanish.

[12] F. J. Díez and S. F. Galán. Efficient computation for the noisy MAX. *International Journal of Approximate Reasoning*, 18:165–177, 2003.

[13] F. J. Díez, J. Mira, E. Iturralde, and S. Zubillaga. DIAVAL, a Bayesian expert system for echocardiography. *Artificial Intelligence in Medicine*, 10:59–73, 1997.

[14] M. J. Druzdzel. Verbal uncertainty expressions: Literature review. Technical Report CMU-EPP-1990-03-02, Department of Engineering and Public Policy, Carnegie Mellon University, Pittsburgh, PA, 1989.

[15] M. J. Druzdzel. *Probabilistic Reasoning in Decision Support Systems: From Computation to Common Sense*. PhD thesis, Dept. Engineering and Public Policy, Carnegie Mellon University, Pittsburgh, PA, 1993.

[16] M. J. Druzdzel. SMILE: Structural Modeling, Inference, and Learning Engine, and GeNIe: A development environment for graphical decision-theoretic models. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI–99)*, pages 902–903, Orlando, FL, 1999.

[17] M. J. Druzdzel and F. J. Díez. Combining knowledge from different sources in probabilistic models. *Journal of Machine Learning Research*, 4:295–316, 2003.

[18] M. J. Druzdzel and M. Henrion. Intercausal reasoning with uninstantiated ancestor nodes. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, pages 317–325, Washington, D.C., 1993. Morgan Kaufmann, San Mateo, CA.

[19] M. J. Druzdzel and H. A. Simon. Causality in Bayesian belief networks. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, pages 3–11, Washington, D.C., 1993. Morgan Kaufmann, San Mateo, CA.

[20] M. J. Druzdzel and L. C. van der Gaag. Elicitation of probabilities for belief networks: Combining qualitative and quantitative information. In *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI'95)*, pages 141–148, Montreal, Canada, 1995. Morgan Kaufmann, San Francisco, CA.

[21] M. J. Druzdzel and L. C. van der Gaag. Building probabilistic networks: "Where do the numbers come from?" Guest editors' introduction. *IEEE Transactions on Knowledge and Data Engineering*, 12:481–486, 2000.

[22] M. J. Druzdzel, L. C. van der Gaag, M. Henrion, and F. V. Jensen, editors. *Working Notes of the IJCAI'95 Workshop "Building Probabilistic Networks: Where Do the Numbers Come From?"*, Montreal, Canada, 1995.

[23] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, New York, 1973.

[24] The Elvira Consortium. Elvira: An environment for creating and using probabilistic graphical models. In *Proceedings of the First European Workshop on Probabilistic Graphical Models (PGM'02)*, pages 1–11, Cuenca, Spain, 2002.

[25] H. B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, San Diego, CA, second edition, 2001.

[26] N. Friedman and M. Goldszmidt. Learning Bayesian networks with local structure. In *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence (UAI'96)*, pages 252–262, Portland, OR, 1996. Morgan Kaufmann, San Francisco, CA.

[27] S. F. Galán, F. Aguado, F. J. Díez, and J. Mira. NasoNet. Modelling the spread of nasopharyngeal cancer with temporal Bayesian networks. *Artificial Intelligence in Medicine*, 25:247–254, 2002.

[28] S. F. Galán, G. Arroyo-Figueroa, F. J. Díez, and L. E. Sucar. Comparison of two types of event Bayesian networks: A case study. *Applied Artificial Intelligence*, 21:185–209, 2007.

[29] S. F. Galán and F. J. Díez. Modelling dynamic causal interactions with Bayesian networks: Temporal noisy gates. In *Working Notes of the 2nd International Workshop on Bayesian and Causal Networks (CaNew'2000)*, pages 1–5, Berlin, Germany, 2000.

[30] S. F. Galán and F. J. Díez. Networks of probabilistic events in discrete time. *International Journal of Approximate Reasoning*, 30:181–202, 2002.

[31] M. Gómez, C. Bielza, J. A. Fernández del Pozo, and S. Ríos-Insua. A graphical decision-theoretic model for neonatal jaundice. *Medical Decision Making*, 2007. To appear.

[32] I. Good. A causal calculus (I). *British Journal of Philosophy of Science*, 11:305–318, 1961.

[33] J. Habbema. Models for diagnosis and detection of combinations of diseases. In F. de Dombal and F. Gremy, editors, *Decision Making and Medical Care*, pages 399–411. North-Holland, New York, 1976.

[34] D. Heckerman. Causal independence for knowledge acquisition and inference. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, pages 122–127, Washington, D.C., 1993. Morgan Kaufmann, San Mateo, CA.

[35] D. Heckerman and J. S. Breese. A new look at causal independence. In *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence (UAI'94)*, pages 286–292, Seattle, WA, 1994. Morgan Kaufmann, San Francisco, CA.

[36] D. Heckerman and J. S. Breese. Causal independence for probability assessment and inference using Bayesian networks. *IEEE Transactions on Systems, Man and Cybernetics—Part A: Systems and Humans*, 26:826–831, 1996.

[37] D. Heckerman, J. S. Breese, and K. Rommelse. Decision-theoretic troubleshooting. *Communications of the ACM*, 38:49–57, 1995.

[38] D. E. Heckerman. A tractable inference algorithm for diagnosing multiple diseases. In M. Henrion, R. D. Shachter, L. N. Kanal, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence 5*, pages 163–171. Elsevier Science Publishers, Amsterdam, The Netherlands, 1990.

[39] M. Henrion. Propagation of uncertainty by logic sampling in Bayes' networks. In J. F. Lemmer and L. N. Kanal, editors, *Uncertainty in Artificial Intelligence 2*, pages 149–164. Elsevier Science Publishers, Amsterdam, The Netherlands, 1988.

[40] M. Henrion. Some practical issues in constructing belief networks. In L. N. Kanal, T. S. Levitt, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence 3*, pages 161–173. Elsevier Science Publishers, Amsterdam, The Netherlands, 1989.

[41] R. A. Howard and J. E. Matheson. Influence diagrams. In R. A. Howard and J. E. Matheson, editors, *Readings on the Principles and Applications of Decision Analysis*, pages 719–762. Strategic Decisions Group, Menlo Park, CA, 1984.

[42] R. Jurgelenaite and T. Heskes. EM algorithm for symmetric causal independence models. In *Proceedings of the 17th European Conference on Machine Learning*, pages 234–245, Berlin, Germany, 2006. Springer-Verlag (LNAI 4212), Berlin.

[43] R. Jurgelenaite and T. Heskes. Symmetric causal independence models for classification. In *Proceedings of the Third European Workshop on Probabilistic Graphical Models (PGM'06)*, pages 163–170, Prague, Czech Republic, 2006.

[44] R. Jurgelenaite and P. J. F. Lucas. Exploiting causal independence in large Bayesian networks. *Knowledge-Based Systems*, 18:153–162, 2005.

[45] R. Jurgelenaite, P. J. F. Lucas, and T. Heskes. Exploring the noisy threshold function in designing Bayesian networks. In *Proceedings of the Twenty-Fifth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence (AI-2005)*, pages 133–146, Queens' College, Cambridge, UK, 2005.

[46] C. M. Kadie, D. Hovel, and E. Horvitz. MSBNx: A component-centric toolkit for modeling and inference with Bayesian networks. Technical Report MSR-TR-2001-67, Microsoft Research, Redmon, WA, 2001.

[47] D. Kahneman, P. Slovic, and A. Tversky, editors. *Judgement under Uncertainty: Heuristics and Biases*. Cambridge University Press, Cambridge, UK, 1982.

[48] H. J. Kappen and J. P. Neijt. Promedas, a probabilistic decision support system for medical diagnosis. Technical Report, Foundation for Neural Networks (SNN), Nijmegen, The Netherlands, 2002.

[49] J. H. Kim and J. Pearl. A computational model for combined causal and diagnostic reasoning in inference systems. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence (IJCAI–83)*, pages 190–193, Karlsruhe, Germany, 1983.

[50] U. Kuter, D. Nau, D. Gossink, and J. F. Lemmer. Interactive course-of-action planning using causal models. In M. Pechoucek and A. Tate, editors, *Proceedings of the Third International Conference on Knowledge Systems for Coalition Operations (KSCO-2004)*, pages 37–51, Prague, Czech Republic, 2004.

[51] C. Lacave and F. J. Díez. Explanation for causal Bayesian networks in Elvira. In *Proceedings of the Workshop on Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP-2002)*, pages 43–48, Lyon, France, 2002.

[52] C. Lacave and F. J. Díez. A review of explanation methods for Bayesian networks. *Knowledge Engineering Review*, 17:107–127, 2002.

[53] C. Lacave and F. J. Díez. Knowledge acquisition in Prostanet, a Bayesian network for diagnosing prostate cancer. In *Proceedings of the Seventh International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES'2003)*, volume 2774 of *Lecture Notes in Computer Science*, pages 1345–1350, Oxford, UK, 2003. Springer, Berlin, Germany.

[54] C. Lacave, M. Luque, and F. J. Díez. Explanation of Bayesian networks and influence diagrams in Elvira. *IEEE Transactions on Systems, Man and Cybernetics—Part B: Cybernetics*, 2007. In press.

[55] S. Lauritzen and D. Nilsson. Representing and solving decision problems with limited information. *Management Science*, 47:1238–1251, 2001.

[56] J. F. Lemmer and D. E. Gossink. Recursive noisy-OR: A rule for estimating complex probabilistic causal interactions. *IEEE Transactions on Systems, Man and Cybernetics—Part B: Cybernetics*, 34:2252–2261, 2004.

[57] P. J. F. Lucas. Certainty-factor-like structures in Bayesian belief networks. *Knowledge-Based Systems*, 14:327–335, 2001.

[58] P. J. F. Lucas. Bayesian network modelling through qualitative patterns. *Artificial Intelligence*, 163:233–263, 2005.

[59] P. J. F. Lucas, N. C. de Bruijn, K. Schurink, and A. Hoepelman. A probabilistic and decision-theoretic approach to the management of infectious disease at the ICU. *Artificial Intelligence in Medicine*, 19:251–279, 2000.

[60] C. Meek and D. Heckerman. Structure and parameter learning for causal independence and causal interaction models. In *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence (UAI'97)*, pages 366–375, Providence, RI, 1997. Morgan Kaufmann, San Francisco, CA.

[61] M. G. Morgan and M. Henrion. *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*. Cambridge University Press, Cambridge, UK, 1990.

[62] R. Neal. Connectionist learning of Bayesian networks. *Artificial Intelligence*, 56:71–113, 1992.

[63] K. G. Olesen, U. Kjærulff, F. Jensen, F. V. Jensen, B. Falck, S. Andreassen, and S. K. Andersen. A MUNIN network por the median nerve. A case study on loops. *Applied Artificial Intelligence*, 3:385–403, 1989.

[64] A. Oniśko, M. J. Druzdzel, and H. Wasyluk. Extension of the Hepar II model to multiple-disorder diagnosis. In M. Kłopotek, M. Michalewicz, and S.T. Wierzchoń, editors, *Intelligent Information Systems*, pages 303–313. Springer-Verlag, Heidelberg, 2000.

[65] A. Oniśko, M. J. Druzdzel, and H. Wasyluk. Learning Bayesian network parameters from small data sets: Application of Noisy-OR gates. *International Journal of Approximate Reasoning*, 27:165–182, 2001.

[66] J. Pearl. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29:241–288, 1986.

[67] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.

[68] Y. Peng and J. A. Reggia. Plausibility of diagnostic hypotheses. In *Proceedings of the 5th National Conference on AI (AAAI–86)*, pages 140–145, Philadelphia, PA, 1986.

[69] S. Plous. *The Psychology of Judgment and Decision Making*. McGraw-Hill, New York, 1993.

[70] M. Pradhan, G. Provan, B. Middleton, and M. Henrion. Knowledge engineering for large belief networks. In *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence (UAI'94)*, pages 484–490, Seattle, WA, 1994. Morgan Kaufmann, San Francisco, CA.

[71] S. Renooij. Probability elicitation for belief networks: Issues to consider. *Knowledge Engineering Review*, 16:255–269, 2001.

[72] I. Rish, K. Kask, and R. Dechter. Empirical evaluation of approximation algorithms for probabilistic decoding. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI'98)*, pages 455–463, Madison, WI, 1998. Morgan Kaufmann, San Francisco, CA.

[73] R. D. Shachter. Probabilistic inference and influence diagrams. *Operations Research*, 36:589–605, 1988.

[74] M. A. Shwe, B. Middleton, D. Heckerman, M. Henrion, E. J. Horvitz, H. P. Lehmann, and G. F. Cooper. Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. Part I — The probabilistic model and inference algorithms. *Methods of Information in Medicine*, 30:241–255, 1991.

[75] H. A. Simon. Causal ordering and identifiability. In W. C. Hood and T. C. Koopmans, editors, *Studies in Econometric Method. Cowles Commission for Research in Economics. Monograph No. 14*, chapter III, pages 49–74. John Wiley & Sons, Inc., New York, 1953.

[76] H. A. Simon. *The Sciences of the Artificial*. The MIT Press, Cambridge, MA, 1969.

[77] H. A. Simon. *Models of Discovery*. D. Reidel Publishing Company, Dordrecht, The Netherlands, 1977.

[78] S. Srinivas. A generalization of the noisy-OR model. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, pages 208–215, Washington, D.C., 1993. Morgan Kaufmann, San Mateo, CA.

[79] A. Tversky and D. Kahneman. Judgement under uncertainty: Heuristics and biases. *Science*, 185:1124–1131, 1974.

[80] A. Tversky and D. Kahneman. Causal schemata in judgments under uncertainty. In M. Fishbein, editor, *Progress in Social Psychology*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1977.

[81] L. C. van der Gaag, S. Renooij, C. L. M. Witteman, B. M. P. Aleman, and B. G. Taal. How to elicit many probabilities. In *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence (UAI'99)*, pages 647–654, Stockholm, Sweden, 1999. Morgan Kaufmann, San Francisco, CA.

[82] L. C. van der Gaag, S. Renooij, C. L. M. Witteman, B. M. P. Aleman, and B. G. Taal. Probabilities for a probabilistic network: A case-study in oesophageal cancer. *Artificial Intelligence in Medicine*, 25:123–148, 2002.

[83] M. A. J. van Gerven. Efficient Bayesian inference by factorizing conditional probability distributions. Technical Report ICIS-R6032, Radboud University Nijmegen, Nijmegen, The Netherlands, 2006.

[84] M. A. J. van Gerven, F. J. Díez, B. G. Taal, and P. J. F. Lucas. Selecting treatment strategies with dynamic limited-memory influence diagrams. Submitted to *Artificial Intelligence in Medicine*, 2006.

[85] M. A. J. van Gerven, R. Jurgelenaite, B. G. Taal, T. Heskes, and P. J. F. Lucas. Predicting carcinoid heart disease with the noisy threshold classifier. *Artificial Intelligence in Medicine*, 2006. To appear.

[86] M. A. J. van Gerven, P. J. F. Lucas, and T. P. van der Weide. Qualitative characterization of causal independence models. Submitted to *International Journal of Approximate Reasoning*, 2006.

[87] J. Vomlel. Exploiting functional dependence in Bayesian network inference with a computerized adaptive test as an example. In *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence (UAI'02)*, pages 528–535, Edmonton, Canada, 2002. Morgan Kaufmann, San Francisco, CA.

[88] J. W. Wallis and E. H. Shortliffe. Customized explanations using causal knowledge. In B. G. Buchanan and E. H. Shortliffe, editors, *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, chapter 20, pages 371–388. Addison-Wesley, Reading, MA, 1984.

[89] I. Wegener. *The Complexity of Boolean Functions*. John Wiley & Sons, New York, 1987.

[90] M. P. Wellman. Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence*, 44:257–303, 1990.

[91] M. P. Wellman and M. Henrion. Explaining "explaining away". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:287–292, 1993.

[92] C.L.M. Witteman and S. Renooij. Evaluation of a verbal-numerical probability scale. *International Journal of Approximate Reasoning*, 33:117–131, 2003.

[93] A. Zagorecki and M. J. Druzdzel. An empirical study of probability elicitation under Noisy-OR assumption. In V. Barr and Z. Markov, editors, *Proceedings of the 17th International Florida Artificial Intelligence Research Society Conference (FLAIRS–2004)*, pages 880–885, Miami Beach, FL, 2004. AAAI Press, Menlo Park, CA.

[94] A. Zagorecki and M. J. Druzdzel. Knowledge engineering for Bayesian networks: How common are noisy-MAX distributions in practice? In G. Brewka, S. Coradeschi, A. Perini, and P. Traverso, editors, *Proceedings of the Seventeenth European Conference on Artificial Intelligence (ECAI-06)*, pages 482–489, Riva del Garda, Italy, 2006. IOS Press, Amsterdam, The Netherlands.

[95] A. Zagorecki and M. J. Druzdzel. Probabilistic independence of causal influences. In *Proceedings of the Third European Workshop on Probabilistic Graphical Models (PGM'06)*, pages 325–332, Prague, Czech Republic, 2006.

[96] A. Zagorecki, M. Voortman, and M. J. Druzdzel. Decomposing local probability distributions in Bayesian networks for improved inference and parameter learning. In G. Sutcliffe and R. Goebel, editors, *Proceedings of the 19th International Florida Artificial Intelligence Research Society Conference (FLAIRS–2006)*, pages 860–865, Melbourne Beach, FL, 2006. AAAI Press, Menlo Park, CA.

[97] N. L. Zhang and D. Poole. Exploiting causal independence in Bayesian network inference. *Journal of Artificial Intelligence Research*, 5:301–328, 1996.