

Dawn E. Holmes and Lakhmi C. Jain (Eds.)

Innovations in Bayesian Networks

Studies in Computational Intelligence, Volume 156

Editor-in-Chief

Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw
Poland
E-mail: kacprzyk@ibspan.waw.pl

Further volumes of this series can be found on our homepage:
springer.com

Vol. 134. Ngoc Thanh Nguyen and Radoslaw Katarzyniak (Eds.)
New Challenges in Applied Intelligence Technologies, 2008
ISBN 978-3-540-79354-0

Vol. 135. Hsinchun Chen and Christopher C. Yang (Eds.)
Intelligence and Security Informatics, 2008
ISBN 978-3-540-69207-2

Vol. 136. Carlos Cotta, Marc Sevaux
and Kenneth Sörensen (Eds.)
Adaptive and Multilevel Metaheuristics, 2008
ISBN 978-3-540-79437-0

Vol. 137. Lakhmi C. Jain, Mika Sato-Ilic, Maria Virvou,
George A. Tsihrintzis, Valentina Emilia Balas
and Canicious Abeynayake (Eds.)
Computational Intelligence Paradigms, 2008
ISBN 978-3-540-79473-8

Vol. 138. Bruno Apolloni, Witold Pedrycz, Simone Bassi
and Dario Malchiodi
The Puzzle of Granular Computing, 2008
ISBN 978-3-540-79863-7

Vol. 139. Jan Drugowitsch
Design and Analysis of Learning Classifier Systems, 2008
ISBN 978-3-540-79865-1

Vol. 140. Nadia Magnenat-Thalmann, Lakhmi C. Jain
and N. Ichalkaranje (Eds.)
New Advances in Virtual Humans, 2008
ISBN 978-3-540-79867-5

Vol. 141. Christa Sommerer, Lakhmi C. Jain
and Laurent Mignonneau (Eds.)
The Art and Science of Interface and Interaction Design (Vol. 1),
2008
ISBN 978-3-540-79869-9

Vol. 142. George A. Tsihrintzis, Maria Virvou, Robert J. Howlett
and Lakhmi C. Jain (Eds.)
New Directions in Intelligent Interactive Multimedia, 2008
ISBN 978-3-540-68126-7

Vol. 143. Uday K. Chakraborty (Ed.)
Advances in Differential Evolution, 2008
ISBN 978-3-540-68827-3

Vol. 144. Andreas Fink and Franz Rothlauf (Eds.)
*Advances in Computational Intelligence in Transport, Logistics,
and Supply Chain Management*, 2008
ISBN 978-3-540-69024-5

Vol. 145. Mikhail Ju. Moshkov, Marcin Piliszczuk
and Beata Zielosko
Partial Covers, Reducts and Decision Rules in Rough Sets, 2008
ISBN 978-3-540-69027-6

Vol. 146. Fatos Xhafa and Ajith Abraham (Eds.)
*Metaheuristics for Scheduling in Distributed Computing
Environments*, 2008
ISBN 978-3-540-69260-7

Vol. 147. Oliver Kramer
Self-Adaptive Heuristics for Evolutionary Computation, 2008
ISBN 978-3-540-69280-5

Vol. 148. Philipp Limbourg
Dependability Modelling under Uncertainty, 2008
ISBN 978-3-540-69286-7

Vol. 149. Roger Lee (Ed.)
*Software Engineering, Artificial Intelligence, Networking and
Parallel/Distributed Computing*, 2008
ISBN 978-3-540-70559-8

Vol. 150. Roger Lee (Ed.)
*Software Engineering Research, Management and
Applications*, 2008
ISBN 978-3-540-70774-5

Vol. 151. Tomasz G. Smolinski, Mariofanna G. Milanova
and Aboul-Ella Hassanien (Eds.)
Computational Intelligence in Biomedicine and Bioinformatics,
2008
ISBN 978-3-540-70776-9

Vol. 152. Jarosław Stepaniuk
*Rough – Granular Computing in Knowledge Discovery and Data
Mining*, 2008
ISBN 978-3-540-70800-1

Vol. 153. Carlos Cotta and Jano van Hemert (Eds.)
*Recent Advances in Evolutionary Computation for
Combinatorial Optimization*, 2008
ISBN 978-3-540-70806-3

Vol. 154. Oscar Castillo, Patricia Melin, Janusz Kacprzyk and
Witold Pedrycz (Eds.)
Soft Computing for Hybrid Intelligent Systems, 2008
ISBN 978-3-540-70811-7

Vol. 155. Hamid R. Tizhoosh and M. Ventresca (Eds.)
Oppositional Concepts in Computational Intelligence, 2008
ISBN 978-3-540-70826-1

Vol. 156. Dawn E. Holmes and Lakhmi C. Jain (Eds.)
Innovations in Bayesian Networks, 2008
ISBN 978-3-540-85065-6

Dawn E. Holmes
Lakhmi C. Jain
(Eds.)

Innovations in Bayesian Networks

Theory and Applications

Prof. Dawn E. Holmes
Department of Statistics and Applied Probability
University of California
Santa Barbara, CA 93106
USA
Email: holmes@pstat.ucsb.edu

Prof. Lakhmi C. Jain
Professor of Knowledge-Based Engineering
University of South Australia
Adelaide
Mawson Lakes, SA 5095
Australia
Email: Lakhmi.jain@unisa.edu.au

ISBN 978-3-540-85065-6

e-ISBN 978-3-540-85066-3

DOI 10.1007/978-3-540-85066-3

Studies in Computational Intelligence

ISSN 1860949X

Library of Congress Control Number: 2008931424

© 2008 Springer-Verlag Berlin Heidelberg

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Typeset & Cover Design: Scientific Publishing Services Pvt. Ltd., Chennai, India.

Printed in acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

For:

COLIN and HELEN

Preface

There are many invaluable books available on Bayesian networks. However, in compiling a volume titled “Innovations in Bayesian networks” we wish to introduce some of the latest developments to a broad audience of both specialists and non-specialists in this field.

So, what are Bayesian networks? Bayesian networks utilize the probability calculus together with an underlying graphical structure to provide a theoretical framework for modeling uncertainty. Although the philosophical roots of the subject may be traced back to Bayes and the foundations of probability, Bayesian networks as such are a modern device, first appearing in Pearl (1988), and growing out of the research in expert or intelligent systems. In Pearl’s ground-breaking *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, (1988), he presents Bayesian networks for the first time. Since then, Bayesian networks have become the vehicle of choice in many AI applications.

In compiling this volume we have sought to present innovative research from prestigious contributors in the field of Bayesian networks. Each chapter is self-contained and is described below.

Chapter 1 by Holmes presents a brief introduction to Bayesian networks for readers entirely new to the field.

Chapter 2 by Neapolitan, a self-proclaimed convert to Bayesianism, discusses the modern revival of Bayesian statistics research, due in particular to the advent of Bayesian networks. Bayesian and frequentist approaches are compared, with an emphasis on interval estimation and hypothesis testing.

For Chapter 3 we are indebted to MIT Press and Kluwer books for permission to reprint Heckerman’s famous tutorial on learning with Bayesian networks, in which the reader is introduced to many of the techniques fundamental to the success of this formalism.

In Chapter 4, Korb and Nicholson discuss some of the philosophical problems associated with Bayesian networks. In particular, they introduce a causal interpretation of Bayesian networks thus providing a valuable addition to the currently lively and productive research in causal modeling.

Chapter 5 by **Niedermayer** explores how Bayesian Network theory affects issues of advanced computability. Some interesting applications together with a brief discussion of the appropriateness and limitations of Bayesian Networks for human-computer interaction and automated learning make this a strong contribution to the field.

In Chapter 6 Nagl, Williams and Williamson introduce and discuss Objective Bayesian nets and their role in knowledge integration. Their importance for medical informatics is emphasized and a scheme for systems modeling and prognosis in breast cancer is presented.

Chapter 7 by Jiang, Wagner, and Cooper considers epidemic modeling. Epidemic curves are defined and a Bayesian network model for real-time estimation of these curves is given. An evaluation of the experimental results and their accuracy rounds off this valuable contribution.

Chapter 8 by Lauría begins with an excellent introduction to structure learning in Bayesian Networks. The paper continues by exploring the feasibility of applying an information-geometric approach to the task of learning the topology of Bayesian network and also provides an introduction to information geometry.

Chapter 9 by Maes, Leray and Meganck discusses causal graphical models for discrete variables that can handle latent variables without explicitly modeling them quantitatively. The techniques introduced in this chapter have been partially implemented into the structure learning package (SLP) of the Bayesian networks toolbox (BNT) for MATLAB.

Chapter 10 by Flores, Gamez, and Moral presents a new approach to the problem of obtaining the most probable explanations given a set of observations in a Bayesian network. Examples are given and a set of experiments to make a comparison with other existing abductive techniques that were designed with goals similar to those we pursue is described.

Chapter 11 by Holmes describes a maximum entropy approach to inference in Bayesian networks. We are indebted to the American Institute of Physics for kind permission to reproduce this paper.

Chapter 12 by de Salvo Braz, Amir, and Roth presents a survey of first-order probabilistic models. The decision on using directed or undirected models is discussed, as are infinite models. The authors conclude that such algorithms would benefit from choosing to process first the parts of the model that yield greater amounts of information about the query.

This book will prove valuable to theoreticians as well as application scientists/engineers in the area of Bayesian networks. Postgraduate students will also find this a useful sourcebook since it shows the direction of current research.

We have been fortunate in attracting top class researchers as contributors and wish to offer our thanks for their support in this project. We also acknowledge the expertise and time of the reviewers. Finally, we also wish to thank Springer for their support.

Dr Dawn E. Holmes
University of California
Santa Barbara, USA

Dr Lakhmi C. Jain
University of South Australia
Adelaide, Australia

Contents

1 Introduction to Bayesian Networks <i>Dawn E. Holmes, Lakhmi C. Jain</i>	1
2 A Polemic for Bayesian Statistics <i>Richard E. Neapolitan</i>	7
3 A Tutorial on Learning with Bayesian Networks <i>David Heckerman</i>	33
4 The Causal Interpretation of Bayesian Networks <i>Kevin B. Korb, Ann E. Nicholson</i>	83
5 An Introduction to Bayesian Networks and Their Contemporary Applications <i>Daryle Niedermayer, I.S.P.</i>	117
6 Objective Bayesian Nets for Systems Modelling and Prognosis in Breast Cancer <i>Sylvia Nagl, Matt Williams, Jon Williamson</i>	131
7 Modeling the Temporal Trend of the Daily Severity of an Outbreak Using Bayesian Networks <i>Xia Jiang, Michael M. Wagner, Gregory F. Cooper</i>	169
8 An Information-Geometric Approach to Learning Bayesian Network Topologies from Data <i>Eitel J.M. Lauría</i>	187
9 Causal Graphical Models with Latent Variables: Learning and Inference <i>Sam Maes, Philippe Leray, Stijn Meganck</i>	219

10 Use of <i>Explanation Trees</i> to Describe the State Space of a Probabilistic-Based Abduction Problem	
<i>M. Julia Flores, José A. Gámez, Serafín Moral</i>	251
11 Toward a Generalized Bayesian Network	
<i>Dawn E. Holmes</i>	281
12 A Survey of First-Order Probabilistic Models	
<i>Rodrigo de Salvo Braz, Eyal Amir, Dan Roth</i>	289
Author Index	319

Introduction to Bayesian Networks

Dawn E. Holmes¹ and Lakhmi C. Jain²

¹ Department of Statistics and Applied Probability, University of California,
Santa Barbara, USA
holmes@pstat.ucsb.edu

² Knowledge-Based Intelligent Engineering Systems (KES) Centre,
University of South Australia, Adelaide, Mawson Lakes, S.A. 5095, Australia
lakhmi.jain@unisa.edu.au

Abstract. Reasoning with incomplete and unreliable information is a central characteristic of decision making, for example in industry, medicine and finance. Bayesian networks provide a theoretical framework for dealing with this uncertainty using an underlying graphical structure and the probability calculus. Bayesian networks have been successfully implemented in areas as diverse as medical diagnosis and finance. We present a brief introduction to Bayesian networks for those readers new to them and give some pointers to the literature.

1.1 Introduction

Reasoning under uncertainty has a long history and is a major issue in artificial intelligence. Several formalisms have been developed for dealing with uncertainty including fuzzy logic, non-monotonic reasoning, Dempster-Shafer theory, possibilistic logic and probability theory. The choice of approach is usually governed by the nature of the problem, rather than by any commitment to a particular formal approach. However, since probability theory is mathematically sound, it provides the ideal formalism.

The theory of Bayesian networks derives, ultimately from the work of Thomas Bayes. In 1764 Bayes 'An essay towards solving a problem in the doctrine of chances', was published posthumously in the *Philosophical Transactions of the Royal Society of London*, 53:370- 418; this landmark paper contains a special case of Bayes' Theorem, which is concerned with conditional probabilities.

1.2 Reasoning under Uncertainty

The original AI problem in computer science was to produce a computer program that displayed aspects of intelligent human behavior. There was general agreement that one of the most important intelligent activities was problem solving, which seemed a computationally feasible activity. Initial attempts to produce general-purpose problem solving programs had limited success, due to problems with knowledge representation and complex algorithms requiring exponential processing time. Perhaps the most successful of these was GPS (Newell and Simon. 1972) which used a state transformation method. It was soon realized that experts worked in closed domains and that knowledge outside the relevant domain could be ignored. The importance of domain specific knowledge became apparent, and the expert or intelligent system was

developed. There still remained the problem of finding a computationally feasible algorithm and propagation seemed to provide the answer. In addition, the underlying philosophy of replacing the expert by something better was replaced by the more acceptable notion of expert system supported decision-making.

It has been argued to the contrary that the idea of using classical probability theory in expert systems should be abandoned due to the computational overload and knowledge elicitation problems (Gorry G. and Barnett G. 1968). In an attempt to overcome these problems, Prospector, (Duda R.O., *et al* 1976) another early system loosely based on probability, was developed. Although Prospector oversimplified the mathematics involved, it was sufficiently successful to motivate further research into the use of probability in expert systems and subsequently Pearl found a computationally tractable method using Bayesian networks where probabilities are propagated through the graphical structure (Pearl 1988). The application of Pearl's work resulted in the expert system MUNIN (Muscle and Nerve Inference Network) (Andreasson S *et al* 1987) and from this, the shell HUGIN (Handling Uncertainty in General Inference Networks) (Andreasson S.K *et al* 1989) followed, demonstrating one of the main strengths of Bayesian networks; their ability to cope with large joint distributions.

1.3 Bayesian Networks

Informally, Bayesian networks are graphical models of causal relationships in a given domain. Formally, a Bayesian network is defined as follows.

Let \mathbf{V} be a finite set of vertices and \mathbf{B} a set of directed edges between vertices with no feedback loops, the vertices together with the directed edges form a directed acyclic graph $\mathbf{G} = \langle \mathbf{V}, \mathbf{B} \rangle$. A set of events is depicted by the vertices of \mathbf{G} and hence also represented by \mathbf{V} . Let each event have a finite set of mutually exclusive outcomes, where E_i is a variable which can take any of the outcomes e_i^j of the event i , $j = 1, \dots, n_i$. Let \mathbf{P} be a probability distribution over the combinations of events. Let \mathbf{C} be the following set of constraints:

- (i) the requirement that a probability distribution sums to unity.
- (ii) for each event i with a set of parents M_i there are associated conditional probabilities $P\left(E_i \mid \bigwedge_{j \in M_i} E_j\right)$ for each possible outcome that can be assigned to E_i and the E_j .
- (iii) those independence relationships implied by d -separation in the directed acyclic graph

Then $\mathbf{N} = \langle \mathbf{G}, \mathbf{P}, \mathbf{C} \rangle$ is a causal network if \mathbf{P} has to satisfy \mathbf{C} .

D -separation, so-called in contrast to the analogous separation property of undirected graphs, constitutes a set of rules devised by Pearl, which provide a means for determining the independence or dependence of any pair of variables in a Bayesian

network. The theorems arising from d -separation, taken together with the chain rule, are used by Pearl to determine the joint probability distribution of the system being considered.

1.3.1 Propagation through Local Computation

There are several methods of updating probabilities in Bayesian networks. Pearl devised a method for propagating probabilities through a singly connected causal network (Pearl 1988) and this method, where probabilities are updated locally, was implemented in the expert system MUNIN. Pearl argues that this approach models some of the necessary building blocks of human reasoning, but philosophical speculation aside; there are technical reasons in favor of adopting local updating. These are largely to do with implementation; for example, the separation of inference engine from knowledge-base make it possible to build re-usable shells. Pearl has achieved his objective for singly connected networks; propagation can be performed in linear time by Pearl's algorithm and so it is a very useful method. However, it must be emphasized that it only works on networks that are singly connected. It has been shown that for multiply-connected networks the problem of determining the probability distribution is NP -hard (Cooper G.F. 1990).

Following on from Pearl's work, theoretical refinements resulted in the algorithm developed by Lauritzen and Spiegelhalter, which has been successfully used in commercially available expert systems, e.g. from the HUGIN shell (Lauritzen S.L. and Spiegelhalter D.J. 1988). In contrast to Pearl's approach, Lauritzen and Spiegelhalter treated the problem of propagating probabilities in Bayesian networks as wholly mathematical. Their method, based on graph theory, works for any Bayesian network. No attempt was made to model human reasoning but since there is no reason to suppose that machine reasoning mimics human reasoning this cannot be considered a drawback.

The main problem in constructing an algorithm for updating the conditional probabilities in a Bayesian network is that the number of terms involved in a global calculation grows exponentially with the number of nodes in the network. Lauritzen and Spiegelhalter's method overcomes this problem by using a technique involving local computations only and provides a standard method of updating which can be used once the priors have all been found. The requirement of complete knowledge regarding the prior distribution highlights the limitation of causal networks as a vehicle for modeling problem domains. Jensen points out that there is often no theoretical way of determining all the required probabilities and gives examples of how they are ascertained in practice Jensen (1996). Sometimes they are guessed, sometimes a complex and subjective procedure is gone through in order to produce an approximate and necessarily biased value. When multivalued events are to be modeled, the situation becomes complex. In some situations, there are inevitably too many conditional probabilities for an expert to reliably estimate. Thus, the need for a theoretically sound technique for estimating them in a minimally prejudiced fashion becomes apparent. The maximum entropy formalism provides just such a technique. However, Maung and Paris have shown that the general problem of finding the maximum entropy solution in probabilistic systems is NP -complete; generally there are 2^n variables to consider and it is thus clearly infeasible to find the probability distribution of all the

state probabilities (Maung I. and Paris J.B. 1990). Therefore, if we are to apply this methodology to causal networks, we must show that the estimates for missing information can be found efficiently.

1.3.2 Inference in Bayesian Networks

There are several well-known methods of exact inference in Bayesian networks: variable elimination and clique tree propagation being particularly popular. The methods of approximation most used are stochastic MCMC simulation and bucket elimination.

1.3.3 Learning in Bayesian Networks

A very useful and freely available resource on learning in Bayesian networks is Andrew Moore's website:

<http://www.cs.cmu.edu/~awm/tutorials>.

See also Heckerman, this volume.

1.4 Applications

Practically, graphical models are appealing since they provide a bridge between the user and the knowledge engineer. It is particularly important from the user's point of view that they can be constructed gradually and so complex models can be built over a period of time. For the knowledge engineer their appeal lies, at least in part, in that the data structures are already recognizable. Among the more well-known applications, The Pathfinder Project (Heckerman, Horvitz, Nathwani 1992), resulted in a system for diagnosing diseases of the lymph nodes that outperforms expert diagnosis. The AutoClass project is an unsupervised Bayesian classification system. Microsoft's Office Assistant, a result of the Lumiere Project, is also the result of a Bayesian network. See, for example, Niedermayer, this volume.

1.5 Selective Bibliography

Below we mention just a few of the books and papers that a newcomer to the field of Bayesian networks may find useful.

Pearl, J.: Probabilistic Reasoning in Intelligent Systems. In: Networks of Plausible Inference. Morgan Kaufmann Publishers, San Francisco (1988)

Lauritzen, S.L., Spiegelhalter, D.J.: Local Computations with Probabilities on Graphical Structures and their Applications to Expert Systems. *J. Royal Statist.Soc. B* 50(2), 154–227 (1988)

Andreasson, S., Woldbye, M., Falck, B., Andersen, S.K.: MUNIN - A Causal Probabilistic Network for Interpretation of Electromyographic Findings. In: Proc. of the 10th Int. Joint Conference on AI., Milan, pp. 366–372 (1987)

Andreasson, S.K., Oleson, K.G., Jensen, F.V., Jensen, F.: HUGIN- A Shell for Building Bayesian Belief Universes for Expert Systems. In: Proc. of the 11th Int. Joint Conference on AI., Detroit (1989)

- Klir, G.J., Folger, T.A.: Fuzzy Sets. In: Uncertainty and Information. Prentice-Hall, Englewood Cliffs (1995)
- Shannon, C.E., Weaver, W.: The Mathematical Theory of Communication. University of Illinois Press (1948)
- Neapolitan, R.E.: Probabilistic Reasoning in Expert Systems. John Wiley, Chichester (1990)
- Jensen Finn, V.: An Introduction to Bayesian Networks. UCL Press (1996)
- Cooper, G.F.: The Computational Complexity of Probabilistic Inference Using Bayesian Belief Networks. *Artificial Intelligence* 42, 393–405 (1990)

A Polemic for Bayesian Statistics

Richard E. Neapolitan

Northeastern Illinois University, Chicago, IL 60625
RE-Neapolitan@neiu.edu

Abstract. In the early part of the 20th century, forefathers of current statistical methodology were largely Bayesians. However, by the mid-1930's the Bayesian method fell into disfavor for many, and frequentist statistics became popular. Seventy years later the frequentist method continues to dominate. My purpose here is to compare the Bayesian and frequentist approaches. I argue for Bayesian statistics claiming the following: 1) Bayesian methods solve a wider variety of problems; and 2) It is sometimes difficult to interpret frequentist results.

2.1 Introduction

In the early part of the 20th century, forefathers of current statistical methodology were largely Bayesians (e.g. R.A. Fisher and Karl Pearson). However, by the mid-1930's the Bayesian method fell into disfavor for many, and frequentist statistics, in particular the use of confidence intervals and the rejection of null hypotheses using p -values, became popular. Seventy years later the frequentist method dominates, and most university statistics courses present this approach without even a reference to Bayesian methods. However, there has recently been a resurgence of interest in Bayesian statistics, partly due to the use of Bayesian methods in Bayesian networks ([Pearl, 1988], [Neapolitan, 1990]) and in machine learning. Curiously, sometimes systems that learn using Bayesian methods are evaluated using frequentist statistics (e.g. in [Buntine, 1992]). This attests to the dominance of frequentist statistics. That is, it seems researchers in Bayesian machine learning learned frequentist evaluation methods first and therefore adhere to them.

My purpose here is to compare the Bayesian and frequentist approaches focusing on interval estimation and hypothesis testing. I make no effort to cover the frequentist or Bayesian method in any detail; rather I show only sufficient mathematics and examples to contrast the two approaches. I argue for Bayesian statistics claiming the following: 1) Bayesian methods solve a wider variety of problems; and 2) It is sometimes difficult to interpret frequentist results, particularly in the case of hypothesis testing, whereas the interpretation of Bayesian results is straightforward and meaningful. Bolstad [2004] solves many more problems using the two approaches, while Zabell [1992] presents some of the history of statistics over the 20th century, focusing on the life of R.A. Fisher. I

assume the reader is familiar with both the Bayesian and frequentist approaches to probability. See [Neapolitan, 2004] for a brief, facile discussion of the two.

After methods for interval estimation are compared, hypothesis testing is discussed.

2.2 Interval Estimation

A **random sample** consists of independent, identically distributed random variables X_1, X_2, \dots, X_n . In practice a random sample is obtained by selecting individuals (or items) at random from some population.

Example 1. Suppose we select 100 American males at random from the population of all American males. Let X_i be a random variable whose value is the height of the i th individual. If, for example, the 10th individual is 71 inches, then the value of X_{10} is 71.

Mathematically, the **population** is really the collection of all values of the variable of interest. For example, if our set of entities is {George, Sam, Dave, Clyde}, their heights are 68 inches, 70 inches, 68 inches, and 72 inches, and the variable of interest is height, then the population is the collection [68, 70, 68, 72]. However, it is common to refer to the collection of all entities as the population.

A common task is to assume that the random variables are normally distributed, obtain values x_1, x_2, \dots, x_n of X_1, X_2, \dots, X_n , estimate the true population mean (expected value) μ by

$$\bar{x} = \sum_{i=1}^n x_i,$$

and obtain some measure of confidence as to how good this estimate is.

Example 2. Suppose we sample 100 American males and the average value of the height turns out to be 70 inches. Then $\bar{x} = 70$, and our goal is to obtain some measure of confidence as to how close the true average height μ is to 70.

Ordinarily, we don't know the variance either. However, for the sake of simplicity, in the discussion here we will assume the variance is known. Although it is elementary statistics, we will review the classical statistical technique for obtaining a confidence interval so that we can compare the method to Fisher's fiducial probability interval, and the Bayesian probability interval.

2.2.1 Confidence Intervals and Fiducial Probability Intervals

First denote the normal density function and normal distribution as follows:

$$\text{NormalDen}(x; \mu, \sigma^2) \equiv \frac{1}{2\pi\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

$$\text{NormalDist}(\xi; \mu, \sigma^2) \equiv \int_{-\infty}^{\xi} \text{NormalDen}(x; \mu, \sigma^2) dx.$$

Next we have the following lemma, whose proof can be found in most classical statistics texts.

Lemma 1. *Suppose we have a random sample X_1, X_2, \dots, X_n , where each X_i has density function $N(x_i : \mu, \sigma^2)$. Then the random variable \bar{X} , which is the average of the n random variables, has density function*

$$\text{NormalDen}(n; \mu, \sigma^2/n).$$

Intuitively, the average has the same mean as the variables in the sample, but a smaller variance. For example, if we are sampling one individual there will be a lot of variance in the height obtained. However, if we are sampling 1000 individuals and taking the average of the heights, there will be much less variance in the average obtained. Due to the result in this lemma, we have that

$$\begin{aligned} P(\bar{X} < \xi) &= \text{NormalDist}(\xi; \mu, \sigma^2/n) \\ &= \text{NormalDist}\left(\frac{\xi - \mu}{\sigma/\sqrt{n}}; 0, 1\right). \end{aligned}$$

Recall $\text{NormalDist}(x; 0, 1)$ is called the **standard normal distribution**. Now suppose we want to know the value of ξ such that $P(\bar{X} < \xi) = 1 - \alpha$. We set

$$\text{NormalDist}\left(\frac{\xi - \mu}{\sigma/\sqrt{n}}; 0, 1\right) = 1 - \alpha,$$

which means

$$\frac{\xi - \mu}{\sigma/\sqrt{n}} = z_\alpha,$$

where $z_\alpha \equiv \text{NormalDist}^{-1}(1 - \alpha)$. We then have

$$\xi = \mu + z_\alpha \frac{\sigma}{\sqrt{n}}.$$

We therefore conclude that

$$P(\bar{X} - z_\alpha \frac{\sigma}{\sqrt{n}} < \mu) = 1 - \alpha.$$

Proceeding in the same way we can show that

$$P(\bar{X} - z_{\alpha/2} \frac{\sigma}{\sqrt{n}} < \mu < \bar{X} + z_{\alpha/2} \frac{\sigma}{\sqrt{n}}) = 1 - \alpha. \quad (2.1)$$

Example 3. Suppose in a sample of size $n = 100$ the value \bar{x} of \bar{X} is 70, and we assume $\sigma = 4$. Let $\alpha = .05$. Then

$$\bar{x} - z_{\alpha/2} \frac{\sigma}{\sqrt{n}} = 70 - 1.96 \frac{4}{\sqrt{100}} = 67.52$$

$$\bar{x} + z_{\alpha/2} \frac{\sigma}{\sqrt{n}} = 70 + 1.96 \frac{4}{\sqrt{100}} = 72.48.$$

Our purpose was not to review elementary statistics. Rather we wanted to show the mathematics involved in obtaining the interval (67.52, 72.48) so we could discuss the meaning we could attach to the interval. In 1937 Jerzy Neyman noted that, if we take a relatively frequency approach to probability, owing to Equality 2.1, in 95% of trials the interval obtained will contain the mean μ , and in only 5% of trials it will not (By 'trial' I mean a single occurrence of sampling n individuals and taking the average their heights.). So Neyman called (67.52, 72.48) a 95% **confidence interval**. Specifically, he said the following:

The functions $\underline{\theta}(E)$ and $\bar{\theta}(E)$ satisfying the above conditions will be called the lower and upper confidence limits of θ_1 . The value α of the probability will be called the confidence coefficient, and the interval, say $\delta(E)$, from $\underline{\theta}(E)$ to $\bar{\theta}(E)$, the confidence interval corresponding to the confidence coefficient α .

- Jerzy Neyman [1937, p. 348]

In general,

$$\left(\bar{x} - z_{\alpha/2} \frac{\sigma}{\sqrt{n}}, \bar{x} + z_{\alpha/2} \frac{\sigma}{\sqrt{n}} \right) \quad (2.2)$$

is called a $100(1 - \alpha)\%$ **confidence interval** for the unknown mean μ .

Note that Neyman was not saying that the probability that μ is in (67.52, 72.48) is .95. That is, μ is a parameter (constant) and not a random variable in the probability space, and therefore, according to Neyman's theory of probability, we cannot make probabilistic statements about μ . This is why Neyman coined the new term 'confidence'. Furthermore, Neyman was not making the statement that we can be confident that the particular interval obtained in this trial contains the unknown parameter, but rather that over all trials over all distributions, we can have 95% confidence that we will obtain an interval containing the parameter. The following statement makes this clear:

During a period of time the statistician may deal with a thousand problems of estimation and in each case the parameter θ_1 to be estimated and the probability law of the X's may be different. As far as in each case the functions $\underline{\theta}(E)$ to $\bar{\theta}(E)$ are properly calculated and correspond to the same value of α , his steps (a), (b), and (c), though different in details of sampling and arithmetic, will have this in common - the probability of their resulting in a correct statement will be the same, α . Hence the frequency of actually correct statements will approach α

... Can we say that in this particular case the probability of the true value of θ_1 falling between 1 and 2 is equal to α ?

The answer is obviously negative. The parameter θ_1 is an unknown constant and no probability statement concerning its value may be made.

- Jerzy Neyman [1937, p. 349]

Yet many modern frequentist texts apply the concept to a single trial. For example, Anderson et al. [2005] state 'Because 95% of all the intervals constructed

using $\bar{x} \pm 3.92$ will contain the population mean, we say that we are 95% confident that the interval 78.08 to 85.92 includes the population mean μ .⁷

R.A. Fisher was at one time a Bayesian, but he abandoned the approach in favor of the frequentist approach. However, he felt we should still make a probabilistic statement about μ , but do so without making any prior assumptions, even ones about prior ignorance. He noted that if we say the probability that μ is in the interval is .95, this is a true probability in the relative frequency sense because 95% of the time μ will be in the interval. Like Neyman, he is not saying that the probability that μ is in a particular interval obtained in a given trial is .95, but rather over all trials it will be in the interval generated 95% of the time. He called this a **fiducial probability**. Specifically, he said the following

We may express this relationship by saying that the true value of θ will be less than the fiducial 5 per cent value corresponding to the observed value of T in exactly 5 trials in 100. ... This then is a definite probability statement about the unknown parameter θ which is true irrespective of any assumption as to its *a priori* distribution.

- R.A. Fisher [1930, p. 533]

It's not clear (at least to me) whether Fisher meant over all trials for a particular θ , or over all trials over all possible θ (that is, all normal distributions). Both statements are true. If he meant the latter it seems he is just stating the same thing as Neyman in a slightly different way, which critics noted. If he meant the former, the distinction between his interpretation and Neyman's is more clear. In any case, Neyman's confidence interval won out and is the current interpretation and terminology used in frequentist statistics. Zabell [1992] provides an excellent discussion of Fisher's fiduciary probabilities, how Fisher's description of them changed over time from the one we gave above, and why they failed to gain favor.

2.2.2 Probability Intervals

The Bayesian approach to representing uncertainty in the value of an unknown parameter is somewhat different. The fundamental difference between the Bayesian and frequentist approach is that the Bayesian represents uncertainty in an outcome of a single experiment or in a variable having a particular value with a probability of that outcome or that value, whereas the frequentist reserves the term probability for the relative frequency of occurrence of an outcome in repeated trials of an experiment. Bayesians call probabilities in the frequentist's sense **relative frequencies**, and probabilities in their sense **beliefs**. I have reasoned elsewhere for the Bayesian interpretation of probability and the use of probability to represent belief (See [Neapolitan, 1990, 2004]).

The Bayesian, therefore, represents uncertainty concerning the value of an unknown parameter with a **higher order probability distribution** in which the parameter is a random variable. For example, if we are going to repeatedly do the experiment of tossing a thumb tack, we can assume there is some relative

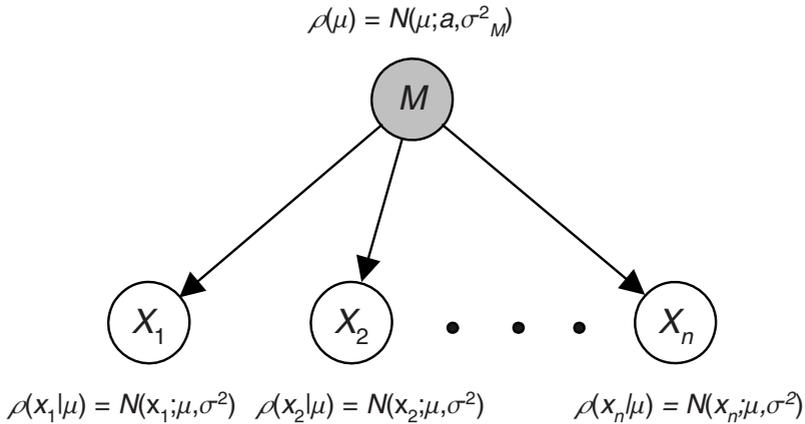


Fig. 2.1. A Bayesian network representing the assumption that the X_i s are independent conditional on M . We have simply used N rather than `NormalDen` to denote the normal density function.

frequency p (frequentist probability) with which it will land ‘heads’ (on its flat end rather than with the edge of the flat end and the point touching the ground). So $P(\text{Heads}) = p$ is an unknown parameter in the probability space determined by this experiment. We can represent our uncertainty as to this parameter’s value by creating a probability distribution of p . If we thought all values of p were equally like, we could use the uniform prior density function. That is, we set $\rho(p) = 1$ over the interval $[0, 1]$. Before we toss the thumb tack, our probability distribution of p is called our **prior probability distribution** of p . After tossing it a number of times, the updated probability distribution, based on the prior distribution and results of the tosses, is called our **posterior probability distribution** of p .

In the case of the unknown mean μ of a normal distribution, we can represent our uncertainty as to the value of μ using a higher order normal distribution. That is, we represent the unknown mean μ by a random variable M , whose space consists of the possible values of μ , and which has density function

$$\text{NormalDen}(\mu; a, \sigma_M^2).$$

Note that for this distribution the expected value of M is a and the variance of M is σ_M^2 . Given a random sample X_1, X_2, \dots, X_n , we assume the X_i are independent conditional on the value of M . The Bayesian network in Figure 2.1 represents this assumption. This assumption is more readily explained with the example of tossing a thumbtack. If we knew the relative frequency of heads was, for example, .3, our belief in heads for the outcome of any particular toss would be .3 regardless of the outcomes of the other tosses. So the outcomes are independent conditional on the value of $P(\text{Heads})$. However, if we did not know $P(\text{Heads})$, and the first 8 tosses were heads, we would consider it more

likely that the thumbtack has a propensity to land head often, and increase our belief in heads for the 9th toss.

The theorem that follows gives us the posterior distribution of M given a random sample. In this theorem, we will find it more convenient to reference the precision of a normal distribution rather than the variance. If σ^2 is the variance of a normal distribution than the precision r is given by

$$r = \frac{1}{\sigma^2}.$$

Theorem 1. *Suppose we have a normal distribution with unknown mean μ , known precision $r > 0$, and we represent our prior belief concerning the mean with a random variable M that has density function*

$$\text{NormalDen}(\mu; a, 1/v),$$

where $v > 0$. Then, given a random sample of size n , and a value \bar{x} of the mean of the sample, the posterior density function of M is

$$\text{NormalDen}(\mu; a^*, 1/v^*),$$

where

$$a^* = \frac{va + nr\bar{x}}{v + nr} \quad \text{and} \quad v^* = v + nr. \tag{2.3}$$

Proof. Let $d = \{x_1, x_2, \dots, x_n\}$ be the set of values of the random sample. It is not hard to show that

$$\rho(d|\mu) \simeq \exp \left[-\frac{r}{2} \sum_{i=1}^n (x_i - \mu)^2 \right], \tag{2.4}$$

where $\exp(y)$ denotes e^y and \simeq means ‘proportionate to’, and that

$$\sum_{i=1}^n (x_i - \mu)^2 = n(\mu - \bar{x})^2 + \sum_{i=1}^n (x_i - \bar{x})^2. \tag{2.5}$$

Since the far right term in Equality 2.5 does not contain μ , we may rewrite Relation 2.4 as follows:

$$\rho(d|\mu) \simeq \exp \left[-\frac{nr}{2} (\mu - \bar{x})^2 \right]. \tag{2.6}$$

The prior density function of M satisfies the following:

$$\rho(\mu) \simeq \exp \left[-\frac{v}{2} (\mu - a)^2 \right]. \tag{2.7}$$

We have

$$\begin{aligned} \rho(\mu|d) &\simeq \rho(d|\mu)\rho(\mu) \\ &\simeq \exp \left[-\frac{v}{2} (\mu - a)^2 \right] \exp \left[-\frac{nr}{2} (\mu - \bar{x})^2 \right]. \end{aligned} \tag{2.8}$$

The first proportionality in Relation 2.8 is due to Bayes' Theorem, and the second is due to Relations 2.6 and 2.7. It is not hard to show

$$v(\mu - a)^2 + nr(\mu - \bar{x})^2 = (v + nr)(\mu - a^*)^2 + \frac{vnr(\bar{x} - a)^2}{v + nr}. \quad (2.9)$$

Since the final term in Equality 2.9 does not contain μ , it can also be included in the proportionality factor. So we can rewrite Relation 2.8 as

$$\begin{aligned} \rho(\mu|d) &\propto \exp\left[-\frac{v + nr}{2}(\mu - a^*)^2\right] \\ &\propto \exp\left[-\frac{(\mu - a^*)^2}{2(1/v^*)}\right] \\ &\propto \text{NormalDen}(\mu; a^*, 1/v^*). \end{aligned} \quad (2.10)$$

Since $\rho(\mu|d)$ and $\text{NormalDen}(\mu; a^*, 1/v^*)$ are both density functions, their integrals over the real line must both equal 1. Therefore, owing to Relation 2.10, they must be the same function.

Example 4. Suppose we assume the heights of American males are normally distributed with known variance 4, and we represent our prior belief concerning the average height of American men with the density function

$$\text{NormalDen}(\mu; 72, 3^2),$$

and in a sample of size $n = 100$ the value \bar{x} of is 70. Then $a = 72$, $v = 1/9$, and $r = 1/4$. We therefore have that

$$a^* = \frac{va + nr\bar{x}}{v + nr} = \frac{(1/9)72 + 100(1/4)70}{1/9 + 100(1/4)} = 70.009.$$

$$v^* = v + nr = (1/9) + 100(1/4) = 25.111.$$

Suppose instead that $n = 10$. Then

$$a^* = \frac{va + nr\bar{x}}{v + nr} = \frac{(1/9)72 + 10(1/4)70}{1/9 + 10(1/4)} = 70.09.$$

$$v^* = v + nr = (1/9) + 10(1/4) = 2.61.$$

A $100(1 - \alpha)\%$ **probability interval** for the unknown mean is an interval centered around the expected value of the unknown mean that contains $100(1 - \alpha)\%$ of the mass in the posterior probability distribution of the mean. That is, the probability that μ is in this interval is .95. It is not hard to see that this interval is given by

$$\left(a^* - z_{\alpha/2} \frac{1}{\sqrt{v^*}}, a^* + z_{\alpha/2} \frac{1}{\sqrt{v^*}}\right).$$

Notice that the precision v in our prior distribution of μ must be positive. That is, the variance cannot be infinite. To represent prior ignorance as to the mean we take the variance to be infinite and therefore the precision to be 0. This would constitute an improper prior distribution. Neapolitan [2004] shows how to compute a posterior distribution using this improper prior. We can obtain the same result by simply taking the limit of the expressions in Equality 2.3 as v goes to 0. To that end, we have

$$\lim_{v \rightarrow 0} a^* = \lim_{v \rightarrow 0} \frac{va + nr\bar{x}}{v + nr} = \bar{x} \quad \text{and} \quad \lim_{v \rightarrow 0} v^* = \lim_{v \rightarrow 0} (v + nr) = nr. \quad (2.11)$$

In this case, our $100(1 - \alpha)\%$ probability interval is equal to

$$\left(\bar{x} - z_{\alpha/2} \frac{1}{\sqrt{nr}}, \bar{x} + z_{\alpha/2} \frac{1}{\sqrt{nr}} \right) = \left(\bar{x} - z_{\alpha/2} \frac{\sigma}{\sqrt{n}}, \bar{x} + z_{\alpha/2} \frac{\sigma}{\sqrt{n}} \right).$$

We see that this interval is identical to the confidence interval shown in Equality 2.2.

Example 5. Suppose we want to obtain an estimate of how many pounds of coffee a coffee manufacturer currently puts, on the average, into its 3 pound coffee cans. Suppose further that, from years of previous data, we know that $\sigma = .3$, and in a sample of size $n = 2$ we obtain the result that $\bar{x} = 2.92$. Then for the frequentist a 95% confidence interval for the unknown average μ is given by

$$\begin{aligned} \left(\bar{x} - z_{.05/2} \frac{\sigma}{\sqrt{n}}, \bar{x} + z_{.05/2} \frac{\sigma}{\sqrt{n}} \right) &= \left(2.92 - 1.96 \frac{.3}{\sqrt{2}}, 2.92 + 1.96 \frac{.3}{\sqrt{2}} \right) \\ &= (2.504, 3.336). \end{aligned} \quad (2.12)$$

This is also a 95% probability interval for the Bayesian if we assume prior ignorance.

I stress that although the confidence interval and the probability interval assuming prior ignorance are the same interval, the frequentist and the Bayesian attach different meaning to them. If we take an empirical Bayesian stance and acknowledge the existence of relative frequencies in nature, and if we assume the means μ of normal distributions are distributed uniformly in nature, then a 95% posterior probability distribution can be described as follows. Let a trial consist of selecting a population from the set of all normally distributed populations in nature, taking a sample of size n from this population, and obtaining a *particular* value of \bar{x} . In 95% of these trials, the population selected will have a mean μ that is in the *particular* interval determined by \bar{x} and n . This is a probabilistic statement about μ . On the other hand, a 95% confidence interval can be described like this. Suppose we have a particular population with unknown mean μ . Let a trial consist of taking a sample of size n from this population. In 95% of these trials, the confidence interval obtained will contain the mean μ of this population. This is a probabilistic statement about the intervals obtained in repeated trials. Furthermore, if a trial consists of selecting a population from the set of

all normally distributed populations in nature, and then taking a sample size n from the population selected, the confidence interval obtained will still contain the population mean 95% of the time (This is Neyman's explanation). The confidence interval approach (and its related fiduciary probability approach) does not say that the mean of the selected population is in the particular resultant interval 95% of the time we have this particular resultant value of \bar{x} . Rather it says the interval obtained will contain the population mean 95% of the time. Since the approach does not entail any assumption about the prior distribution of means in nature, it cannot lead to a posterior distribution of means, and so it cannot yield a probability statement that the mean is in a particular interval.

You might say "What if means are not uniformly distributed in nature? Then the Bayesian will get the 'wrong' answer." My statement about the uniform distribution of means was just a frequentist allegory. That is, it is the frequentist who models reality by claiming that there is an objective probability distribution out there, and we can only make probabilistic statements that are true relative to this distribution. The Bayesian feels probability is about coherent belief and updating that belief in the light of new evidence. So all that matters to the Bayesian is that for this particular distribution there is no reason to believe one mean is more likely than another. Whose interpretation of probability is more cogent - the frequentist's or the Bayesian's? Let's answer that with another question: Is there a relative frequency, accurate to an arbitrary number of digits, with which the coin in my pocket has been somehow predetermined to land heads if I toss it indefinitely? Or rather do we say the probability of heads is .5 because we have no reason to prefer heads to tails, and would we change that belief if sufficient tosses indicate otherwise?

Actually, it does not seem reasonable to assume prior ignorance in Example 5 since we know the average amount of coffee cannot be negative. Rather than assuming ignorance, we should at least assume knowledge upon which everyone would agree. So minimally we should assume $\mu > 0$. If we did this we would obtain a different probability interval. Instead of showing an example that makes this assumption (which would result in more complex mathematics to compute the posterior probability), we show a simpler case.

Example 6. Let's assume we can get everyone to agree the prior expected value of the mean is 3 and it is very unlikely to be less than 2 or greater than 4. In this case a standard technique is to use $1/6$ of the difference between the high and low possible values of the mean as our prior standard deviation. So in this case our prior standard deviation is $(1/6)(4 - 2) = 1/3$, which means our prior precision is $v = 1/(1/3)^2 = 9$. So our prior density function for the mean is

$$\text{NormalDen}(\mu; a, 1/v),$$

where $a = 3$ and $v = 9$. If we obtain the results in the previous example, we have then that

$$a^* = \frac{va + nr\bar{x}}{v + nr} = \frac{(9)3 + 2\left(\frac{1}{.3}\right)^2 2.92}{9 + 2\left(\frac{1}{.3}\right)^2} = 2.943.$$

$$v^* = v + nr = 9 + 2 \left(\frac{1}{.3} \right)^2 = 31.222.$$

So our posterior 95% probability interval is given by

$$\begin{aligned} & \left(a^* - z_{.05/2} \frac{1}{\sqrt{v^*}}, a^* + z_{.05/2} \frac{1}{\sqrt{v^*}} \right) \\ &= \left(2.943 - 1.96 \frac{1}{\sqrt{31.222}}, 2.943 + 1.96 \frac{1}{\sqrt{31.222}} \right) \\ &= (2.592, 3.294). \end{aligned}$$

Note that the 95% probability interval obtained in the preceding example is substantially different from the confidence interval obtained in Example 5. I purposely used a small sample size to show the difference can be substantial. As mentioned before, a given individual may not agree with the prior distribution we used for the mean in this example, but no individual would have the prior belief that the mean could be negative. Which means no one would obtain the posterior probability interval in Equality 2.12.

Even in the case of an unknown variance a $100(1 - \alpha)\%$ **confidence interval** for the unknown mean μ of a normal distribution is identical to a $100(1 - \alpha)\%$ **probability interval** for μ under the assumption of prior ignorance. Neapolitan [2004] shows this result. However, in general the two are usually close but not identical.

2.2.3 Confidence Intervals or Probability Intervals?

We presented two methods for obtaining our belief in the location of the mean of a normal distribution from a random sample, namely the confidence interval and the probability interval. Which method is more compelling? Next we review arguments in favor of each.

Arguments for the Confidence Interval

As noted previously, R.A. Fisher was originally a Bayesian. However, by 1921 he said that the Bayesian approach “depends upon an arbitrary assumption, so the whole method has been widely discredited,” and in 1922 he denounced “inverse probability, which like an impenetrable jungle arrests progress towards precision of statistical concepts.” It was this displeasure with Bayesian methods that led statisticians to embrace the notion of ‘complete objectivity’ and the confidence interval interpretation. They wanted a technique that did not require an individual’s subjective belief; their aim was to assure that everyone must agree on statistical results. But why couldn’t the ‘objective’ statistician be content with a uniform prior distribution for the unknown parameter, since it represents prior ignorance? Fisher also noted in 1922 that the uniformity of a prior distribution is not invariant under a parametric transformation. For example, let p be a random variable representing the unknown relative frequency (probability) with which

a thumbtack lands heads. We can assume prior ignorance as to the value of p using the uniform density function. That is, $\rho(p) = 1$ for all $p \in [0, 1]$. However, if we let $q = p^2$, then q is not uniformly distributed in the interval $[0, 1]$. It is more probable that q takes small values. For example, the probability that q is in $[0, .25]$ is equal to the probability that p is in $[0, .5]$, which is .5. So our claim as to ignorance about p is a claim as to knowledge about q .

We see that the argument for using the confidence interval methodology is actually that the Bayesian approach is not reasonable. The first objection about ‘arbitrary assumptions,’ however, is not very compelling because the Bayesian method does not require an individual’s prior belief; rather it just allows them. We can always use a uniform prior. But, as noted above, Fisher also objected to this because a uniform prior on one random variable can result in a nonuniform prior on another. So he concludes that a uniform prior does not really represent prior ignorance. This is, however, exactly the point. That is, uniformity is relative to a particular choice of scale, and therefore represents knowledge, not ignorance. As Sandy Zabell (private correspondence) puts it, “Before you impose a uniform prior, you have to choose the units or scale. Once you do that, you can write down the uniform prior. But if you had chosen another scale, you would have gotten a different prior, uniform with respect to the second scale. So the question becomes: what is the justification for the particular scale chosen? If you claim not to know anything at all, then obviously you can’t justify the choice of any particular scale.” So when we say that p is uniformly distributed in $[0, 1]$ our knowledge is that we are completely ignorant as to the value of p . This entails that we are not completely ignorant as to the value of p^2 .

Arguments for the Probability Interval

The standard argument for the Bayesian approach is that it allows us to combine information and evidence from different sources. For example, when developing an expert system we may want to combine expert information with information obtained from data in order to estimate a probability value. When using a medical expert system we may want to compute the probability of lung cancer given the evidence that the patient smokes and has a positive chest X-ray. When trying to determine how likely it is that a husband killed his wife, we may want to combine the evidence that the husband’s gun was used to shoot the wife and that the husband’s blood was found at the scene of the crime. We will not belabor this matter further as it is discussed in detail in Bayesian statistics books such as [Berry, 1996].

Rather we present the more subtle argument that the use of the confidence interval can lead one to a ‘Dutch book,’ which is a bet one cannot win. The Dutch book argument in probability is based on the assumption that if one claims the probability of an event is p , then that person would deem it fair to give $\$p$ for the promise to receive $\$1$ if the event occurs (or is true). For example, if Clyde says that the probability that the Chicago Cubs will beat the Detroit Tigers in an upcoming American baseball game is .6, then Clyde should consider it fair to give $\$.6$ for the promise to receive $\$1$ if the Cubs win. Since he deems it a

fair bet, Clyde should be willing to take either side of the bet. It is possible to show (See [Finetti, 1964] or [Neapolitan, 1990].) that, if an individual assigns numbers (representing the individual's beliefs) between 0 and 1 to uncertain events, then unless the individual agrees to assign and combine those numbers using the rules of probability, the individual can be forced into a Dutch book. Here is a simple example. Suppose Clyde maintains that the probability of the Cubs winning is .6 and the probability of the Tigers winning is .6. Then we can require that he give \$.6 for the promise to receive \$1 if the Cubs win, and also require that he give \$.6 for the promise to receive \$1 if Detroit wins. Since precisely one team will win (There are no ties in American baseball), Clyde will definitely give \$1.20 and receive \$1. Poor Clyde is sure to lose \$.2. So he should have set $P(\text{Cubs_win}) + P(\text{Tigers_win}) = 1$ as the rules of probability require.

Consider now the confidence assigned to a confidence interval. Regardless of the physical interpretation we give to it, is it not a number between 0 and 1 that represents its user's belief as to the location of the unknown parameter? If so, it satisfies the requirements to be called a subjective probability. Consider again Example 5. If Clyde says he is 95% confident the mean is in (2.504, 3.336), wouldn't Clyde say it is fair to give up \$.95 for the promise to receive \$1 if the mean is in that interval? If Clyde acknowledges he would do this, then the confidence measure is indeed a subjective probability for Clyde.

Suppose now that Clyde accepts Bayesian methodology. As noted following Example 5, everyone, including Clyde, knows that the mean cannot be negative. So, whatever Clyde's prior belief is concerning the value of the mean, it is different from the improper uniform prior over the real line. Therefore, Clyde's posterior 95% probability interval, obtained using Bayesian updating, will not be (2.504, 3.336) (Recall that this is also the 95% probability interval obtained when we assume a uniform prior over the real line.). For the sake of illustration, let's say Clyde's posterior 95% probability interval is the interval (2.592, 3.294) obtained in Example 6. Since this interval was obtained using the rules of probability, Clyde must use it to represent his posterior belief to avoid a Dutch book. However, from his confidence interval analysis, he also has the interval (2.504, 3.336) representing his posterior belief. This means we can get him to bet on both intervals. We have him give \$.95 for the promise to receive \$1 if the mean is in (2.592, 3.294), and we have him receive \$.95 for the promise to give \$1 if the mean is in (2.504, 3.336) (Recall that he would take either side of the bet). If it turns out the mean is in (2.592, 3.294) or outside (2.504, 3.336), he breaks even. However, if it is in (2.504, 2.592) or in (3.294, 2.336), he loses \$.95 + \$.05 = \$1.00. Clyde can't win money, but can lose some. Football gamblers call this situation a 'middle', and they actively look for two different spreads on the same game to be on the advantageous end of the middle. They would like to find Clyde.

So if Clyde (or any Bayesian) accepts the argument concerning Dutch books, he cannot, in general, use a confidence interval to represent his belief. He can of course use it for its literal meaning as the probability of an interval that contains

the mean. There is no disputing this. But its literal meaning has limited value as he can't bet on it (literally or figuratively).

Where does all this leave the frequentist? The frequentist can escape from this conundrum by refusing to play the Bayesian's game. Suppose Ann is a frequentist. Ann could argue that her confidence in a confidence integral does not represent what she considers a fair bet. So we can't force her to bet according to it. Or, Ann could maintain that, even though she knows the mean cannot be negative, she cannot assign a numerical prior probability distribution to the value of the mean. So she has no posterior probability interval for the mean, and therefore there is no bet we could force concerning such an interval. Regardless, it should be somewhat disconcerting to one who would use a confidence interval that, in some cases, the confidence interval obtained is identical to the probability interval obtained assuming a mean could be negative when we know it cannot.

2.3 Hypothesis Testing

After discussing frequentist hypothesis testing, we show the Bayesian method.

2.3.1 Frequentist Hypothesis Testing

We illustrate the method with an example of one of the most common applications of hypothesis testing.

Example 7. Suppose a coffee manufacturer claims that on the average it puts 3 pounds of coffee in its 3 pound coffee cans. However, consumers have complained that they have not been receiving that much. So we decide to investigate the claim by obtaining a random sample of size $n = 40$ coffee cans. Suppose further that the result of that sample is that $\bar{x} = 2.92$, and that, from years of previous data, we know that $\sigma = .3$. We are interested in whether the true mean μ is ≥ 3 because, if so, the manufacturer is meeting its claim or doing better. We call this event the null hypothesis and denote it H_0 . We call the event that μ is < 3 the alternate hypothesis H_A . If this event is true the manufacturer is cheating on the quantity of coffee supplied. Our goal is to see if we can reject the null hypothesis, and thereby suspect the coffee manufacturer of cheating. In summary, we have

$$\begin{aligned} H_0 : \mu &\geq 3 \\ H_A : \mu &< 3. \end{aligned}$$

Now owing to Lemma 1, if $\mu = 3$ then \bar{X} has density function

$$\text{NormalDen}(\bar{x}; 3, (.3)^2/40).$$

This density function appears in Figure 2.2. Recall that our sample mean had $\bar{x} = 2.92$. We have that

$$\int_{-\infty}^{2.92} \text{NormalDen}(\bar{x}; 3, (.3)^2/40) d\bar{x} = .046.$$

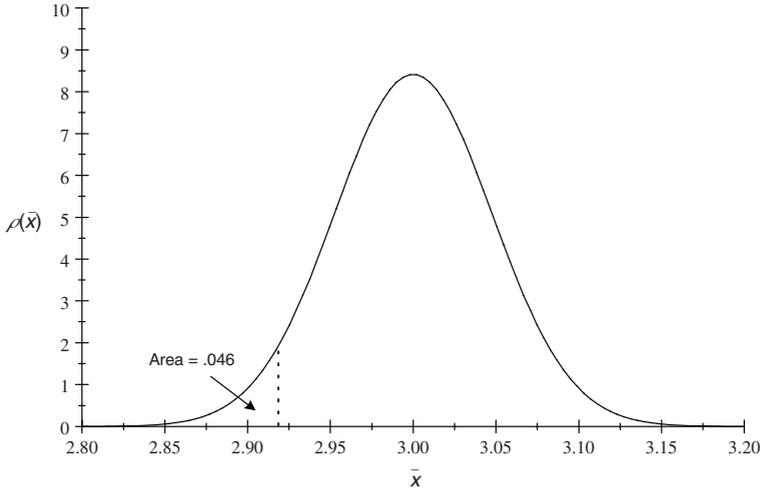


Fig. 2.2. The p -value at 2.92 is .046

This means that .046 of the area under the density function falls to the left of 2.92. This is illustrated in Figure 2.2. Frequentists call .046 the p -value for the sample result, and they say we can reject H_0 at the .046 significance level. Their explanation is that if we reject H_0 whenever the p -value lies at .046 or to the left of it, if H_0 is true, we will reject it at most .046 fraction of the time. We say ‘at most’ because H_0 is true if $\mu \geq 3$. We computed the p -value relative to 3. The p -values relative to numbers greater than 3 will be smaller.

Often frequentists set a significance level α before doing the study, and then reject H_0 at the α significance level if the result falls to the left of the point \bar{x}_α determined by α . For example, if $\alpha = .05$, $\bar{x}_\alpha = 2.988$. In the previous example our result is $\bar{x} = 2.92$, which is to the left of 2.988. So we reject H_0 at the .05 significance level. Regarding this practice, Brownlee [1965] says “The first criterion is that only in a small fraction α of the time shall we commit a *Type I error* or an *error of the first kind*, which is to reject the null hypothesis when in fact it is true. Clearly we wish to commit such an error only rarely: common choices for α are 0.05 and 0.01.” It has become credo among many practitioners of frequentist statistics to deem a result significant whenever the p -value is $\leq .05$.

We postpone further discussion of frequentist hypothesis testing until after we show the Bayesian method.

2.3.2 Bayesian Hypothesis Testing

Recall that the Bayesian represents uncertainty concerning the unknown mean with a random variable M that has density function

$$\text{NormalDen}(\mu; a, 1/v).$$

Then, if $r = 1/\sigma^2$ where σ^2 is the known variance, given a random sample of size n and a value \bar{x} of the mean of the sample, the posterior density function of M is

$$\rho(\mu|\bar{x}) = \text{NormalDen}(\mu; a^*, 1/v^*),$$

where

$$a^* = \frac{va + nr\bar{x}}{v + nr} \quad \text{and} \quad v^* = v + nr.$$

Now suppose we have the following hypothesis:

$$\begin{aligned} H_0 &: \mu \geq \xi \\ H_A &: \mu < \xi. \end{aligned}$$

Let d denote the information on which we are conditioning. That is, $d = \{n, \bar{x}\}$. Then, due to the law of total probability, the posterior probability of the null hypothesis H_0 given d is given by

$$\begin{aligned} P(H_0|d) &= \int_{-\infty}^{\infty} P(H_0|d, \mu) \times \rho(\mu|d) d\mu \\ &= \int_{-\infty}^{\xi} 0 \times \rho(\mu|d) d\mu + \int_{\xi}^{\infty} 1 \times \rho(\mu|d) d\mu \\ &= \int_{\xi}^{\infty} \text{NormalDen}(\mu; a^*, 1/v^*) d\mu \end{aligned}$$

Similarly,

$$P(H_A|d) = \int_{-\infty}^{\xi} \text{NormalDen}(\mu; a^*, 1/v^*) d\mu.$$

Example 8. Suppose we have the situation described in Example 7. Then $\sigma = .3$, $n = 40$, $\bar{x} = 2.92$, and

$$\begin{aligned} H_0 &: \mu \geq 3 \\ H_A &: \mu < 3. \end{aligned}$$

If we assume prior ignorance as to the value of μ , then owing to Equality 2.11,

$$\begin{aligned} \rho(\mu|d) &= \text{NormalDen}(\mu; \bar{x}, 1/nr) \\ &= \text{NormalDen}(\mu; \bar{x}, \sigma^2/n) \\ &= \text{NormalDen}(\mu; 2.92, (.3)^2/40). \end{aligned}$$

Therefore,

$$P(H_0|d) = \int_3^{\infty} \text{NormalDen}(\mu; 2.92, (.3)^2/40) d\mu = .046$$

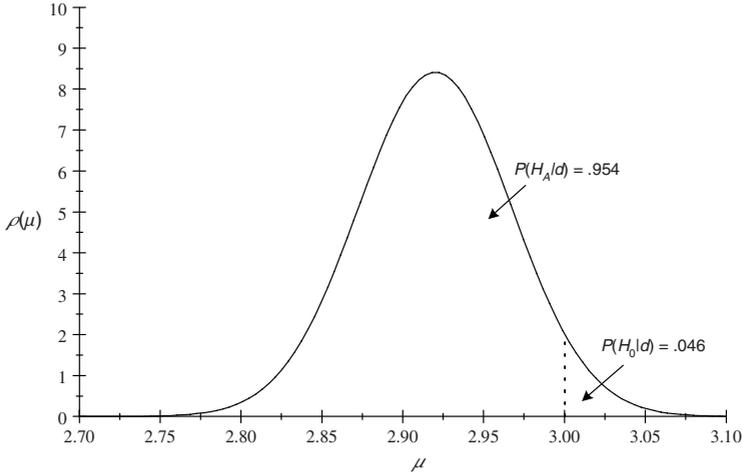


Fig. 2.3. The posterior probability of H_0 is the area to the right of the dotted line, while that of H_A is the area to the left

$$P(H_A|d) = \int_{-\infty}^3 \text{NormalDen}(\mu; 2.92, (.3)^2 / 40) d\mu = .954.$$

Notice in the previous example that $P(H_0|d)$ is equal to the p -value obtained in Example 7. It is not hard to see that this is a general result. That is, if we assume prior ignorance then the Bayesian's posterior probability of H_0 is equal to the frequentist's p -value. Furthermore, this result holds for the normal distribution even if we do not assume we know the variance. However, in general the two are usually close but not identical. The following example illustrates this.

Example 9. Suppose the new drug xhair is being introduced to prevent hair loss, but it seems that it may have the unpleasant side effect of acne. The drug company wishes to investigate this possibility. Let

$$\begin{aligned} p_1 &= P(\text{acne}|xhair) \\ p_2 &= P(\text{acne}|\neg xhair), \end{aligned}$$

where $xhair$ denotes the event that one uses xhair, and $\neg xhair$ denotes the event that one does not use it. To investigate whether xhair may cause acne, the company formulates these hypotheses:

$$\begin{aligned} H_0 &: p_1 \leq p_2 \\ H_A &: p_1 > p_2. \end{aligned}$$

They wish to see if they need to reject H_0 and admit a problem with their drug. Suppose in a study consisting of 900 xhair users and 900 control subjects they obtain these results:

	<i>acne</i>	\neg <i>acne</i>
<i>xhair</i>	34	866
\neg <i>xhair</i>	19	881

Then Fisher’s exact test for 2×2 tables yields this result:

$$p\text{-value} = \frac{1}{\binom{900+900}{34+19}} \sum_{k=0}^{19} \binom{900}{k} \binom{900}{34+19-k} = .0250.$$

To assume prior ignorance as to the values of p_1 and p_2 , we assign the uniform prior density functions as follows:

$$\begin{aligned} \rho(p_1) &= 1 && \text{for } p_1 \in [0, 1] \\ \rho(p_2) &= 1 && \text{for } p_2 \in [0, 1]. \end{aligned}$$

Given the data d in the above table and these priors, it is possible to show (See [Neapolitan, 2004]) that the posterior density functions are given by

$$\begin{aligned} \rho(p_1|d) &= \text{BetaDen}(p_1; 1 + 34, 1 + 866) \\ \rho(p_2|d) &= \text{BetaDen}(p_2; 1 + 19, 1 + 881), \end{aligned}$$

where BetaDen is the beta density function. We then have that

$$\begin{aligned} P(H_0|d) &= P(p_1 \leq p_2|d) \\ &= \int_0^1 \int_0^{p_2} \text{BetaDen}(p_1, 35, 867) \text{BetaDen}(p_2, 20, 882) dp_1 dp_2 \\ &= .0186. \end{aligned}$$

We see that the p -value and posterior probability are close but not equal.

2.3.3 Comparing the Two Approaches

Next we compare the two approaches to hypothesis testing by citing difficulties with the frequentist approach and showing the Bayesian approach does not have these difficulties.

Frequentist Results are Difficult to Understand

Recall that Brownlee [1965] explained significance level as follows: “The first criterion is that only in a small fraction α of the time shall we commit a *Type I error* or an *error of the first kind*, which is to reject the null hypothesis when in fact it is true. Clearly we wish to commit such an error only rarely: common choices for α are 0.05 and 0.01.” A Question immediately come to bear. Why do we choose the far left region of the curve as our rejection interval? In Example 7 if we choose a region just to the right of 3 with area .05, Brownlee’s criterion

will still be satisfied. However, this makes no sense at all since, if the result falls in this region, it seems to support H_0 rather than discredit it. So further explanation is required. Now we must consider errors of the second kind. Again quoting [Brownlee, 1965] “To accept H_0 when it is false is an error, known as an *error of the second kind* or *Type II error*. The probability of the complementary event, namely of correctly rejecting the null hypothesis, is known as the *power* of the test, say π . The power of the test is thus equal to the probability of x falling in the region of rejection when it is has the alternative distribution. ... We would like to have a large power.” In Example 7 the alternative hypothesis is composite consisting of all values < 3 . So the situation is somewhat more complex than than just choosing a region that attempts to minimize Type I errors while maximizing power for a particular value. Further development is necessary to justify the interval choice, and a theoretical frequentist text such as [Brownlee, 1965] provides that justification, although many of the practitioner frequentist texts do not. Briefly, if we let $\rho(x : \mu)$ denote the density function if the value of the unknown mean is μ , then the Neyman-Pearson Lemma [Neyman and Pearson, 1933] state that the rejection region should include those x for which $\rho(x : \mu)$ is as large as possible for every μ in H_A compared to $\rho(x : \mu')$ where μ' is the endmost value in H_0 . The lemma states that such a region will result in maximum power. The region is then found by maximizing a likelihood ratio. In the case of Example 7, the region turns out to be the far left region shown in Figure 2.2 (as our intuition tells us it should be). For every x in that region and for every $\mu \in H_A$, $\rho(x : \mu) > \rho(x : 3)$.

If the preceding explanation seems confusing, I suspect it is because the underlying methodology is confusing. It is no wonder that many practitioners have simply chosen to consider a result significant if the p -value is below the enigmatic .05 level.

The Bayesian’s life is so much easier. In Example 8, we conditioned on the fact that $n = 40$ and $\bar{x} = 2.92$, and learned that the posterior probability of H_0 was .046 and that of H_A was .954. This result tells the whole story. There is no need to talk about rejection regions, errors of the first, errors of the second kind, etc. When I first encountered hypothesis testing as a college student, I was somewhat flummoxed by the concern for a region when in fact the result was a point value. The Bayesian only conditions on this point value.

Frequentist Results Do Not Give a Confidence Measure

The fact that frequentist methods are harder to understand should not in itself preclude them. A more serious problem is that the p -value is not a measure of our confidence in the falsity of H_0 . That is, we may know that the rejection region is unlikely given any value in H_0 and it is more likely given any value in H_A , but in general we don’t have a measure of how much more likely it is. That is, a small p -value does not rule out that the result could also be unlikely given H_A . The frequentist is not able to give us a numeric measure of how confident we should be that H_0 is false. The Bayesian does not have this problem. In Example 8,

the posterior probability of H_0 was .046 and that of H_A was .954. We know exactly how likely H_0 is.

The situation is a little more complex when H_0 consists of a point value μ_0 . For example, H_0 may be that the mean of a normal distribution is 3, while H_A is that the mean is not 3. Then the probability of H_0 is 0 since the probability of every point value is 0. In this case, we calculate a $100(1 - \alpha)\%$ posterior probability interval for the unknown mean, and see if μ_0 is in the interval. If μ_0 is not in the interval we reject H_0 at the .05 probability level. This is the standard Bayesian method for handling this situation. However, it seems it may be a carryover of frequentist methodology. That is, in most (if not all) situations it seems that no reasonable hypothesis could be that a parameter has a precise value. In such situations the hypothesis could just be that the mean is close to 3. Then we can investigate the posterior probability of intervals centered at 3. How small these intervals need be and how large the probabilities need be (to not reject H_0) would depend on the application.

2.3.4 The Multiple Hypothesis Paradox

Another difficulty with frequentist hypothesis testing is that it can lead to a problem which I call the multiple hypothesis paradox. I first describe the paradox with a classic example. Then I show a practical situation in which the problem arises, and a frequentist effort to solve the problem that leads to questionable results.

The Case of the Positive Blood Test



Suppose a woman is murdered and her husband is the primary suspect. Suppose further that a very accurate blood test can be run to see if the husband's

blood matches that found at the scene of the crime. If the test comes back positive, the jury will become almost certain the husband is the murderer because, if the blood at the crime scene is not the husband's blood, a very rare event would have occurred. On the other hand, if we select 1,000,000 people in the city and test their blood, it is probable that at least one's blood will match that found at the crime scene (even if the blood is not from any of them). So if one of their blood matches that found at the crime scene, we will not become confident that the person is the murderer. The husband, realizing this, can insist that they test him along with 1,000,000 other people so that, if his test is positive, the jury cannot be confident he is the murderer. I call this the **multiple hypothesis paradox** for frequentist statistics.

Let's formalize the previous discussion with two experiments:

Experiment 1

Assume the accuracy of the blood test is as follows:

$$\begin{aligned} P(\text{match}|\text{blood}) &= 1 \\ P(\text{match}|\neg\text{blood}) &= .00001, \end{aligned}$$

where *blood* is the event that the blood at the scene of the crime is blood from the individual being tested, $\neg\text{blood}$ is the event that the blood at the scene of the crime is not from the given individual, and *match* is the event that the blood at the scene of the crime matches blood taken from the given individual. Let our hypotheses be as follows:

$$\begin{aligned} H_0 &: \text{The blood at the crime scene is not from the given individual} \\ H_A &: \text{The blood at the crime scene is from the given individual.} \end{aligned}$$

Then

$$\begin{aligned} P(\text{match}|H_0) &= .00001 \\ P(\text{match}|H_A) &= 1. \end{aligned}$$

The region of rejection is *match*, and the region of acceptance is $\neg\text{match}$. So the significance level of the test is .00001, while the power is 1. If the husband's test comes back positive (event *match* occurs), we reject H_0 at a very significant level and believe the husband is the murderer.

Experiment 2

We test the blood of 1,000,000 people (including the husband) to see if any of their blood matches that found at the crime scene. Let our hypotheses be as follows:

$$\begin{aligned} H_0 &: \text{The blood at the crime scene is not from any of them} \\ H_A &: \text{The blood at the crime scene is from one of them.} \end{aligned}$$

Then

$$P(\text{any_match}|H_0) = 1 - (1 - .00001)^{1000000} = .99995.$$

$$P(\text{any_match}|H_A) = 1,$$

where *any_match* is the event that at least one person's blood matches that at the crime scene. The test has little significance. So if the husband's blood matches we cannot reject H_0 with any significance, which means we cannot believe he is the murderer.

How can the frequentist resolve this matter? Perhaps the frequentist could make claims about the experimental setup. That is, our only purpose was to investigate the husband. So we should run an experiment that involves only him. We should not involve individuals we do not suspect. But isn't this implicitly updating using the prior belief that the husband is the murderer, which frequentists emphatically claim they do not do. The Bayesian makes this prior belief explicit.

The Bayesian Method

The Bayesian can assign a prior belief to the hypothesis that the husband is the murderer based on evidence so far analyzed. Let's say we assign $P(\text{husband_murderer}) = .1$. For simplicity let's also assume someone is the murderer if and only if it is their blood that is at the crime scene. Then for the husband $P(\text{blood}) = .1$. We then have

$$P(\text{blood}|\text{match}) = \frac{P(\text{match}|\text{blood})P(\text{blood})}{P(\text{match}|\text{blood})P(\text{blood}) + P(\text{match}|\neg\text{blood})P(\neg\text{blood})}$$

$$= \frac{1(.1)}{1(.1) + .00001(.9)} = .9999.$$

If the blood matches the husband's blood, we will become almost certain it is his blood at the crime scene and therefore he is the murderer. You might ask how we could obtain this prior probability of .1. Alternatively, we could determine the posterior probability needed to believe the husband is guilty "beyond a reasonable doubt." Let's say that probability is .999. Then we solve

$$\frac{1(p)}{1(p) + .00001(1-p)} = .999$$

for p . The solution is $p = .0099$. So if the prior evidence entails that the chances are at least 1 in 100 that he is the murderer, a match should be sufficient additional evidence to convict him.

Now what about an arbitrary person whose blood matches the crime scene blood when that person is one of 1,000,000 people we test. Again it all depends on the prior probability that this person is the murderer. If, for example, we were checking all 1,000,000 in the city and we were certain that one of them is the murderer, then this prior probability would be .000001, and we would have

$$\begin{aligned}
 P(\text{blood}|\text{match}) &= \frac{P(\text{match}|\text{blood})P(\text{blood})}{P(\text{match}|\text{blood})P(\text{blood}) + P(\text{match}|\neg\text{blood})P(\neg\text{blood})} \\
 &= \frac{1(.000001)}{1(.000001) + .00001(.999999)} = .091.
 \end{aligned}$$

Note that this is the probability of the person being the murderer in the absence of information about the test results of others. For example, in the scenario we described, if all other results were negative, we would be certain this person is the murderer.

Spatial Cluster Detection

Spatial cluster detection consists of detecting clusters of some event in subregions of a region being analyzed. Practitioners in spatial cluster detection are troubled by the multiple hypothesis paradox. Next I discuss their problem, and a frequentist effort to solve it. On page 16, Neill [2006] describes the problem in the context of disease outbreak detection as follows:

Let us consider the example of disease surveillance, assuming that we are given the count (number of disease cases) c_i , as well as the expected count (mean μ_i and standard deviation σ_i), for each zip code s_i . How can we tell whether any zip code has a number of cases that is significantly higher than expected? One simple possibility would be to perform a separate statistical test for each zip code, and report all zip codes that are significant at some level α ... A second, and somewhat more subtle, problem is that of *multiple hypothesis testing*. We typically perform statistical tests to determine if an area is significant at some level α , such as $\alpha = .05$, which means that if there is no abnormality in that area (i.e. the “null hypothesis” of no clusters is true) our probability of a false alarm is at most α . A lower value of α results in less false alarms, but also reduces our chance of detecting a true cluster. Now let us imagine that we are searching for disease clusters in a large area containing 1000 zip codes, and that there happen to be no outbreaks today, so any areas we detect are false alarms. If we perform a separate significance test for each zip code, we expect to trigger an alarm with probability $\alpha = 0.05$. But because we are doing 1000 separate tests, our expected number of false alarms is $1000 \times 0.05 = 50$. Moreover, if these 1000 tests were independent, we would expect to get at least one false alarm with probability $1 - (1 - .05)^{1000} \approx 1$... The main point here, though, is that we are almost certain to get false alarms every day, and the number of such false alarms is proportional to the number of tests performed.

Later on page 20, Neill [2006] offers the following frequentist solution to this problem:

Once we have found the regions with the highest scores $F(S)$, we must still determine which of these “potential clusters” are likely to be

“true clusters” resulting from a disease outbreak, and which are likely to be due to be due to chance... Because of the multiple hypothesis testing problem discussed above, we cannot simply compute separately whether each region score $F(S)$ is significant, because we would obtain a large number of false positives, proportional to the number of regions searched. Instead for each region S , we ask the question, “If this data were generated under the null hypothesis H_0 , how likely would we be to find any regions with scores higher than $F(S)$?” To answer this question, we use the method known as *randomization testing*: we randomly generate a large number of “replicas” under the null hypothesis, and compute the maximum score $F^* = \max_S F(S)$ of each replica...

Once we have obtained F^* for each replica, we can compute the statistical significance of any region S by comparing $F(S)$ to these replica values of F^* ... The p -value of region S can be computed as $\frac{R_{beat}+1}{R+1}$, where R is the total number of replicas created, and R_{beat} is the number of replicas with F^* greater than $F(S)$. If the p -value is less than our significance level α , we conclude that the region is not significant (likely to be due to chance).

Suppose epidemiologists decide to use this strategy to monitor for disease outbreaks in Westchester, IL., a suburb of Chicago, and it works quite well. That is, outbreaks are detected early enough that preventative measures can be taken, while false positives are kept at an acceptable level. It works so well that they decide to extend the system to monitor the entire Chicagoland area. Since we now have a much larger region and we are committed to keeping the total false positive rate (over the entire monitored region) under α , we will report outbreaks in Westchester far less often. If we extend the monitored region to all of Illinois, or even all of the United States, we will almost never report an outbreak in Westchester. By the mere fact that we wanted to monitor a larger region, we have messed up a system that was working in Westchester! The point is that the threshold at which we report an outbreak in a given region should be based on how early we feel outbreaks must be detected in the region versus the number of false positives we would tolerate for the region. However, the randomization testing technique discussed above entails that outbreak detection in a given region depends on how many other regions we decide to monitor along with the region, all due to our need to satisfy some arbitrary significance level! Neill et al. [2005] developed a Bayesian spatial scan statistic which does not do randomization testing.

A Final Example

These considerations apply to many domains. For example, several individuals have argued the following to me: “There is no reason to believe any one mutual fund manager is better than another one. Sure, there are quite a few with good records over the past 20 years; even ones who did not lose during the years

2000-2003. However, with all the fund managers out there by chance there should be some with good records.” I personally have a prior belief that some managers could be better than others. This seems true in sports, science, and other domains. So I have no reason to believe it is not true in finance. Therefore, I have the posterior belief that those with good track records are better and are likely to do better in the future. Curiously, there are some who believe those with good records are likely to do worse in the future because they are due for a fall. But this is a another topic!

Acknowledgement. I would like to thank Sandy Zabell, Piotr Gmytrasiewicz, and Scott Morris for reading this paper and providing useful criticisms and suggestions. As a personal historical note, I was a frequentist up until the late 1980’s as that was the only probability and statistics I knew. However, I was never comfortable with the methods. At that time I fortuitously met Sandy Zabell at the University of Chicago, and we subsequently spent many hours at a local campus diner drinking coffee and discussing the foundations of probability and statistics. I was eventually ‘converted’ to being a Bayesian. I say ‘converted’ because, as Sandy used to say, one cannot think clearly about probability until one purges oneself of frequentist dogma and realizes probability is about belief.

References

1. Anderson, D., Sweeny, D., Williams, T.: Statistics for Business and Economics, South-Western, Mason, Ohio (2005)
2. Berry, D.: Statistics: A Bayesian Perspective, Duxbury, Belmont, CA (1996)
3. Bolstad, W.: Introduction to Bayesian Statistics. Wiley, NY (2004)
4. Brownlee, K.: Statistical Theory and Methodology. Wiley, NY (1965)
5. Buntine, W.: Learning Classification Trees. Statistics and Computing 2 (1992)
6. de Finetti, B.: Foresight: Its Logical Laws, Its Subjective Source. In: Kyburg Jr., H.E., Smokler, H.E. (eds.) Studies in Subjective Probability. Wiley, NY (1964)
7. Fisher, R.A.: On the Probable Error of a Coefficient of Correlation Deduced from a Small Sample. *Metron* 1 (1921)
8. Fisher, R.A.: On the Mathematical Foundations of Theoretical Statistics. *Philos. Trans. Roy. Soc. London, Ser. A* 222 (1922)
9. Fisher, R.E.: Inverse Probability. *Proceedings of the Cambridge Philosophical Society* 26 (1930)
10. Neapolitan, R.E.: Probabilistic Reasoning in Expert Systems. Wiley, NY (1990)
11. Neapolitan, R.E.: Learning Bayesian Networks. Prentice Hall, Upper Saddle River (2004)
12. Neill, D.B.: Detection of Spatial and Spatio-Temporal Clusters, Ph.D. thesis, Department of Computer Science, Carnegie Mellon University, Technical Report CMU-CS-06-142 (2006)
13. Neill, D.B., Moore, A.W., Cooper, G.F.: A Bayesian Spatial Scan Statistic. *Advances in Neural Information Processing Systems (NIPS)* 18 (2005)
14. Neyman, J.: Outline of a Theory of Statistical Estimation Based on the Classical Theory of Probability. *Philos. Trans. Roy. Soc. London, Ser. A* 236 (1937)

15. Neyman, J., Pearson, E.S.: On the Problem of the Most Efficient Type of Statistical Hypotheses. *Philos. Trans. Roy. Soc. London, Ser. A* 231 (1933)
16. Pearl, J.: *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Mateo (1988)
17. Zabell, S.: R.A.Fisher and the Fiducial Argument. *Statistical Sciences* 7(3) (1992)

A Tutorial on Learning with Bayesian Networks*

David Heckerman

Microsoft Research
Advanced Technology Division
Microsoft Corporation
One Microsoft Way
Redmond WA, 98052
heckerma@microsoft.com

Abstract. A Bayesian network is a graphical model that encodes probabilistic relationships among variables of interest. When used in conjunction with statistical techniques, the graphical model has several advantages for data analysis. One, because the model encodes dependencies among all variables, it readily handles situations where some data entries are missing. Two, a Bayesian network can be used to learn causal relationships, and hence can be used to gain understanding about a problem domain and to predict the consequences of intervention. Three, because the model has both a causal and probabilistic semantics, it is an ideal representation for combining prior knowledge (which often comes in causal form) and data. Four, Bayesian statistical methods in conjunction with Bayesian networks offer an efficient and principled approach for avoiding the overfitting of data. In this paper, we discuss methods for constructing Bayesian networks from prior knowledge and summarize Bayesian statistical methods for using data to improve these models. With regard to the latter task, we describe methods for learning both the parameters and structure of a Bayesian network, including techniques for learning with incomplete data. In addition, we relate Bayesian-network methods for learning to techniques for supervised and unsupervised learning. We illustrate the graphical-modeling approach using a real-world case study.

3.1 Introduction

A Bayesian network is a graphical model for probabilistic relationships among a set of variables. Over the last decade, the Bayesian network has become a popular representation for encoding uncertain expert knowledge in expert systems (Heckerman et al., 1995a). More recently, researchers have developed methods for learning Bayesian networks from data. The techniques that have been developed are new and still evolving, but they have been shown to be remarkably effective for some data-analysis problems.

In this paper, we provide a tutorial on Bayesian networks and associated Bayesian techniques for extracting and encoding knowledge from data. There are numerous representations available for data analysis, including rule bases, decision trees, and artificial neural networks; and there are many techniques for

* Re-printed with kind permission of MIT Press and Kluwer books.

data analysis such as density estimation, classification, regression, and clustering. So what do Bayesian networks and Bayesian methods have to offer? There are at least four answers.

One, Bayesian networks can readily handle incomplete data sets. For example, consider a classification or regression problem where two of the explanatory or input variables are strongly anti-correlated. This correlation is not a problem for standard supervised learning techniques, provided all inputs are measured in every case. When one of the inputs is not observed, however, most models will produce an inaccurate prediction, because they do not encode the correlation between the input variables. Bayesian networks offer a natural way to encode such dependencies.

Two, Bayesian networks allow one to learn about causal relationships. Learning about causal relationships are important for at least two reasons. The process is useful when we are trying to gain understanding about a problem domain, for example, during exploratory data analysis. In addition, knowledge of causal relationships allows us to make predictions in the presence of interventions. For example, a marketing analyst may want to know whether or not it is worthwhile to increase exposure of a particular advertisement in order to increase the sales of a product. To answer this question, the analyst can determine whether or not the advertisement is a cause for increased sales, and to what degree. The use of Bayesian networks helps to answer such questions even when no experiment about the effects of increased exposure is available.

Three, Bayesian networks in conjunction with Bayesian statistical techniques facilitate the combination of domain knowledge and data. Anyone who has performed a real-world analysis knows the importance of prior or domain knowledge, especially when data is scarce or expensive. The fact that some commercial systems (i.e., expert systems) can be built from prior knowledge alone is a testament to the power of prior knowledge. Bayesian networks have a causal semantics that makes the encoding of causal prior knowledge particularly straightforward. In addition, Bayesian networks encode the strength of causal relationships with probabilities. Consequently, prior knowledge and data can be combined with well-studied techniques from Bayesian statistics.

Four, Bayesian methods in conjunction with Bayesian networks and other types of models offers an efficient and principled approach for avoiding the over fitting of data. As we shall see, there is no need to hold out some of the available data for testing. Using the Bayesian approach, models can be “smoothed” in such a way that all available data can be used for training.

This tutorial is organized as follows. In Section 3.2, we discuss the Bayesian interpretation of probability and review methods from Bayesian statistics for combining prior knowledge with data. In Section 3.3, we describe Bayesian networks and discuss how they can be constructed from prior knowledge alone. In Section 3.4, we discuss algorithms for probabilistic inference in a Bayesian network. In Sections 3.5 and 3.6, we show how to learn the probabilities in a fixed Bayesian-network structure, and describe techniques for handling incomplete data including Monte-Carlo methods and the Gaussian approximation. In Sections 3.7 through 3.12, we show how to learn both the probabilities and

structure of a Bayesian network. Topics discussed include methods for assessing priors for Bayesian-network structure and parameters, and methods for avoiding the overfitting of data including Monte-Carlo, Laplace, BIC, and MDL approximations. In Sections 3.13 and 3.14, we describe the relationships between Bayesian-network techniques and methods for supervised and unsupervised learning. In Section 3.15, we show how Bayesian networks facilitate the learning of causal relationships. In Section 3.16, we illustrate techniques discussed in the tutorial using a real-world case study. In Section 3.17, we give pointers to software and additional literature.

3.2 The Bayesian Approach to Probability and Statistics

To understand Bayesian networks and associated learning techniques, it is important to understand the Bayesian approach to probability and statistics. In this section, we provide an introduction to the Bayesian approach for those readers familiar only with the classical view.

In a nutshell, the Bayesian probability of an event x is a person's *degree of belief* in that event. Whereas a classical probability is a physical property of the world (e.g., the probability that a coin will land heads), a Bayesian probability is a property of the person who assigns the probability (e.g., your degree of belief that the coin will land heads). To keep these two concepts of probability distinct, we refer to the classical probability of an event as the true or physical probability of that event, and refer to a degree of belief in an event as a Bayesian or personal probability. Alternatively, when the meaning is clear, we refer to a Bayesian probability simply as a probability.

One important difference between physical probability and personal probability is that, to measure the latter, we do not need repeated trials. For example, imagine the repeated tosses of a sugar cube onto a wet surface. Every time the cube is tossed, its dimensions will change slightly. Thus, although the classical statistician has a hard time measuring the probability that the cube will land with a particular face up, the Bayesian simply restricts his or her attention to the next toss, and assigns a probability. As another example, consider the question: What is the probability that the Chicago Bulls will win the championship in 2001? Here, the classical statistician must remain silent, whereas the Bayesian can assign a probability (and perhaps make a bit of money in the process).

One common criticism of the Bayesian definition of probability is that probabilities seem arbitrary. Why should degrees of belief satisfy the rules of probability? On what scale should probabilities be measured? In particular, it makes sense to assign a probability of one (zero) to an event that will (not) occur, but what probabilities do we assign to beliefs that are not at the extremes? Not surprisingly, these questions have been studied intensely.

With regards to the first question, many researchers have suggested different sets of properties that should be satisfied by degrees of belief (e.g., Ramsey 1931, Cox 1946, Good 1950, Savage 1954, DeFinetti 1970). It turns out that each set of properties leads to the same rules: the rules of probability. Although each set

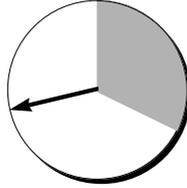


Fig. 3.1. The probability wheel: a tool for assessing probabilities

of properties is in itself compelling, the fact that different sets all lead to the rules of probability provides a particularly strong argument for using probability to measure beliefs.

The answer to the question of scale follows from a simple observation: people find it fairly easy to say that two events are equally likely. For example, imagine a simplified wheel of fortune having only two regions (shaded and not shaded), such as the one illustrated in Figure 3.1. Assuming everything about the wheel as symmetric (except for shading), you should conclude that it is equally likely for the wheel to stop in any one position. From this judgment and the sum rule of probability (probabilities of mutually exclusive and collectively exhaustive sum to one), it follows that your probability that the wheel will stop in the shaded region is the percent area of the wheel that is shaded (in this case, 0.3).

This *probability wheel* now provides a reference for measuring your probabilities of other events. For example, what is your probability that Al Gore will run on the Democratic ticket in 2000? First, ask yourself the question: Is it more likely that Gore will run or that the wheel when spun will stop in the shaded region? If you think that it is more likely that Gore will run, then imagine another wheel where the shaded region is larger. If you think that it is more likely that the wheel will stop in the shaded region, then imagine another wheel where the shaded region is smaller. Now, repeat this process until you think that Gore running and the wheel stopping in the shaded region are equally likely. At this point, your probability that Gore will run is just the percent surface area of the shaded area on the wheel.

In general, the process of measuring a degree of belief is commonly referred to as a *probability assessment*. The technique for assessment that we have just described is one of many available techniques discussed in the Management Science, Operations Research, and Psychology literature. One problem with probability assessment that is addressed in this literature is that of precision. Can one really say that his or her probability for event x is 0.601 and not 0.599? In most cases, no. Nonetheless, in most cases, probabilities are used to make decisions, and these decisions are not sensitive to small variations in probabilities. Well-established practices of *sensitivity analysis* help one to know when additional precision is unnecessary (e.g., Howard and Matheson, 1983). Another problem with probability assessment is that of accuracy. For example, recent experiences or the way a question is phrased can lead to assessments that do not reflect a person's true beliefs (Tversky and Kahneman, 1974). Methods for improving

accuracy can be found in the decision-analysis literature (e.g., Spetzler et al. (1975)).

Now let us turn to the issue of learning with data. To illustrate the Bayesian approach, consider a common thumbtack—one with a round, flat head that can be found in most supermarkets. If we throw the thumbtack up in the air, it will come to rest either on its point (*heads*) or on its head (*tails*).¹ Suppose we flip the thumbtack $N + 1$ times, making sure that the physical properties of the thumbtack and the conditions under which it is flipped remain stable over time. From the first N observations, we want to determine the probability of heads on the $N + 1$ th toss.

In the classical analysis of this problem, we assert that there is some physical probability of heads, which is unknown. We *estimate* this physical probability from the N observations using criteria such as low bias and low variance. We then use this estimate as our probability for heads on the $N + 1$ th toss. In the Bayesian approach, we also assert that there is some physical probability of heads, but we encode our uncertainty about this physical probability using (Bayesian) probabilities, and use the rules of probability to compute our probability of heads on the $N + 1$ th toss.²

To examine the Bayesian analysis of this problem, we need some notation. We denote a variable by an upper-case letter (e.g., X, Y, X_i, Θ), and the state or value of a corresponding variable by that same letter in lower case (e.g., x, y, x_i, θ). We denote a set of variables by a bold-face upper-case letter (e.g., $\mathbf{X}, \mathbf{Y}, \mathbf{X}_i$). We use a corresponding bold-face lower-case letter (e.g., $\mathbf{x}, \mathbf{y}, \mathbf{x}_i$) to denote an assignment of state or value to each variable in a given set. We say that variable set \mathbf{X} is in *configuration* \mathbf{x} . We use $p(X = x|\xi)$ (or $p(x|\xi)$ as a shorthand) to denote the probability that $X = x$ of a person with state of information ξ . We also use $p(x|\xi)$ to denote the probability distribution for X (both mass functions and density functions). Whether $p(x|\xi)$ refers to a probability, a probability density, or a probability distribution will be clear from context. We use this notation for probability throughout the paper. A summary of all notation is given at the end of the chapter.

Returning to the thumbtack problem, we define Θ to be a variable³ whose values θ correspond to the possible true values of the physical probability. We sometimes refer to θ as a *parameter*. We express the uncertainty about Θ using the probability density function $p(\theta|\xi)$. In addition, we use X_l to denote the variable representing the outcome of the l th flip, $l = 1, \dots, N + 1$, and $D = \{X_1 = x_1, \dots, X_N = x_N\}$ to denote the set of our observations. Thus, in Bayesian terms, the thumbtack problem reduces to computing $p(x_{N+1}|D, \xi)$ from $p(\theta|\xi)$.

¹ This example is taken from Howard (1970).

² Strictly speaking, a probability belongs to a single person, not a collection of people. Nonetheless, in parts of this discussion, we refer to “our” probability to avoid awkward English.

³ Bayesians typically refer to Θ as an *uncertain variable*, because the value of Θ is uncertain. In contrast, classical statisticians often refer to Θ as a *random variable*. In this text, we refer to Θ and all uncertain/random variables simply as variables.

To do so, we first use Bayes' rule to obtain the probability distribution for Θ given D and background knowledge ξ :

$$p(\theta|D, \xi) = \frac{p(\theta|\xi) p(D|\theta, \xi)}{p(D|\xi)} \quad (3.1)$$

where

$$p(D|\xi) = \int p(D|\theta, \xi) p(\theta|\xi) d\theta \quad (3.2)$$

Next, we expand the term $p(D|\theta, \xi)$. Both Bayesians and classical statisticians agree on this term: it is the likelihood function for binomial sampling. In particular, given the value of Θ , the observations in D are mutually independent, and the probability of heads (tails) on any one observation is θ ($1 - \theta$). Consequently, Equation 3.1 becomes

$$p(\theta|D, \xi) = \frac{p(\theta|\xi) \theta^h (1 - \theta)^t}{p(D|\xi)} \quad (3.3)$$

where h and t are the number of heads and tails observed in D , respectively. The probability distributions $p(\theta|\xi)$ and $p(\theta|D, \xi)$ are commonly referred to as the *prior* and *posterior* for Θ , respectively. The quantities h and t are said to be *sufficient statistics* for binomial sampling, because they provide a summarization of the data that is sufficient to compute the posterior from the prior. Finally, we average over the possible values of Θ (using the expansion rule of probability) to determine the probability that the $N + 1$ th toss of the thumbtack will come up heads:

$$\begin{aligned} p(X_{N+1} = \text{heads}|D, \xi) &= \int p(X_{N+1} = \text{heads}|\theta, \xi) p(\theta|D, \xi) d\theta \\ &= \int \theta p(\theta|D, \xi) d\theta \equiv E_{p(\theta|D, \xi)}(\theta) \end{aligned} \quad (3.4)$$

where $E_{p(\theta|D, \xi)}(\theta)$ denotes the expectation of θ with respect to the distribution $p(\theta|D, \xi)$.

To complete the Bayesian story for this example, we need a method to assess the prior distribution for Θ . A common approach, usually adopted for convenience, is to assume that this distribution is a *beta* distribution:

$$p(\theta|\xi) = \text{Beta}(\theta|\alpha_h, \alpha_t) \equiv \frac{\Gamma(\alpha)}{\Gamma(\alpha_h)\Gamma(\alpha_t)} \theta^{\alpha_h-1} (1 - \theta)^{\alpha_t-1} \quad (3.5)$$

where $\alpha_h > 0$ and $\alpha_t > 0$ are the parameters of the beta distribution, $\alpha = \alpha_h + \alpha_t$, and $\Gamma(\cdot)$ is the *Gamma* function which satisfies $\Gamma(x + 1) = x\Gamma(x)$ and $\Gamma(1) = 1$. The quantities α_h and α_t are often referred to as *hyperparameters* to distinguish them from the parameter θ . The hyperparameters α_h and α_t must be greater than zero so that the distribution can be normalized. Examples of beta distributions are shown in Figure 3.2.

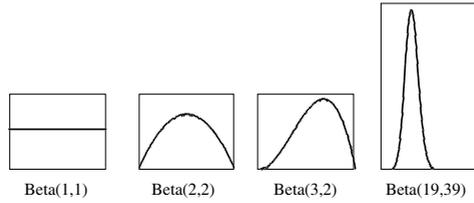


Fig. 3.2. Several beta distributions

The beta prior is convenient for several reasons. By Equation 3.3, the posterior distribution will also be a beta distribution:

$$p(\theta|D, \xi) = \frac{\Gamma(\alpha + N)}{\Gamma(\alpha_h + h)\Gamma(\alpha_t + t)} \theta^{\alpha_h + h - 1} (1 - \theta)^{\alpha_t + t - 1} = \text{Beta}(\theta|\alpha_h + h, \alpha_t + t) \quad (3.6)$$

We say that the set of beta distributions is a *conjugate family of distributions* for binomial sampling. Also, the expectation of θ with respect to this distribution has a simple form:

$$\int \theta \text{Beta}(\theta|\alpha_h, \alpha_t) d\theta = \frac{\alpha_h}{\alpha} \quad (3.7)$$

Hence, given a beta prior, we have a simple expression for the probability of heads in the $N + 1$ th toss:

$$p(X_{N+1} = \text{heads}|D, \xi) = \frac{\alpha_h + h}{\alpha + N} \quad (3.8)$$

Assuming $p(\theta|\xi)$ is a beta distribution, it can be assessed in a number of ways. For example, we can assess our probability for heads in the first toss of the thumbtack (e.g., using a probability wheel). Next, we can imagine having seen the outcomes of k flips, and reassess our probability for heads in the next toss. From Equation 3.8, we have (for $k = 1$)

$$p(X_1 = \text{heads}|\xi) = \frac{\alpha_h}{\alpha_h + \alpha_t} \quad p(X_2 = \text{heads}|X_1 = \text{heads}, \xi) = \frac{\alpha_h + 1}{\alpha_h + \alpha_t + 1}$$

Given these probabilities, we can solve for α_h and α_t . This assessment technique is known as the method of *imagined future data*.

Another assessment method is based on Equation 3.6. This equation says that, if we start with a $\text{Beta}(0, 0)$ prior⁴ and observe α_h heads and α_t tails, then our posterior (i.e., new prior) will be a $\text{Beta}(\alpha_h, \alpha_t)$ distribution. Recognizing that a $\text{Beta}(0, 0)$ prior encodes a state of minimum information, we can assess α_h and α_t by determining the (possibly fractional) number of observations of heads and tails that is equivalent to our actual knowledge about flipping thumbtacks. Alternatively, we can assess $p(X_1 = \text{heads}|\xi)$ and α , which can be regarded as an

⁴ Technically, the hyperparameters of this prior should be small positive numbers so that $p(\theta|\xi)$ can be normalized.

equivalent sample size for our current knowledge. This technique is known as the method of *equivalent samples*. Other techniques for assessing beta distributions are discussed by Winkler (1967) and Chaloner and Duncan (1983).

Although the beta prior is convenient, it is not accurate for some problems. For example, suppose we think that the thumbtack may have been purchased at a magic shop. In this case, a more appropriate prior may be a mixture of beta distributions—for example,

$$p(\theta|\xi) = 0.4 \text{Beta}(20, 1) + 0.4 \text{Beta}(1, 20) + 0.2 \text{Beta}(2, 2)$$

where 0.4 is our probability that the thumbtack is heavily weighted toward heads (tails). In effect, we have introduced an additional *hidden* or unobserved variable H , whose states correspond to the three possibilities: (1) thumbtack is biased toward heads, (2) thumbtack is biased toward tails, and (3) thumbtack is normal; and we have asserted that θ conditioned on each state of H is a beta distribution. In general, there are simple methods (e.g., the method of imagined future data) for determining whether or not a beta prior is an accurate reflection of one's beliefs. In those cases where the beta prior is inaccurate, an accurate prior can often be assessed by introducing additional hidden variables, as in this example.

So far, we have only considered observations drawn from a binomial distribution. In general, observations may be drawn from any physical probability distribution:

$$p(x|\theta, \xi) = f(x, \theta)$$

where $f(x, \theta)$ is the likelihood function with parameters θ . For purposes of this discussion, we assume that the number of parameters is finite. As an example, X may be a continuous variable and have a Gaussian physical probability distribution with mean μ and variance v :

$$p(x|\theta, \xi) = (2\pi v)^{-1/2} e^{-(x-\mu)^2/2v}$$

where $\theta = \{\mu, v\}$.

Regardless of the functional form, we can learn about the parameters given data using the Bayesian approach. As we have done in the binomial case, we define variables corresponding to the unknown parameters, assign priors to these variables, and use Bayes' rule to update our beliefs about these parameters given data:

$$p(\theta|D, \xi) = \frac{p(D|\theta, \xi) p(\theta|\xi)}{p(D|\xi)} \quad (3.9)$$

We then average over the possible values of θ to make predictions. For example,

$$p(x_{N+1}|D, \xi) = \int p(x_{N+1}|\theta, \xi) p(\theta|D, \xi) d\theta \quad (3.10)$$

For a class of distributions known as the *exponential family*, these computations can be done efficiently and in closed form.⁵ Members of this class include the

⁵ Recent advances in Monte-Carlo methods have made it possible to work efficiently with many distributions outside the exponential family. See, for example, Gilks et al. (1996).

binomial, multinomial, normal, Gamma, Poisson, and multivariate-normal distributions. Each member of this family has sufficient statistics that are of fixed dimension for any random sample, and a simple conjugate prior.⁶ Bernardo and Smith (pp. 436–442, 1994) have compiled the important quantities and Bayesian computations for commonly used members of the exponential family. Here, we summarize these items for multinomial sampling, which we use to illustrate many of the ideas in this paper.

In multinomial sampling, the observed variable X is discrete, having r possible states x^1, \dots, x^r . The likelihood function is given by

$$p(X = x^k | \boldsymbol{\theta}, \xi) = \theta_k, \quad k = 1, \dots, r$$

where $\boldsymbol{\theta} = \{\theta_2, \dots, \theta_r\}$ are the parameters. (The parameter θ_1 is given by $1 - \sum_{k=2}^r \theta_k$.) In this case, as in the case of binomial sampling, the parameters correspond to physical probabilities. The sufficient statistics for data set $D = \{X_1 = x_1, \dots, X_N = x_N\}$ are $\{N_1, \dots, N_r\}$, where N_i is the number of times $X = x^k$ in D . The simple conjugate prior used with multinomial sampling is the Dirichlet distribution:

$$p(\boldsymbol{\theta} | \xi) = \text{Dir}(\boldsymbol{\theta} | \alpha_1, \dots, \alpha_r) \equiv \frac{\Gamma(\alpha)}{\prod_{k=1}^r \Gamma(\alpha_k)} \prod_{k=1}^r \theta_k^{\alpha_k - 1} \quad (3.11)$$

where $\alpha = \sum_{i=1}^r \alpha_k$, and $\alpha_k > 0, k = 1, \dots, r$. The posterior distribution $p(\boldsymbol{\theta} | D, \xi) = \text{Dir}(\boldsymbol{\theta} | \alpha_1 + N_1, \dots, \alpha_r + N_r)$. Techniques for assessing the beta distribution, including the methods of imagined future data and equivalent samples, can also be used to assess Dirichlet distributions. Given this conjugate prior and data set D , the probability distribution for the next observation is given by

$$p(X_{N+1} = x^k | D, \xi) = \int \theta_k \text{Dir}(\boldsymbol{\theta} | \alpha_1 + N_1, \dots, \alpha_r + N_r) d\boldsymbol{\theta} = \frac{\alpha_k + N_k}{\alpha + N} \quad (3.12)$$

As we shall see, another important quantity in Bayesian analysis is the *marginal likelihood* or *evidence* $p(D | \xi)$. In this case, we have

$$p(D | \xi) = \frac{\Gamma(\alpha)}{\Gamma(\alpha + N)} \cdot \prod_{k=1}^r \frac{\Gamma(\alpha_k + N_k)}{\Gamma(\alpha_k)} \quad (3.13)$$

We note that the explicit mention of the state of knowledge ξ is useful, because it reinforces the notion that probabilities are subjective. Nonetheless, once this concept is firmly in place, the notation simply adds clutter. In the remainder of this tutorial, we shall not mention ξ explicitly.

In closing this section, we emphasize that, although the Bayesian and classical approaches may sometimes yield the same prediction, they are fundamentally different methods for learning from data. As an illustration, let us revisit the

⁶ In fact, except for a few, well-characterized exceptions, the exponential family is the only class of distributions that have sufficient statistics of fixed dimension (Koopman, 1936; Pitman, 1936).

thumbtack problem. Here, the Bayesian “estimate” for the physical probability of heads is obtained in a manner that is essentially the opposite of the classical approach.

Namely, in the classical approach, θ is fixed (albeit unknown), and we imagine all data sets of size N that *may be* generated by sampling from the binomial distribution determined by θ . Each data set D will occur with some probability $p(D|\theta)$ and will produce an estimate $\theta^*(D)$. To evaluate an estimator, we compute the expectation and variance of the estimate with respect to all such data sets:

$$\begin{aligned} E_{p(D|\theta)}(\theta^*) &= \sum_D p(D|\theta) \theta^*(D) \\ \text{Var}_{p(D|\theta)}(\theta^*) &= \sum_D p(D|\theta) (\theta^*(D) - E_{p(D|\theta)}(\theta^*))^2 \end{aligned} \quad (3.14)$$

We then choose an estimator that somehow balances the bias ($\theta - E_{p(D|\theta)}(\theta^*)$) and variance of these estimates over the possible values for θ .⁷ Finally, we apply this estimator to the data set that we actually observe. A commonly-used estimator is the maximum-likelihood (ML) estimator, which selects the value of θ that maximizes the likelihood $p(D|\theta)$. For binomial sampling, we have

$$\theta_{\text{ML}}^*(D) = \frac{N_k}{\sum_{k=1}^r N_k}$$

For this (and other types) of sampling, the ML estimator is *unbiased*. That is, for all values of θ , the ML estimator has zero bias. In addition, for all values of θ , the variance of the ML estimator is no greater than that of any other unbiased estimator (see, e.g., Schervish, 1995).

In contrast, in the Bayesian approach, D is fixed, and we imagine all possible values of θ from which this data set *could have been* generated. Given θ , the “estimate” of the physical probability of heads is just θ itself. Nonetheless, we are uncertain about θ , and so our final estimate is the expectation of θ with respect to our posterior beliefs about its value:

$$E_{p(\theta|D,\xi)}(\theta) = \int \theta p(\theta|D,\xi) d\theta \quad (3.15)$$

The expectations in Equations 3.14 and 3.15 are different and, in many cases, lead to different “estimates”. One way to frame this difference is to say that the classical and Bayesian approaches have different definitions for what it means to be a good estimator. Both solutions are “correct” in that they are self consistent. Unfortunately, both methods have their drawbacks, which has led to endless debates about the merit of each approach. For example, Bayesians argue that it does not make sense to consider the expectations in Equation 3.14, because we only see a single data set. If we saw more than one data set, we should

⁷ Low bias and variance are not the only desirable properties of an estimator. Other desirable properties include consistency and robustness.

combine them into one larger data set. In contrast, classical statisticians argue that sufficiently accurate priors can not be assessed in many situations. The common view that seems to be emerging is that one should use whatever method that is most sensible for the task at hand. We share this view, although we also believe that the Bayesian approach has been under used, especially in light of its advantages mentioned in the introduction (points three and four). Consequently, in this paper, we concentrate on the Bayesian approach.

3.3 Bayesian Networks

So far, we have considered only simple problems with one or a few variables. In real learning problems, however, we are typically interested in looking for relationships among a large number of variables. The Bayesian network is a representation suited to this task. It is a graphical model that efficiently encodes the joint probability distribution (physical or Bayesian) for a large set of variables. In this section, we define a Bayesian network and show how one can be constructed from prior knowledge.

A Bayesian network for a set of variables $\mathbf{X} = \{X_1, \dots, X_n\}$ consists of (1) a network structure S that encodes a set of conditional independence assertions about variables in \mathbf{X} , and (2) a set P of local probability distributions associated with each variable. Together, these components define the joint probability distribution for \mathbf{X} . The network structure S is a directed acyclic graph. The nodes in S are in one-to-one correspondence with the variables \mathbf{X} . We use X_i to denote both the variable and its corresponding node, and \mathbf{Pa}_i to denote the parents of node X_i in S as well as the variables corresponding to those parents. The *lack* of possible arcs in S encode conditional independencies. In particular, given structure S , the joint probability distribution for \mathbf{X} is given by

$$p(\mathbf{x}) = \prod_{i=1}^n p(x_i | \mathbf{pa}_i) \quad (3.16)$$

The local probability distributions P are the distributions corresponding to the terms in the product of Equation 3.16. Consequently, the pair (S, P) encodes the joint distribution $p(\mathbf{x})$.

The probabilities encoded by a Bayesian network may be Bayesian or physical. When building Bayesian networks from prior knowledge alone, the probabilities will be Bayesian. When learning these networks from data, the probabilities will be physical (and their values may be uncertain). In subsequent sections, we describe how we can learn the structure and probabilities of a Bayesian network from data. In the remainder of this section, we explore the construction of Bayesian networks from prior knowledge. As we shall see in Section 3.10, this procedure can be useful in learning Bayesian networks as well.

To illustrate the process of building a Bayesian network, consider the problem of detecting credit-card fraud. We begin by determining the variables to model. One possible choice of variables for our problem is *Fraud* (F), *Gas* (G), *Jewelry*

(J), *Age* (A), and *Sex* (S), representing whether or not the current purchase is fraudulent, whether or not there was a gas purchase in the last 24 hours, whether or not there was a jewelry purchase in the last 24 hours, and the age and sex of the card holder, respectively. The states of these variables are shown in Figure 3.3. Of course, in a realistic problem, we would include many more variables. Also, we could model the states of one or more of these variables at a finer level of detail. For example, we could let *Age* be a continuous variable.

This initial task is not always straightforward. As part of this task we must (1) correctly identify the goals of modeling (e.g., prediction versus explanation versus exploration), (2) identify many possible observations that may be relevant to the problem, (3) determine what subset of those observations is worthwhile to model, and (4) organize the observations into variables having mutually exclusive and collectively exhaustive states. Difficulties here are not unique to modeling with Bayesian networks, but rather are common to most approaches. Although there are no clean solutions, some guidance is offered by decision analysts (e.g., Howard and Matheson, 1983) and (when data are available) statisticians (e.g., Tukey, 1977).

In the next phase of Bayesian-network construction, we build a directed acyclic graph that encodes assertions of conditional independence. One approach for doing so is based on the following observations. From the chain rule of probability, we have

$$p(\mathbf{x}) = \prod_{i=1}^n p(x_i | x_1, \dots, x_{i-1}) \quad (3.17)$$

Now, for every X_i , there will be some subset $\Pi_i \subseteq \{X_1, \dots, X_{i-1}\}$ such that X_i and $\{X_1, \dots, X_{i-1}\} \setminus \Pi_i$ are conditionally independent given Π_i . That is, for any \mathbf{x} ,

$$p(x_i | x_1, \dots, x_{i-1}) = p(x_i | \pi_i) \quad (3.18)$$

Combining Equations 3.17 and 3.18, we obtain

$$p(\mathbf{x}) = \prod_{i=1}^n p(x_i | \pi_i) \quad (3.19)$$

Comparing Equations 3.16 and 3.19, we see that the variables sets (Π_1, \dots, Π_n) correspond to the Bayesian-network parents $(\mathbf{Pa}_1, \dots, \mathbf{Pa}_n)$, which in turn fully specify the arcs in the network structure S .

Consequently, to determine the structure of a Bayesian network we (1) order the variables somehow, and (2) determine the variables sets that satisfy Equation 3.18 for $i = 1, \dots, n$. In our example, using the ordering (F, A, S, G, J) , we have the conditional independencies

$$\begin{aligned} p(a|f) &= p(a) \\ p(s|f, a) &= p(s) \\ p(g|f, a, s) &= p(g|f) \\ p(j|f, a, s, g) &= p(j|f, a, s) \end{aligned} \quad (3.20)$$

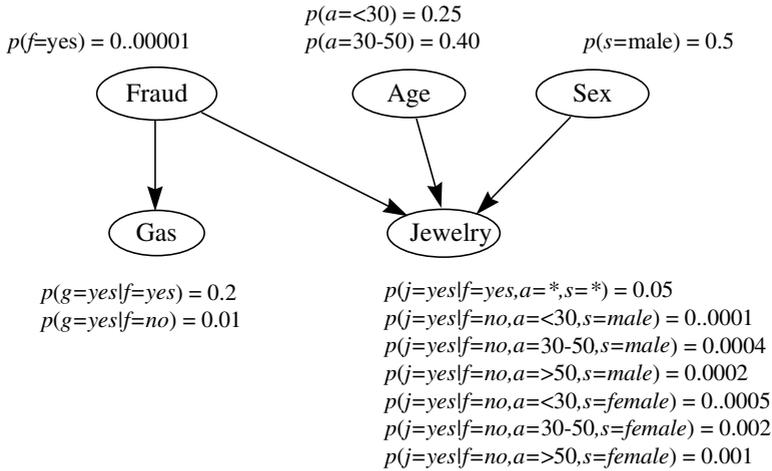


Fig. 3.3. A Bayesian-network for detecting credit-card fraud. Arcs are drawn from cause to effect. The local probability distribution(s) associated with a node are shown adjacent to the node. An asterisk is a shorthand for “any state.”

Thus, we obtain the structure shown in Figure 3.3.

This approach has a serious drawback. If we choose the variable order carelessly, the resulting network structure may fail to reveal many conditional independencies among the variables. For example, if we construct a Bayesian network for the fraud problem using the ordering (J, G, S, A, F) , we obtain a fully connected network structure. Thus, in the worst case, we have to explore $n!$ variable orderings to find the best one. Fortunately, there is another technique for constructing Bayesian networks that does not require an ordering. The approach is based on two observations: (1) people can often readily assert causal relationships among variables, and (2) causal relationships typically correspond to assertions of conditional dependence. In particular, to construct a Bayesian network for a given set of variables, we simply draw arcs from cause variables to their immediate effects. In almost all cases, doing so results in a network structure that satisfies the definition Equation 3.16. For example, given the assertions that *Fraud* is a direct cause of *Gas*, and *Fraud*, *Age*, and *Sex* are direct causes of *Jewelry*, we obtain the network structure in Figure 3.3. The causal semantics of Bayesian networks are in large part responsible for the success of Bayesian networks as a representation for expert systems (Heckerman et al., 1995a). In Section 3.15, we will see how to learn causal relationships from data using these causal semantics.

In the final step of constructing a Bayesian network, we assess the local probability distribution(s) $p(x_i | \mathbf{pa}_i)$. In our fraud example, where all variables are discrete, we assess one distribution for X_i for every configuration of \mathbf{pa}_i . Example distributions are shown in Figure 3.3.

Note that, although we have described these construction steps as a simple sequence, they are often intermingled in practice. For example, judgments of conditional independence and/or cause and effect can influence problem formulation. Also, assessments of probability can lead to changes in the network structure. Exercises that help one gain familiarity with the practice of building Bayesian networks can be found in Jensen (1996).

3.4 Inference in a Bayesian Network

Once we have constructed a Bayesian network (from prior knowledge, data, or a combination), we usually need to determine various probabilities of interest from the model. For example, in our problem concerning fraud detection, we want to know the probability of fraud given observations of the other variables. This probability is not stored directly in the model, and hence needs to be computed. In general, the computation of a probability of interest given a model is known as *probabilistic inference*. In this section we describe probabilistic inference in Bayesian networks.

Because a Bayesian network for \mathbf{X} determines a joint probability distribution for \mathbf{X} , we can—in principle—use the Bayesian network to compute any probability of interest. For example, from the Bayesian network in Figure 3.3, the probability of fraud given observations of the other variables can be computed as follows:

$$p(f|a, s, g, j) = \frac{p(f, a, s, g, j)}{p(a, s, g, j)} = \frac{p(f, a, s, g, j)}{\sum_{f'} p(f', a, s, g, j)} \quad (3.21)$$

For problems with many variables, however, this direct approach is not practical. Fortunately, at least when all variables are discrete, we can exploit the conditional independencies encoded in a Bayesian network to make this computation more efficient. In our example, given the conditional independencies in Equation 3.20, Equation 3.21 becomes

$$\begin{aligned} p(f|a, s, g, j) &= \frac{p(f)p(a)p(s)p(g|f)p(j|f, a, s)}{\sum_{f'} p(f')p(a)p(s)p(g|f')p(j|f', a, s)} \\ &= \frac{p(f)p(g|f)p(j|f, a, s)}{\sum_{f'} p(f')p(g|f')p(j|f', a, s)} \end{aligned} \quad (3.22)$$

Several researchers have developed probabilistic inference algorithms for Bayesian networks with discrete variables that exploit conditional independence roughly as we have described, although with different twists. For example, Howard and Matheson (1981), Olmsted (1983), and Shachter (1988) developed an algorithm that reverses arcs in the network structure until the answer to the given probabilistic query can be read directly from the graph. In this algorithm, each arc reversal corresponds to an application of Bayes' theorem. Pearl (1986) developed a message-passing scheme that updates the probability distributions for each node in a Bayesian network in response to observations of one or more

variables. Lauritzen and Spiegelhalter (1988), Jensen et al. (1990), and Dawid (1992) created an algorithm that first transforms the Bayesian network into a tree where each node in the tree corresponds to a subset of variables in \mathbf{X} . The algorithm then exploits several mathematical properties of this tree to perform probabilistic inference. Most recently, D'Ambrosio (1991) developed an inference algorithm that simplifies sums and products symbolically, as in the transformation from Equation 3.21 to 3.22. The most commonly used algorithm for discrete variables is that of Lauritzen and Spiegelhalter (1988), Jensen et al (1990), and Dawid (1992).

Methods for exact inference in Bayesian networks that encode multivariate-Gaussian or Gaussian-mixture distributions have been developed by Shachter and Kenley (1989) and Lauritzen (1992), respectively. These methods also use assertions of conditional independence to simplify inference. Approximate methods for inference in Bayesian networks with other distributions, such as the generalized linear-regression model, have also been developed (Saul et al., 1996; Jaakkola and Jordan, 1996).

Although we use conditional independence to simplify probabilistic inference, exact inference in an arbitrary Bayesian network for discrete variables is NP-hard (Cooper, 1990). Even approximate inference (for example, Monte-Carlo methods) is NP-hard (Dagum and Luby, 1993). The source of the difficulty lies in undirected cycles in the Bayesian-network structure—cycles in the structure where we ignore the directionality of the arcs. (If we add an arc from *Age* to *Gas* in the network structure of Figure 3.3, then we obtain a structure with one undirected cycle: $F-G-A-J-F$.) When a Bayesian-network structure contains many undirected cycles, inference is intractable. For many applications, however, structures are simple enough (or can be simplified sufficiently without sacrificing much accuracy) so that inference is efficient. For those applications where generic inference methods are impractical, researchers are developing techniques that are custom tailored to particular network topologies (Heckerman 1989; Suermondt and Cooper, 1991; Saul et al., 1996; Jaakkola and Jordan, 1996) or to particular inference queries (Ramamurthi and Agogino, 1988; Shachter et al., 1990; Jensen and Andersen, 1990; Darwiche and Provan, 1996).

3.5 Learning Probabilities in a Bayesian Network

In the next several sections, we show how to refine the structure and local probability distributions of a Bayesian network given data. The result is set of techniques for data analysis that combines prior knowledge with data to produce improved knowledge. In this section, we consider the simplest version of this problem: using data to update the probabilities of a given Bayesian network structure.

Recall that, in the thumbtack problem, we do not learn the probability of heads. Instead, we update our posterior distribution for the variable that represents the physical probability of heads. We follow the same approach for probabilities in a Bayesian network. In particular, we assume—perhaps from causal

knowledge about the problem—that the physical joint probability distribution for \mathbf{X} can be encoded in some network structure S . We write

$$p(\mathbf{x}|\boldsymbol{\theta}_s, S^h) = \prod_{i=1}^n p(x_i|\mathbf{pa}_i, \boldsymbol{\theta}_i, S^h) \quad (3.23)$$

where $\boldsymbol{\theta}_i$ is the vector of parameters for the distribution $p(x_i|\mathbf{pa}_i, \boldsymbol{\theta}_i, S^h)$, $\boldsymbol{\theta}_s$ is the vector of parameters $(\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_n)$, and S^h denotes the event (or “hypothesis” in statistics nomenclature) that the physical joint probability distribution can be factored according to S .⁸ In addition, we assume that we have a random sample $D = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ from the physical joint probability distribution of \mathbf{X} . We refer to an element \mathbf{x}_i of D as a *case*. As in Section 3.2, we encode our uncertainty about the parameters $\boldsymbol{\theta}_s$ by defining a (vector-valued) variable Θ_s , and assessing a prior probability density function $p(\boldsymbol{\theta}_s|S^h)$. The problem of learning probabilities in a Bayesian network can now be stated simply: Given a random sample D , compute the posterior distribution $p(\boldsymbol{\theta}_s|D, S^h)$.

We refer to the distribution $p(x_i|\mathbf{pa}_i, \boldsymbol{\theta}_i, S^h)$, viewed as a function of $\boldsymbol{\theta}_i$, as a *local distribution function*. Readers familiar with methods for supervised learning will recognize that a local distribution function is nothing more than a probabilistic classification or regression function. Thus, a Bayesian network can be viewed as a collection of probabilistic classification/regression models, organized by conditional-independence relationships. Examples of classification/regression models that produce probabilistic outputs include linear regression, generalized linear regression, probabilistic neural networks (e.g., MacKay, 1992a, 1992b), probabilistic decision trees (e.g., Buntine, 1993; Friedman and Goldszmidt, 1996), kernel density estimation methods (Book, 1994), and dictionary methods (Friedman, 1995). In principle, any of these forms can be used to learn probabilities in a Bayesian network; and, in most cases, Bayesian techniques for learning are available. Nonetheless, the most studied models include the unrestricted multinomial distribution (e.g., Cooper and Herskovits, 1992), linear regression with Gaussian noise (e.g., Buntine, 1994; Heckerman and Geiger, 1996), and generalized linear regression (e.g., MacKay, 1992a and 1992b; Neal, 1993; and Saul et al., 1996).

In this tutorial, we illustrate the basic ideas for learning probabilities (and structure) using the unrestricted multinomial distribution. In this case, each variable $X_i \in \mathbf{X}$ is discrete, having r_i possible values $x_i^1, \dots, x_i^{r_i}$, and each local distribution function is collection of multinomial distributions, one distribution for each configuration of \mathbf{Pa}_i . Namely, we assume

$$p(x_i^k|\mathbf{pa}_i^j, \boldsymbol{\theta}_i, S^h) = \theta_{ijk} > 0 \quad (3.24)$$

⁸ As defined here, network-structure hypotheses overlap. For example, given $\mathbf{X} = \{X_1, X_2\}$, any joint distribution for \mathbf{X} that can be factored according the network structure containing no arc, can also be factored according to the network structure $X_1 \rightarrow X_2$. Such overlap presents problems for model averaging, described in Section 3.7. Therefore, we should add conditions to the definition to insure no overlap. Heckerman and Geiger (1996) describe one such set of conditions.

where $\mathbf{pa}_i^1, \dots, \mathbf{pa}_i^{q_i}$ ($q_i = \prod_{X_i \in \mathbf{Pa}_i} r_i$) denote the configurations of \mathbf{Pa}_i , and $\theta_i = ((\theta_{ijk})_{k=2}^{r_i})_{j=1}^{q_i}$ are the parameters. (The parameter θ_{ij1} is given by $1 - \sum_{k=2}^{r_i} \theta_{ijk}$.) For convenience, we define the vector of parameters

$$\theta_{ij} = (\theta_{ij2}, \dots, \theta_{ijr_i})$$

for all i and j . We use the term “unrestricted” to contrast this distribution with multinomial distributions that are low-dimensional functions of \mathbf{Pa}_i —for example, the generalized linear-regression model.

Given this class of local distribution functions, we can compute the posterior distribution $p(\theta_s | D, S^h)$ efficiently and in closed form under two assumptions. The first assumption is that there are no missing data in the random sample D . We say that the random sample D is *complete*. The second assumption is that the parameter vectors θ_{ij} are mutually independent.⁹ That is,

$$p(\theta_s | S^h) = \prod_{i=1}^n \prod_{j=1}^{q_i} p(\theta_{ij} | S^h)$$

We refer to this assumption, which was introduced by Spiegelhalter and Lauritzen (1990), as *parameter independence*.

Given that the joint physical probability distribution factors according to some network structure S , the assumption of parameter independence can itself be represented by a larger Bayesian-network structure. For example, the network structure in Figure 3.4 represents the assumption of parameter independence for $\mathbf{X} = \{X, Y\}$ (X, Y binary) and the hypothesis that the network structure $X \rightarrow Y$ encodes the physical joint probability distribution for \mathbf{X} .

Under the assumptions of complete data and parameter independence, the parameters remain independent given a random sample:

$$p(\theta_s | D, S^h) = \prod_{i=1}^n \prod_{j=1}^{q_i} p(\theta_{ij} | D, S^h) \quad (3.25)$$

Thus, we can update each vector of parameters θ_{ij} independently, just as in the one-variable case. Assuming each vector θ_{ij} has the prior distribution $\text{Dir}(\theta_{ij} | \alpha_{ij1}, \dots, \alpha_{ijr_i})$, we obtain the posterior distribution

$$p(\theta_{ij} | D, S^h) = \text{Dir}(\theta_{ij} | \alpha_{ij1} + N_{ij1}, \dots, \alpha_{ijr_i} + N_{ijr_i}) \quad (3.26)$$

where N_{ijk} is the number of cases in D in which $X_i = x_i^k$ and $\mathbf{Pa}_i = \mathbf{pa}_i^j$.

As in the thumbtack example, we can average over the possible configurations of θ_s to obtain predictions of interest. For example, let us compute

⁹ The computation is also straightforward if two or more parameters are equal. For details, see Thiesson (1995).

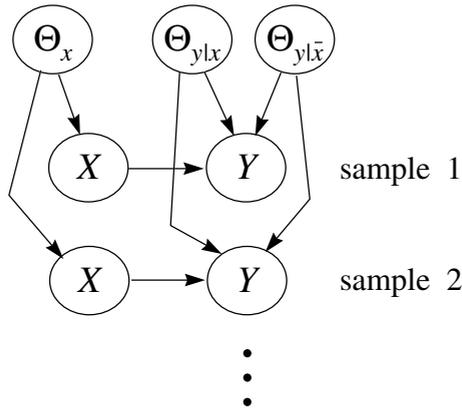


Fig. 3.4. A Bayesian-network structure depicting the assumption of parameter independence for learning the parameters of the network structure $X \rightarrow Y$. Both variables X and Y are binary. We use x and \bar{x} to denote the two states of X , and y and \bar{y} to denote the two states of Y .

$p(\mathbf{x}_{N+1}|D, S^h)$, where \mathbf{x}_{N+1} is the next case to be seen after D . Suppose that, in case \mathbf{x}_{N+1} , $X_i = x_i^k$ and $\mathbf{Pa}_i = \mathbf{pa}_i^j$, where k and j depend on i . Thus,

$$p(\mathbf{x}_{N+1}|D, S^h) = E_{p(\boldsymbol{\theta}_s|D, S^h)} \left(\prod_{i=1}^n \theta_{ijk} \right)$$

To compute this expectation, we first use the fact that the parameters remain independent given D :

$$p(\mathbf{x}_{N+1}|D, S^h) = \int \prod_{i=1}^n \theta_{ijk} p(\boldsymbol{\theta}_s|D, S^h) d\boldsymbol{\theta}_s = \prod_{i=1}^n \int \theta_{ijk} p(\theta_{ij}|D, S^h) d\theta_{ij}$$

Then, we use Equation 3.12 to obtain

$$p(\mathbf{x}_{N+1}|D, S^h) = \prod_{i=1}^n \frac{\alpha_{ijk} + N_{ijk}}{\alpha_{ij} + N_{ij}} \quad (3.27)$$

where $\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$ and $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$.

These computations are simple because the unrestricted multinomial distributions are in the exponential family. Computations for linear regression with Gaussian noise are equally straightforward (Buntine, 1994; Heckerman and Geiger, 1996).

3.6 Methods for Incomplete Data

Let us now discuss methods for learning about parameters when the random sample is incomplete (i.e., some variables in some cases are not observed). An

important distinction concerning missing data is whether or not the absence of an observation is dependent on the actual states of the variables. For example, a missing datum in a drug study may indicate that a patient became too sick—perhaps due to the side effects of the drug—to continue in the study. In contrast, if a variable is hidden (i.e., never observed in any case), then the absence of this data is independent of state. Although Bayesian methods and graphical models are suited to the analysis of both situations, methods for handling missing data where absence is independent of state are simpler than those where absence and state are dependent. In this tutorial, we concentrate on the simpler situation only. Readers interested in the more complicated case should see Rubin (1978), Robins (1986), and Pearl (1995).

Continuing with our example using unrestricted multinomial distributions, suppose we observe a single incomplete case. Let $\mathbf{Y} \subset \mathbf{X}$ and $\mathbf{Z} \subset \mathbf{X}$ denote the observed and unobserved variables in the case, respectively. Under the assumption of parameter independence, we can compute the posterior distribution of θ_{ij} for network structure S as follows:

$$\begin{aligned} p(\theta_{ij}|\mathbf{y}, S^h) &= \sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{y}, S^h) p(\theta_{ij}|\mathbf{y}, \mathbf{z}, S^h) \\ &= (1 - p(\mathbf{pa}_i^j|\mathbf{y}, S^h)) \{p(\theta_{ij}|S^h)\} + \sum_{k=1}^{r_i} p(x_i^k, \mathbf{pa}_i^j|\mathbf{y}, S^h) \{p(\theta_{ij}|x_i^k, \mathbf{pa}_i^j, S^h)\} \end{aligned} \quad (3.28)$$

(See Spiegelhalter and Lauritzen (1990) for a derivation.) Each term in curly brackets in Equation 3.28 is a Dirichlet distribution. Thus, unless both X_i and all the variables in \mathbf{Pa}_i are observed in case \mathbf{y} , the posterior distribution of θ_{ij} will be a linear combination of Dirichlet distributions—that is, a Dirichlet mixture with mixing coefficients $(1 - p(\mathbf{pa}_i^j|\mathbf{y}, S^h))$ and $p(x_i^k, \mathbf{pa}_i^j|\mathbf{y}, S^h)$, $k = 1, \dots, r_i$.

When we observe a second incomplete case, some or all of the Dirichlet components in Equation 3.28 will again split into Dirichlet mixtures. That is, the posterior distribution for θ_{ij} we become a mixture of Dirichlet mixtures. As we continue to observe incomplete cases, each missing values for \mathbf{Z} , the posterior distribution for θ_{ij} will contain a number of components that is exponential in the number of cases. In general, for any interesting set of local likelihoods and priors, the exact computation of the posterior distribution for θ_s will be intractable. Thus, we require an approximation for incomplete data.

3.6.1 Monte-Carlo Methods

One class of approximations is based on Monte-Carlo or sampling methods. These approximations can be extremely accurate, provided one is willing to wait long enough for the computations to converge.

In this section, we discuss one of many Monte-Carlo methods known as *Gibbs sampling*, introduced by Geman and Geman (1984). Given variables $\mathbf{X} = \{X_1, \dots, X_n\}$ with some joint distribution $p(\mathbf{x})$, we can use a Gibbs sampler to approximate the expectation of a function $f(\mathbf{x})$ with respect to $p(\mathbf{x})$ as

follows. First, we choose an initial state for each of the variables in \mathbf{X} somehow (e.g., at random). Next, we pick some variable X_i , unassign its current state, and compute its probability distribution given the states of the other $n - 1$ variables. Then, we sample a state for X_i based on this probability distribution, and compute $f(\mathbf{x})$. Finally, we iterate the previous two steps, keeping track of the average value of $f(\mathbf{x})$. In the limit, as the number of cases approach infinity, this average is equal to $E_{p(\mathbf{x})}(f(\mathbf{x}))$ provided two conditions are met. First, the Gibbs sampler must be *irreducible*: The probability distribution $p(\mathbf{x})$ must be such that we can eventually sample any possible configuration of \mathbf{X} given any possible initial configuration of \mathbf{X} . For example, if $p(\mathbf{x})$ contains no zero probabilities, then the Gibbs sampler will be irreducible. Second, each X_i must be chosen infinitely often. In practice, an algorithm for deterministically rotating through the variables is typically used. Introductions to Gibbs sampling and other Monte-Carlo methods—including methods for initialization and a discussion of convergence—are given by Neal (1993) and Madigan and York (1995).

To illustrate Gibbs sampling, let us approximate the probability density $p(\boldsymbol{\theta}_s|D, S^h)$ for some particular configuration of $\boldsymbol{\theta}_s$, given an incomplete data set $D = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ and a Bayesian network for discrete variables with independent Dirichlet priors. To approximate $p(\boldsymbol{\theta}_s|D, S^h)$, we first initialize the states of the unobserved variables in each case somehow. As a result, we have a complete random sample D_c . Second, we choose some variable X_{il} (variable X_i in case l) that is not observed in the original random sample D , and reassign its state according to the probability distribution

$$p(x'_{il}|D_c \setminus x_{il}, S^h) = \frac{p(x'_{il}, D_c \setminus x_{il}|S^h)}{\sum_{x''_{il}} p(x''_{il}, D_c \setminus x_{il}|S^h)}$$

where $D_c \setminus x_{il}$ denotes the data set D_c with observation x_{il} removed, and the sum in the denominator runs over all states of variable X_{il} . As we shall see in Section 3.7, the terms in the numerator and denominator can be computed efficiently (see Equation 3.35). Third, we repeat this reassignment for all unobserved variables in D , producing a new complete random sample D'_c . Fourth, we compute the posterior density $p(\boldsymbol{\theta}_s|D'_c, S^h)$ as described in Equations 3.25 and 3.26. Finally, we iterate the previous three steps, and use the average of $p(\boldsymbol{\theta}_s|D'_c, S^h)$ as our approximation.

3.6.2 The Gaussian Approximation

Monte-Carlo methods yield accurate results, but they are often intractable—for example, when the sample size is large. Another approximation that is more efficient than Monte-Carlo methods and often accurate for relatively large samples is the *Gaussian approximation* (e.g., Kass et al., 1988; Kass and Raftery, 1995).

The idea behind this approximation is that, for large amounts of data, $p(\boldsymbol{\theta}_s|D, S^h) \propto p(D|\boldsymbol{\theta}_s, S^h) \cdot p(\boldsymbol{\theta}_s|S^h)$ can often be approximated as a multivariate-Gaussian distribution.

In particular, let

$$g(\boldsymbol{\theta}_s) \equiv \log(p(D|\boldsymbol{\theta}_s, S^h) \cdot p(\boldsymbol{\theta}_s|S^h)) \quad (3.29)$$

Also, define $\tilde{\boldsymbol{\theta}}_s$ to be the configuration of $\boldsymbol{\theta}_s$ that maximizes $g(\boldsymbol{\theta}_s)$. This configuration also maximizes $p(\boldsymbol{\theta}_s|D, S^h)$, and is known as the *maximum a posteriori* (MAP) configuration of $\boldsymbol{\theta}_s$. Using a second degree Taylor polynomial of $g(\boldsymbol{\theta}_s)$ about the $\tilde{\boldsymbol{\theta}}_s$ to approximate $g(\boldsymbol{\theta}_s)$, we obtain

$$g(\boldsymbol{\theta}_s) \approx g(\tilde{\boldsymbol{\theta}}_s) - \frac{1}{2}(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s)A(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s)^t \quad (3.30)$$

where $(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s)^t$ is the transpose of row vector $(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s)$, and A is the negative Hessian of $g(\boldsymbol{\theta}_s)$ evaluated at $\tilde{\boldsymbol{\theta}}_s$. Raising $g(\boldsymbol{\theta}_s)$ to the power of e and using Equation 3.29, we obtain

$$\begin{aligned} p(\boldsymbol{\theta}_s|D, S^h) &\propto p(D|\boldsymbol{\theta}_s, S^h) p(\boldsymbol{\theta}_s|S^h) \\ &\approx p(D|\tilde{\boldsymbol{\theta}}_s, S^h) p(\tilde{\boldsymbol{\theta}}_s|S^h) \exp\left\{-\frac{1}{2}(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s)A(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s)^t\right\} \end{aligned} \quad (3.31)$$

Hence, $p(\boldsymbol{\theta}_s|D, S^h)$ is approximately Gaussian.

To compute the Gaussian approximation, we must compute $\tilde{\boldsymbol{\theta}}_s$ as well as the negative Hessian of $g(\boldsymbol{\theta}_s)$ evaluated at $\tilde{\boldsymbol{\theta}}_s$. In the following section, we discuss methods for finding $\tilde{\boldsymbol{\theta}}_s$. Meng and Rubin (1991) describe a numerical technique for computing the second derivatives. Raftery (1995) shows how to approximate the Hessian using likelihood-ratio tests that are available in many statistical packages. Thiesson (1995) demonstrates that, for unrestricted multinomial distributions, the second derivatives can be computed using Bayesian-network inference.

3.6.3 The MAP and ML Approximations and the EM Algorithm

As the sample size of the data increases, the Gaussian peak will become sharper, tending to a delta function at the MAP configuration $\tilde{\boldsymbol{\theta}}_s$. In this limit, we do not need to compute averages or expectations. Instead, we simply make predictions based on the MAP configuration.

A further approximation is based on the observation that, as the sample size increases, the effect of the prior $p(\boldsymbol{\theta}_s|S^h)$ diminishes. Thus, we can approximate $\tilde{\boldsymbol{\theta}}_s$ by the maximum *maximum likelihood* (ML) configuration of $\boldsymbol{\theta}_s$:

$$\hat{\boldsymbol{\theta}}_s = \arg \max_{\boldsymbol{\theta}_s} \{p(D|\boldsymbol{\theta}_s, S^h)\}$$

One class of techniques for finding a ML or MAP is gradient-based optimization. For example, we can use gradient ascent, where we follow the derivatives of $g(\boldsymbol{\theta}_s)$ or the likelihood $p(D|\boldsymbol{\theta}_s, S^h)$ to a local maximum. Russell et al. (1995) and Thiesson (1995) show how to compute the derivatives of the likelihood for a Bayesian network with unrestricted multinomial distributions. Buntine

(1994) discusses the more general case where the likelihood function comes from the exponential family. Of course, these gradient-based methods find only local maxima.

Another technique for finding a local ML or MAP is the expectation–maximization (EM) algorithm (Dempster et al., 1977). To find a local MAP or ML, we begin by assigning a configuration to θ_s somehow (e.g., at random). Next, we compute the *expected sufficient statistics* for a complete data set, where expectation is taken with respect to the joint distribution for \mathbf{X} conditioned on the assigned configuration of θ_s and the known data D . In our discrete example, we compute

$$E_{p(\mathbf{x}|D, \theta_s, S^h)}(N_{ijk}) = \sum_{l=1}^N p(x_i^k, \mathbf{pa}_i^j | \mathbf{y}_l, \theta_s, S^h) \quad (3.32)$$

where \mathbf{y}_l is the possibly incomplete l th case in D . When X_i and all the variables in \mathbf{Pa}_i are observed in case \mathbf{x}_l , the term for this case requires a trivial computation: it is either zero or one. Otherwise, we can use any Bayesian network inference algorithm to evaluate the term. This computation is called the *expectation step* of the EM algorithm.

Next, we use the expected sufficient statistics as if they were actual sufficient statistics from a complete random sample D_c . If we are doing an ML calculation, then we determine the configuration of θ_s that maximize $p(D_c | \theta_s, S^h)$. In our discrete example, we have

$$\theta_{ijk} = \frac{E_{p(\mathbf{x}|D, \theta_s, S^h)}(N_{ijk})}{\sum_{k=1}^{r_i} E_{p(\mathbf{x}|D, \theta_s, S^h)}(N_{ijk})}$$

If we are doing a MAP calculation, then we determine the configuration of θ_s that maximizes $p(\theta_s | D_c, S^h)$. In our discrete example, we have¹⁰

$$\theta_{ijk} = \frac{\alpha_{ijk} + E_{p(\mathbf{x}|D, \theta_s, S^h)}(N_{ijk})}{\sum_{k=1}^{r_i} (\alpha_{ijk} + E_{p(\mathbf{x}|D, \theta_s, S^h)}(N_{ijk}))}$$

This assignment is called the *maximization step* of the EM algorithm. Dempster et al. (1977) showed that, under certain regularity conditions, iteration of the expectation and maximization steps will converge to a local maximum. The EM

¹⁰ The MAP configuration $\tilde{\theta}_s$ depends on the coordinate system in which the parameter variables are expressed. The expression for the MAP configuration given here is obtained by the following procedure. First, we transform each variable set $\theta_{ij} = (\theta_{ij2}, \dots, \theta_{ijr_i})$ to the new coordinate system $\phi_{ij} = (\phi_{ij2}, \dots, \phi_{ijr_i})$, where $\phi_{ijk} = \log(\theta_{ijk}/\theta_{ij1})$, $k = 2, \dots, r_i$. This coordinate system, which we denote by ϕ_s , is sometimes referred to as the *canonical* coordinate system for the multinomial distribution (see, e.g., Bernardo and Smith, 1994, pp. 199–202). Next, we determine the configuration of ϕ_s that maximizes $p(\phi_s | D_c, S^h)$. Finally, we transform this MAP configuration to the original coordinate system. Using the MAP configuration corresponding to the coordinate system ϕ_s has several advantages, which are discussed in Thiesson (1995b) and MacKay (1996).

algorithm is typically applied when sufficient statistics exist (i.e., when local distribution functions are in the exponential family), although generalizations of the EM algorithm have been used for more complicated local distributions (see, e.g., Saul et al. 1996).

3.7 Learning Parameters and Structure

Now we consider the problem of learning about both the structure and probabilities of a Bayesian network given data.

Assuming we think structure can be improved, we must be uncertain about the network structure that encodes the physical joint probability distribution for \mathbf{X} . Following the Bayesian approach, we encode this uncertainty by defining a (discrete) variable whose states correspond to the possible network-structure hypotheses S^h , and assessing the probabilities $p(S^h)$. Then, given a random sample D from the physical probability distribution for \mathbf{X} , we compute the posterior distribution $p(S^h|D)$ and the posterior distributions $p(\theta_s|D, S^h)$, and use these distributions in turn to compute expectations of interest. For example, to predict the next case after seeing D , we compute

$$p(\mathbf{x}_{N+1}|D) = \sum_{S^h} p(S^h|D) \int p(\mathbf{x}_{N+1}|\theta_s, S^h) p(\theta_s|D, S^h) d\theta_s \quad (3.33)$$

In performing the sum, we assume that the network-structure hypotheses are mutually exclusive. We return to this point in Section 3.9.

The computation of $p(\theta_s|D, S^h)$ is as we have described in the previous two sections. The computation of $p(S^h|D)$ is also straightforward, at least in principle. From Bayes' theorem, we have

$$p(S^h|D) = p(S^h) p(D|S^h)/p(D) \quad (3.34)$$

where $p(D)$ is a normalization constant that does not depend upon structure. Thus, to determine the posterior distribution for network structures, we need to compute the marginal likelihood of the data ($p(D|S^h)$) for each possible structure.

We discuss the computation of marginal likelihoods in detail in Section 3.9. As an introduction, consider our example with unrestricted multinomial distributions, parameter independence, Dirichlet priors, and complete data. As we have discussed, when there are no missing data, each parameter vector θ_{ij} is updated independently. In effect, we have a separate multi-sided thumbtack problem for every i and j . Consequently, the marginal likelihood of the data is just the product of the marginal likelihoods for each i - j pair (given by Equation 3.13):

$$p(D|S^h) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \cdot \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})} \quad (3.35)$$

This formula was first derived by Cooper and Herskovits (1992).

Unfortunately, the full Bayesian approach that we have described is often impractical. One important computation bottleneck is produced by the average over models in Equation 3.33. If we consider Bayesian-network models with n variables, the number of possible structure hypotheses is more than exponential in n . Consequently, in situations where the user can not exclude almost all of these hypotheses, the approach is intractable.

Statisticians, who have been confronted by this problem for decades in the context of other types of models, use two approaches to address this problem: *model selection* and *selective model averaging*. The former approach is to select a “good” model (i.e., structure hypothesis) from among all possible models, and use it as if it were the correct model. The latter approach is to select a manageable number of good models from among all possible models and pretend that these models are exhaustive. These related approaches raise several important questions. In particular, do these approaches yield accurate results when applied to Bayesian-network structures? If so, how do we search for good models? And how do we decide whether or not a model is “good”?

The question of accuracy is difficult to answer in theory. Nonetheless, several researchers have shown experimentally that the selection of a single good hypothesis often yields accurate predictions (Cooper and Herskovits 1992; Aliferis and Cooper 1994; Heckerman et al., 1995b) and that model averaging using Monte-Carlo methods can sometimes be efficient and yield even better predictions (Madigan et al., 1996). These results are somewhat surprising, and are largely responsible for the great deal of recent interest in learning with Bayesian networks. In Sections 3.8 through 3.10, we consider different definitions of what is means for a model to be “good”, and discuss the computations entailed by some of these definitions. In Section 3.11, we discuss model search.

We note that model averaging and model selection lead to models that generalize well to *new* data. That is, these techniques help us to avoid the overfitting of data. As is suggested by Equation 3.33, Bayesian methods for model averaging and model selection are efficient in the sense that all cases in D can be used to both smooth and train the model. As we shall see in the following two sections, this advantage holds true for the Bayesian approach in general.

3.8 Criteria for Model Selection

Most of the literature on learning with Bayesian networks is concerned with model selection. In these approaches, some *criterion* is used to measure the degree to which a network structure (equivalence class) fits the prior knowledge and data. A search algorithm is then used to find an equivalence class that receives a high score by this criterion. Selective model averaging is more complex, because it is often advantageous to identify network structures that are significantly different. In many cases, a single criterion is unlikely to identify such complementary network structures. In this section, we discuss criteria for the simpler problem of model selection. For a discussion of selective model averaging, see Madigan and Raftery (1994).

3.8.1 Relative Posterior Probability

A criterion that is often used for model selection is the log of the relative posterior probability $\log p(D, S^h) = \log p(S^h) + \log p(D|S^h)$.¹¹ The logarithm is used for numerical convenience. This criterion has two components: the log prior and the log marginal likelihood. In Section 3.9, we examine the computation of the log marginal likelihood. In Section 3.10.2, we discuss the assessment of network-structure priors. Note that our comments about these terms are also relevant to the full Bayesian approach.

The log marginal likelihood has the following interesting interpretation described by Dawid (1984). From the chain rule of probability, we have

$$\log p(D|S^h) = \sum_{l=1}^N \log p(\mathbf{x}_l | \mathbf{x}_1, \dots, \mathbf{x}_{l-1}, S^h) \quad (3.36)$$

The term $p(\mathbf{x}_l | \mathbf{x}_1, \dots, \mathbf{x}_{l-1}, S^h)$ is the prediction for \mathbf{x}_l made by model S^h after averaging over its parameters. The log of this term can be thought of as the utility or reward for this prediction under the utility function $\log p(\mathbf{x})$.¹² Thus, a model with the highest log marginal likelihood (or the highest posterior probability, assuming equal priors on structure) is also a model that is the best sequential predictor of the data D under the log utility function.

Dawid (1984) also notes the relationship between this criterion and cross validation. When using one form of cross validation, known as *leave-one-out* cross validation, we first train a model on all but one of the cases in the random sample—say, $V_l = \{\mathbf{x}_1, \dots, \mathbf{x}_{l-1}, \mathbf{x}_{l+1}, \dots, \mathbf{x}_N\}$. Then, we predict the omitted case, and reward this prediction under some utility function. Finally, we repeat this procedure for every case in the random sample, and sum the rewards for each prediction. If the prediction is probabilistic and the utility function is $\log p(\mathbf{x})$, we obtain the cross-validation criterion

$$CV(S^h, D) = \sum_{l=1}^N \log p(\mathbf{x}_l | V_l, S^h) \quad (3.37)$$

which is similar to Equation 3.36. One problem with this criterion is that training and test cases are interchanged. For example, when we compute $p(\mathbf{x}_1 | V_1, S^h)$ in Equation 3.37, we use \mathbf{x}_2 for training and \mathbf{x}_1 for testing. Whereas, when we compute $p(\mathbf{x}_2 | V_2, S^h)$, we use \mathbf{x}_1 for training and \mathbf{x}_2 for testing. Such interchanges can lead to the selection of a model that over fits the data (Dawid, 1984). Various approaches for attenuating this problem have been described, but we see from Equation 3.36 that the log-marginal-likelihood criterion avoids the problem

¹¹ An equivalent criterion that is often used is $\log(p(S^h|D)/p(S_0^h|D)) = \log(p(S^h)/p(S_0^h)) + \log(p(D|S^h)/p(D|S_0^h))$. The ratio $p(D|S^h)/p(D|S_0^h)$ is known as a *Bayes' factor*.

¹² This utility function is known as a *proper scoring rule*, because its use encourages people to assess their true probabilities. For a characterization of proper scoring rules and this rule in particular, see Bernardo (1979).

altogether. Namely, when using this criterion, we never interchange training and test cases.

3.8.2 Local Criteria

Consider the problem of diagnosing an ailment given the observation of a set of findings. Suppose that the set of ailments under consideration are mutually exclusive and collectively exhaustive, so that we may represent these ailments using a single variable A . A possible Bayesian network for this classification problem is shown in Figure 3.5.

The posterior-probability criterion is *global* in the sense that it is equally sensitive to all possible dependencies. In the diagnosis problem, the posterior-probability criterion is just as sensitive to dependencies among the finding variables as it is to dependencies between ailment and findings. Assuming that we observe all (or perhaps all but a few) of the findings in D , a more reasonable criterion would be *local* in the sense that it ignores dependencies among findings and is sensitive only to the dependencies among the ailment and findings. This observation applies to all classification and regression problems with complete data.

One such local criterion, suggested by Spiegelhalter et al. (1993), is a variation on the sequential log-marginal-likelihood criterion:

$$LC(S^h, D) = \sum_{l=1}^N \log p(a_l | \mathbf{F}_l, D_l, S^h) \quad (3.38)$$

where a_l and \mathbf{F}_l denote the observation of the ailment A and findings \mathbf{F} in the l th case, respectively. In other words, to compute the l th term in the product, we train our model S with the first $l - 1$ cases, and then determine how well it predicts the ailment given the findings in the l th case. We can view this criterion, like the log-marginal-likelihood, as a form of cross validation where training and test cases are never interchanged.

The log utility function has interesting theoretical properties, but it is sometimes inaccurate for real-world problems. In general, an appropriate reward or utility function will depend on the decision-making problem or problems to which

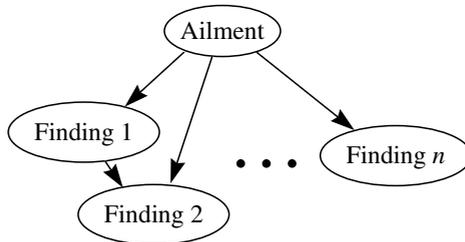


Fig. 3.5. A Bayesian-network structure for medical diagnosis

the probabilistic models are applied. Howard and Matheson (1983) have collected a series of articles describing how to construct utility models for specific decision problems. Once we construct such utility models, we can use suitably modified forms of Equation 3.38 for model selection.

3.9 Computation of the Marginal Likelihood

As mentioned, an often-used criterion for model selection is the log relative posterior probability $\log p(D, S^h) = \log p(S^h) + \log p(D|S^h)$. In this section, we discuss the computation of the second component of this criterion: the log marginal likelihood.

Given (1) local distribution functions in the exponential family, (2) mutual independence of the parameters θ_i , (3) conjugate priors for these parameters, and (4) complete data, the log marginal likelihood can be computed efficiently and in closed form. Equation 3.35 is an example for unrestricted multinomial distributions. Buntine (1994) and Heckerman and Geiger (1996) discuss the computation for other local distribution functions. Here, we concentrate on approximations for incomplete data.

The Monte-Carlo and Gaussian approximations for learning about parameters that we discussed in Section 3.6 are also useful for computing the marginal likelihood given incomplete data. One Monte-Carlo approach, described by Chib (1995) and Raftery (1996), uses Bayes' theorem:

$$p(D|S^h) = \frac{p(\theta_s|S^h) p(D|\theta_s, S^h)}{p(\theta_s|D, S^h)} \quad (3.39)$$

For any configuration of θ_s , the prior term in the numerator can be evaluated directly. In addition, the likelihood term in the numerator can be computed using Bayesian-network inference. Finally, the posterior term in the denominator can be computed using Gibbs sampling, as we described in Section 3.6.1. Other, more sophisticated Monte-Carlo methods are described by DiCiccio et al. (1995).

As we have discussed, Monte-Carlo methods are accurate but computationally inefficient, especially for large databases. In contrast, methods based on the Gaussian approximation are more efficient, and can be as accurate as Monte-Carlo methods on large data sets.

Recall that, for large amounts of data, $p(D|\theta_s, S^h) \cdot p(\theta_s|S^h)$ can often be approximated as a multivariate-Gaussian distribution. Consequently,

$$p(D|S^h) = \int p(D|\theta_s, S^h) p(\theta_s|S^h) d\theta_s \quad (3.40)$$

can be evaluated in closed form. In particular, substituting Equation 3.31 into Equation 3.40, integrating, and taking the logarithm of the result, we obtain the approximation:

$$\log p(D|S^h) \approx \log p(D|\tilde{\theta}_s, S^h) + \log p(\tilde{\theta}_s|S^h) + \frac{d}{2} \log(2\pi) - \frac{1}{2} \log |A| \quad (3.41)$$

where d is the dimension of $g(\boldsymbol{\theta}_s)$. For a Bayesian network with unrestricted multinomial distributions, this dimension is typically given by $\sum_{i=1}^n q_i (r_i - 1)$. Sometimes, when there are hidden variables, this dimension is lower. See Geiger et al. (1996) for a discussion of this point.

This approximation technique for integration is known as *Laplace's method*, and we refer to Equation 3.41 as the *Laplace approximation*. Kass et al. (1988) have shown that, under certain regularity conditions, relative errors in this approximation are $O(1/N)$, where N is the number of cases in D . Thus, the Laplace approximation can be extremely accurate. For more detailed discussions of this approximation, see—for example—Kass et al. (1988) and Kass and Raftery (1995).

Although Laplace's approximation is efficient relative to Monte-Carlo approaches, the computation of $|A|$ is nevertheless intensive for large-dimension models. One simplification is to approximate $|A|$ using only the diagonal elements of the Hessian A . Although in so doing, we incorrectly impose independencies among the parameters, researchers have shown that the approximation can be accurate in some circumstances (see, e.g., Becker and Le Cun, 1989, and Chickering and Heckerman, 1996). Another efficient variant of Laplace's approximation is described by Cheeseman and Stutz (1995), who use the approximation in the AutoClass program for data clustering (see also Chickering and Heckerman, 1996.)

We obtain a very efficient (but less accurate) approximation by retaining only those terms in Equation 3.41 that increase with N : $\log p(D|\tilde{\boldsymbol{\theta}}_s, S^h)$, which increases linearly with N , and $\log |A|$, which increases as $d \log N$. Also, for large N , $\tilde{\boldsymbol{\theta}}_s$ can be approximated by the ML configuration of $\boldsymbol{\theta}_s$. Thus, we obtain

$$\log p(D|S^h) \approx \log p(D|\hat{\boldsymbol{\theta}}_s, S^h) - \frac{d}{2} \log N \quad (3.42)$$

This approximation is called the *Bayesian information criterion* (BIC), and was first derived by Schwarz (1978).

The BIC approximation is interesting in several respects. First, it does not depend on the prior. Consequently, we can use the approximation without assessing a prior.¹³ Second, the approximation is quite intuitive. Namely, it contains a term measuring how well the parameterized model predicts the data ($\log p(D|\hat{\boldsymbol{\theta}}_s, S^h)$) and a term that punishes the complexity of the model ($d/2 \log N$). Third, the BIC approximation is exactly minus the Minimum Description Length (MDL) criterion described by Rissanen (1987). Thus, recalling the discussion in Section 3.9, we see that the marginal likelihood provides a connection between cross validation and MDL.

3.10 Priors

To compute the relative posterior probability of a network structure, we must assess the structure prior $p(S^h)$ and the parameter priors $p(\boldsymbol{\theta}_s|S^h)$ (unless we

¹³ One of the technical assumptions used to derive this approximation is that the prior is non-zero around $\hat{\boldsymbol{\theta}}_s$.

are using large-sample approximations such as BIC/MDL). The parameter priors $p(\theta_s | S^h)$ are also required for the alternative scoring functions discussed in Section 3.8. Unfortunately, when many network structures are possible, these assessments will be intractable. Nonetheless, under certain assumptions, we can derive the structure and parameter priors for many network structures from a manageable number of direct assessments. Several authors have discussed such assumptions and corresponding methods for deriving priors (Cooper and Herskovits, 1991, 1992; Buntine, 1991; Spiegelhalter et al., 1993; Heckerman et al., 1995b; Heckerman and Geiger, 1996). In this section, we examine some of these approaches.

3.10.1 Priors on Network Parameters

First, let us consider the assessment of priors for the model parameters. We consider the approach of Heckerman et al. (1995b) who address the case where the local distribution functions are unrestricted multinomial distributions and the assumption of parameter independence holds.

Their approach is based on two key concepts: independence equivalence and distribution equivalence. We say that two Bayesian-network structures for \mathbf{X} are *independence equivalent* if they represent the same set of conditional-independence assertions for \mathbf{X} (Verma and Pearl, 1990). For example, given $\mathbf{X} = \{X, Y, Z\}$, the network structures $X \rightarrow Y \rightarrow Z$, $X \leftarrow Y \rightarrow Z$, and $X \leftarrow Y \leftarrow Z$ represent only the independence assertion that X and Z are conditionally independent given Y . Consequently, these network structures are equivalent. As another example, a *complete network structure* is one that has no missing edge—that is, it encodes no assertion of conditional independence. When \mathbf{X} contains n variables, there are $n!$ possible complete network structures: one network structure for each possible ordering of the variables. All complete network structures for $p(\mathbf{x})$ are independence equivalent. In general, two network structures are independence equivalent if and only if they have the same structure ignoring arc directions and the same v -structures (Verma and Pearl, 1990). A *v-structure* is an ordered tuple (X, Y, Z) such that there is an arc from X to Y and from Z to Y , but no arc between X and Z .

The concept of distribution equivalence is closely related to that of independence equivalence. Suppose that all Bayesian networks for \mathbf{X} under consideration have local distribution functions in the family \mathcal{F} . This is not a restriction, per se, because \mathcal{F} can be a large family. We say that two Bayesian-network structures S_1 and S_2 for \mathbf{X} are *distribution equivalent with respect to (wrt) \mathcal{F}* if they represent the same joint probability distributions for \mathbf{X} —that is, if, for every θ_{s_1} , there exists a θ_{s_2} such that $p(\mathbf{x} | \theta_{s_1}, S_1^h) = p(\mathbf{x} | \theta_{s_2}, S_2^h)$, and vice versa.

Distribution equivalence wrt some \mathcal{F} implies independence equivalence, but the converse does not hold. For example, when \mathcal{F} is the family of generalized linear-regression models, the complete network structures for $n \geq 3$ variables do not represent the same sets of distributions. Nonetheless, there are families \mathcal{F} —for example, unrestricted multinomial distributions and linear-regression

models with Gaussian noise—where independence equivalence implies distribution equivalence wrt \mathcal{F} (Heckerman and Geiger, 1996).

The notion of distribution equivalence is important, because if two network structures S_1 and S_2 are distribution equivalent wrt to a given \mathcal{F} , then the hypotheses associated with these two structures are identical—that is, $S_1^h = S_2^h$. Thus, for example, if S_1 and S_2 are distribution equivalent, then their probabilities must be equal in any state of information. Heckerman et al. (1995b) call this property *hypothesis equivalence*.

In light of this property, we should associate each hypothesis with an equivalence class of structures rather than a single network structure, and our methods for learning network structure should actually be interpreted as methods for learning equivalence classes of network structures (although, for the sake of brevity, we often blur this distinction). Thus, for example, the sum over network-structure hypotheses in Equation 3.33 should be replaced with a sum over equivalence-class hypotheses. An efficient algorithm for identifying the equivalence class of a given network structure can be found in Chickering (1995).

We note that hypothesis equivalence holds provided we interpret Bayesian-network structure simply as a representation of conditional independence. Nonetheless, stronger definitions of Bayesian networks exist where arcs have a causal interpretation (see Section 3.15). Heckerman et al. (1995b) and Heckerman (1995) argue that, although it is unreasonable to assume hypothesis equivalence when working with causal Bayesian networks, it is often reasonable to adopt a weaker assumption of *likelihood equivalence*, which says that the observations in a database can not help to discriminate two equivalent network structures.

Now let us return to the main issue of this section: the derivation of priors from a manageable number of assessments. Geiger and Heckerman (1995) show that the assumptions of parameter independence and likelihood equivalence imply that the parameters for any *complete* network structure S_c must have a Dirichlet distribution with constraints on the hyperparameters given by

$$\alpha_{ijk} = \alpha p(x_i^k, \mathbf{pa}_i^j | S_c^h) \quad (3.43)$$

where α is the user's equivalent sample size,¹⁴ and $p(x_i^k, \mathbf{pa}_i^j | S_c^h)$ is computed from the user's joint probability distribution $p(\mathbf{x} | S_c^h)$. This result is rather remarkable, as the two assumptions leading to the constrained Dirichlet solution are qualitative.

To determine the priors for parameters of *incomplete* network structures, Heckerman et al. (1995b) use the assumption of *parameter modularity*, which says that if X_i has the same parents in network structures S_1 and S_2 , then

$$p(\boldsymbol{\theta}_{ij} | S_1^h) = p(\boldsymbol{\theta}_{ij} | S_2^h)$$

for $j = 1, \dots, q_i$. They call this property parameter modularity, because it says that the distributions for parameters $\boldsymbol{\theta}_{ij}$ depend only on the structure of the network that is local to variable X_i —namely, X_i and its parents.

¹⁴ Recall the method of equivalent samples for assessing beta and Dirichlet distributions discussed in Section 3.2.

Given the assumptions of parameter modularity and parameter independence,¹⁵ it is a simple matter to construct priors for the parameters of an arbitrary network structure given the priors on complete network structures. In particular, given parameter independence, we construct the priors for the parameters of each node separately. Furthermore, if node X_i has parents \mathbf{Pa}_i in the given network structure, we identify a complete network structure where X_i has these parents, and use Equation 3.43 and parameter modularity to determine the priors for this node. The result is that all terms α_{ijk} for all network structures are determined by Equation 3.43. Thus, from the assessments α and $p(\mathbf{x}|S_c^h)$, we can derive the parameter priors for all possible network structures. Combining Equation 3.43 with Equation 3.35, we obtain a model-selection criterion that assigns equal marginal likelihoods to independence equivalent network structures.

We can assess $p(\mathbf{x}|S_c^h)$ by constructing a Bayesian network, called a *prior network*, that encodes this joint distribution. Heckerman et al. (1995b) discuss the construction of this network.

3.10.2 Priors on Structures

Now, let us consider the assessment of priors on network-structure hypotheses. Note that the alternative criteria described in Section 3.8 can incorporate prior biases on network-structure hypotheses. Methods similar to those discussed in this section can be used to assess such biases.

The simplest approach for assigning priors to network-structure hypotheses is to assume that every hypothesis is equally likely. Of course, this assumption is typically inaccurate and used only for the sake of convenience. A simple refinement of this approach is to ask the user to exclude various hypotheses (perhaps based on judgments of of cause and effect), and then impose a uniform prior on the remaining hypotheses. We illustrate this approach in Section 3.12.

Buntine (1991) describes a set of assumptions that leads to a richer yet efficient approach for assigning priors. The first assumption is that the variables can be ordered (e.g., through a knowledge of time precedence). The second assumption is that the presence or absence of possible arcs are mutually independent. Given these assumptions, $n(n-1)/2$ probability assessments (one for each possible arc in an ordering) determines the prior probability of every possible network-structure hypothesis. One extension to this approach is to allow for multiple possible orderings. One simplification is to assume that the probability that an arc is absent or present is independent of the specific arc in question. In this case, only one probability assessment is required.

An alternative approach, described by Heckerman et al. (1995b) uses a prior network. The basic idea is to penalize the prior probability of any structure according to some measure of deviation between that structure and the prior network. Heckerman et al. (1995b) suggest one reasonable measure of deviation.

Madigan et al. (1995) give yet another approach that makes use of imaginary data from a domain expert. In their approach, a computer program helps the user

¹⁵ This construction procedure also assumes that every structure has a non-zero prior probability.

create a hypothetical set of complete data. Then, using techniques such as those in Section 3.7, they compute the posterior probabilities of network-structure hypotheses given this data, assuming the prior probabilities of hypotheses are uniform. Finally, they use these posterior probabilities as priors for the analysis of the real data.

3.11 Search Methods

In this section, we examine search methods for identifying network structures with high scores by some criterion. Consider the problem of finding the best network from the set of all networks in which each node has no more than k parents. Unfortunately, the problem for $k > 1$ is NP-hard even when we use the restrictive prior given by Equation 3.43 (Chickering et al. 1995). Thus, researchers have used heuristic search algorithms, including greedy search, greedy search with restarts, best-first search, and Monte-Carlo methods.

One consolation is that these search methods can be made more efficient when the model-selection criterion is separable. Given a network structure for domain \mathbf{X} , we say that a criterion for that structure is *separable* if it can be written as a product of variable-specific criteria:

$$C(S^h, D) = \prod_{i=1}^n c(X_i, \mathbf{Pa}_i, D_i) \quad (3.44)$$

where D_i is the data restricted to the variables X_i and \mathbf{Pa}_i . An example of a separable criterion is the BD criterion (Equations 3.34 and 3.35) used in conjunction with any of the methods for assessing structure priors described in Section 3.10.

Most of the commonly used search methods for Bayesian networks make successive arc changes to the network, and employ the property of separability to evaluate the merit of each change. The possible changes that can be made are easy to identify. For any pair of variables, if there is an arc connecting them, then this arc can either be reversed or removed. If there is no arc connecting them, then an arc can be added in either direction. All changes are subject to the constraint that the resulting network contains no directed cycles. We use E to denote the set of eligible changes to a graph, and $\Delta(e)$ to denote the change in log score of the network resulting from the modification $e \in E$. Given a separable criterion, if an arc to X_i is added or deleted, only $c(X_i, \mathbf{Pa}_i, D_i)$ need be evaluated to determine $\Delta(e)$. If an arc between X_i and X_j is reversed, then only $c(X_i, \mathbf{Pa}_i, D_i)$ and $c(X_j, \mathbf{Pa}_j, D_j)$ need be evaluated.

One simple heuristic search algorithm is greedy search. First, we choose a network structure. Then, we evaluate $\Delta(e)$ for all $e \in E$, and make the change e for which $\Delta(e)$ is a maximum, provided it is positive. We terminate search when there is no e with a positive value for $\Delta(e)$. When the criterion is separable, we can avoid recomputing all terms $\Delta(e)$ after every change. In particular, if neither X_i , X_j , nor their parents are changed, then $\Delta(e)$ remains unchanged for all changes e involving these nodes as long as the resulting network is acyclic.

Candidates for the initial graph include the empty graph, a random graph, and a prior network.

A potential problem with any local-search method is getting stuck at a local maximum. One method for escaping local maxima is greedy search with random restarts. In this approach, we apply greedy search until we hit a local maximum. Then, we randomly perturb the network structure, and repeat the process for some manageable number of iterations.

Another method for escaping local maxima is simulated annealing. In this approach, we initialize the system at some temperature T_0 . Then, we pick some eligible change e at random, and evaluate the expression $p = \exp(\Delta(e)/T_0)$. If $p > 1$, then we make the change e ; otherwise, we make the change with probability p . We repeat this selection and evaluation process α times or until we make β changes. If we make no changes in α repetitions, then we stop searching. Otherwise, we lower the temperature by multiplying the current temperature T_0 by a decay factor $0 < \gamma < 1$, and continue the search process. We stop searching if we have lowered the temperature more than δ times. Thus, this algorithm is controlled by five parameters: $T_0, \alpha, \beta, \gamma$ and δ . To initialize this algorithm, we can start with the empty graph, and make T_0 large enough so that almost every eligible change is made, thus creating a random graph. Alternatively, we may start with a lower temperature, and use one of the initialization methods described for local search.

Another method for escaping local maxima is best-first search (e.g., Korf, 1993). In this approach, the space of all network structures is searched systematically using a heuristic measure that determines the next best structure to examine. Chickering (1996b) has shown that, for a fixed amount of computation time, greedy search with random restarts produces better models than does best-first search.

One important consideration for any search algorithm is the search space. The methods that we have described search through the space of Bayesian-network structures. Nonetheless, when the assumption of hypothesis equivalence holds, one can search through the space of network-structure equivalence classes. One benefit of the latter approach is that the search space is smaller. One drawback of the latter approach is that it takes longer to move from one element in the search space to another. Work by Spirtes and Meek (1995) and Chickering (1996) confirm these observations experimentally. Unfortunately, no comparisons are yet available that determine whether the benefits of equivalence-class search outweigh the costs.

3.12 A Simple Example

Before we move on to other issues, let us step back and look at our overall approach. In a nutshell, we can construct both structure and parameter priors by constructing a Bayesian network (the prior network) along with additional assessments such as an equivalent sample size and causal constraints. We then use either Bayesian model selection, selective model averaging, or full model

Table 3.1. An imagined database for the fraud problem

Case	Fraud	Gas	Jewelry	Age	Sex
1	no	no	no	30-50	female
2	no	no	no	30-50	male
3	yes	yes	yes	>50	male
4	no	no	no	30-50	male
5	no	yes	no	<30	female
6	no	no	no	<30	female
7	no	no	no	>50	male
8	no	no	yes	30-50	female
9	no	yes	no	<30	male
10	no	no	no	<30	female

averaging to obtain one or more networks for prediction and/or explanation. In effect, we have a procedure for using data to improve the structure and probabilities of an initial Bayesian network.

Here, we present two artificial examples to illustrate this process. Consider again the problem of fraud detection from Section 3.3. Suppose we are given the database D in Table 3.1, and we want to predict the next case—that is, compute $p(\mathbf{x}_{N+1}|D)$. Let us assert that only two network-structure hypotheses have appreciable probability: the hypothesis corresponding to the network structure in Figure 3.3 (S_1), and the hypothesis corresponding to the same structure with an arc added from *Age* to *Gas* (S_2). Furthermore, let us assert that these two hypotheses are equally likely—that is, $p(S_1^h) = p(S_2^h) = 0.5$. In addition, let us use the parameter priors given by Equation 3.43, where $\alpha = 10$ and $p(\mathbf{x}|S_c^h)$ is given by the prior network in Figure 3.3. Using Equations 3.34 and 3.35, we obtain $p(S_1^h|D) = 0.26$ and $p(S_2^h|D) = 0.74$. Because we have only two models to consider, we can model average according to Equation 3.33:

$$p(\mathbf{x}_{N+1}|D) = 0.26 p(\mathbf{x}_{N+1}|D, S_1^h) + 0.74 p(\mathbf{x}_{N+1}|D, S_2^h)$$

where $p(\mathbf{x}_{N+1}|D, S^h)$ is given by Equation 3.27. (We don't display these probability distributions.) If we had to choose one model, we would choose S_2 , assuming the posterior-probability criterion is appropriate. Note that the data favors the presence of the arc from *Age* to *Gas* by a factor of three. This is not surprising, because in the two cases in the database where fraud is absent and gas was purchased recently, the card holder was less than 30 years old.

An application of model selection, described by Spirtes and Meek (1995), is illustrated in Figure 3.6. Figure 3.6a is a hand-constructed Bayesian network for the domain of ICU ventilator management, called the Alarm network (Beinlich et al.,

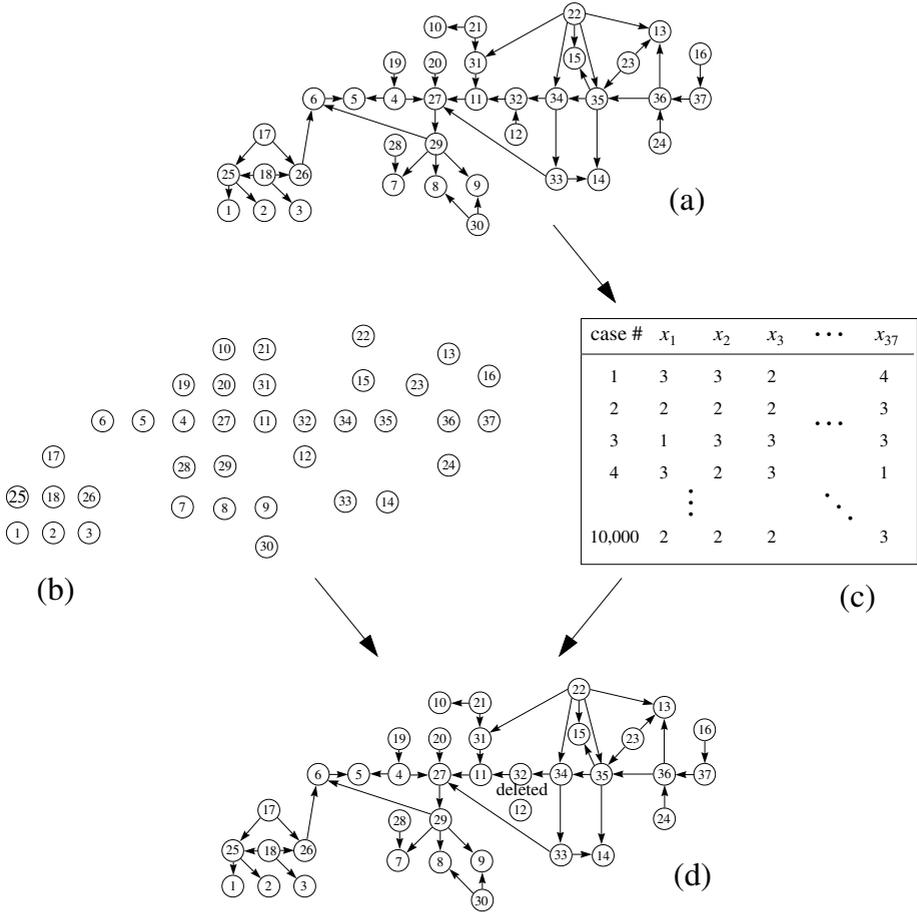


Fig. 3.6. (a) The Alarm network structure. (b) A prior network encoding a user's beliefs about the Alarm domain. (c) A random sample of size 10,000 generated from the Alarm network. (d) The network learned from the prior network and the random sample. The only difference between the learned and true structure is an arc deletion as noted in (d). Network probabilities are not shown.

1989). Figure 3.6c is a random sample from the Alarm network of size 10,000. Figure 3.6b is a simple prior network for the domain. This network encodes mutual independence among the variables, and (not shown) uniform probability distributions for each variable.

Figure 3.6d shows the most likely network structure found by a two-pass greedy search in equivalence-class space. In the first pass, arcs were added until the model score did not improve. In the second pass, arcs were deleted until the model score did not improve. Structure priors were uniform; and parameter priors were computed from the prior network using Equation 3.43 with $\alpha = 10$.

The network structure learned from this procedure differs from the true network structure only by a single arc deletion. In effect, we have used the data to improve dramatically the original model of the user.

3.13 Bayesian Networks for Supervised Learning

As we discussed in Section 3.5, the local distribution functions $p(x_i|\mathbf{pa}_i, \boldsymbol{\theta}_i, S^h)$ are essentially classification/regression models. Therefore, if we are doing supervised learning where the explanatory (input) variables cause the outcome (target) variable and data is complete, then the Bayesian-network and classification/regression approaches are identical.

When data is complete but input/target variables do not have a simple cause/effect relationship, tradeoffs emerge between the Bayesian-network approach and other methods. For example, consider the classification problem in Figure 3.5. Here, the Bayesian network encodes dependencies between findings and ailments as well as among the findings, whereas another classification model such as a decision tree encodes only the relationships between findings and ailment. Thus, the decision tree may produce more accurate classifications, because it can encode the necessary relationships with fewer parameters. Nonetheless, the use of local criteria for Bayesian-network model selection mitigates this advantage. Furthermore, the Bayesian network provides a more natural representation in which to encode prior knowledge, thus giving this model a possible advantage for sufficiently small sample sizes. Another argument, based on bias–variance analysis, suggests that neither approach will dramatically outperform the other (Friedman, 1996).

Singh and Provan (1995) compare the classification accuracy of Bayesian networks and decision trees using complete data sets from the University of California, Irvine Repository of Machine Learning databases. Specifically, they compare C4.5 with an algorithm that learns the structure and probabilities of a Bayesian network using a variation of the Bayesian methods we have described. The latter algorithm includes a model-selection phase that discards some input variables. They show that, overall, Bayesian networks and decisions trees have about the same classification error. These results support the argument of Friedman (1996).

When the input variables cause the target variable and data is incomplete, the dependencies between input variables becomes important, as we discussed in the introduction. Bayesian networks provide a natural framework for learning about and encoding these dependencies. Unfortunately, no studies have been done comparing these approaches with other methods for handling missing data.

3.14 Bayesian Networks for Unsupervised Learning

The techniques described in this paper can be used for unsupervised learning. A simple example is the AutoClass program of Cheeseman and Stutz (1995), which performs data clustering. The idea behind AutoClass is that there is a single hidden (i.e., never observed) variable that causes the observations. This

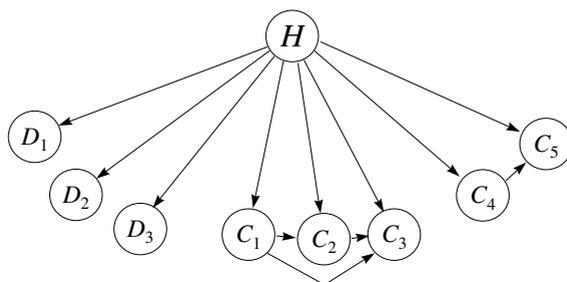


Fig. 3.7. A Bayesian-network structure for AutoClass. The variable H is hidden. Its possible states correspond to the underlying classes in the data.

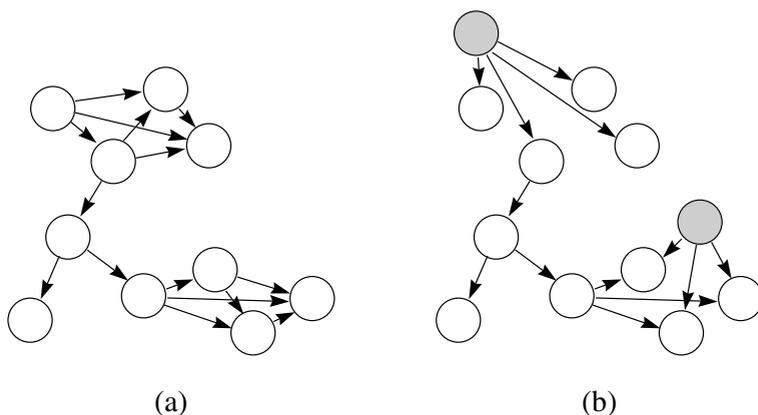


Fig. 3.8. (a) A Bayesian-network structure for observed variables. (b) A Bayesian-network structure with hidden variables (shaded) suggested by the network structure in (a).

hidden variable is discrete, and its possible states correspond to the underlying classes in the data. Thus, AutoClass can be described by a Bayesian network such as the one in Figure 3.7. For reasons of computational efficiency, Cheeseman and Stutz (1995) assume that the discrete variables (e.g., D_1, D_2, D_3 in the figure) and user-defined sets of continuous variables (e.g., $\{C_1, C_2, C_3\}$ and $\{C_4, C_5\}$) are mutually independent given H . Given a data set D , AutoClass searches over variants of this model (including the number of states of the hidden variable) and selects a variant whose (approximate) posterior probability is a local maximum.

AutoClass is an example where the user presupposes the existence of a hidden variable. In other situations, we may be unsure about the presence of a hidden variable. In such cases, we can score models with and without hidden variables to reduce our uncertainty. We illustrate this approach on a real-world case study in Section 3.16. Alternatively, we may have little idea about what hidden variables to model. The search algorithms of Spirtes et al. (1993) provide one method

for identifying possible hidden variables in such situations. Martin and VanLehn (1995) suggest another method.

Their approach is based on the observation that if a set of variables are mutually dependent, then a simple explanation is that these variables have a single hidden common cause rendering them mutually independent. Thus, to identify possible hidden variables, we first apply some learning technique to select a model containing no hidden variables. Then, we look for sets of mutually dependent variables in this learned model. For each such set of variables (and combinations thereof), we create a new model containing a hidden variable that renders that set of variables conditionally independent. We then score the new models, possibly finding one better than the original. For example, the model in Figure 3.8a has two sets of mutually dependent variables. Figure 3.8b shows another model containing hidden variables suggested by this model.

3.15 Learning Causal Relationships

As we have mentioned, the causal semantics of a Bayesian network provide a means by which we can learn causal relationships. In this section, we examine these semantics, and provide a basic discussion on how causal relationships can be learned. We note that these methods are new and controversial. For critical discussions on both sides of the issue, see Spirtes et al. (1993), Pearl (1995), and Humphreys and Freedman (1995).

For purposes of illustration, suppose we are marketing analysts who want to know whether or not we should increase, decrease, or leave alone the exposure of a particular advertisement in order to maximize our profit from the sales of a product. Let variables *Ad* (A) and *Buy* (B) represent whether or not an individual has seen the advertisement and has purchased the product, respectively. In one component of our analysis, we would like to learn the physical probability that $B = \textit{true}$ given that we *force* A to be true, and the physical probability that $B = \textit{true}$ given that we force A to be false.¹⁶ We denote these probabilities $p(b|\hat{a})$ and $p(b|\bar{\hat{a}})$, respectively. One method that we can use to learn these probabilities is to perform a randomized experiment: select two similar populations at random, force A to be true in one population and false in the other, and observe B . This method is conceptually simple, but it may be difficult or expensive to find two similar populations that are suitable for the study.

An alternative method follows from causal knowledge. In particular, suppose A causes B . Then, whether we force A to be true or simply observe that A is true in the current population, the advertisement should have the same causal influence on the individual's purchase. Consequently, $p(b|\hat{a}) = p(b|a)$, where $p(b|a)$ is the physical probability that $B = \textit{true}$ given that we observe $A = \textit{true}$ in the current population. Similarly, $p(b|\bar{\hat{a}}) = p(b|\bar{a})$. In contrast, if B causes A , forcing A to some state should not influence B at all. Therefore, we have $p(b|\hat{a}) = p(b|\bar{\hat{a}}) = p(b)$. In general, knowledge that X causes Y allows us to

¹⁶ It is important that these interventions do not interfere with the normal effect of A on B . See Heckerman and Shachter (1995) for a discussion of this point.

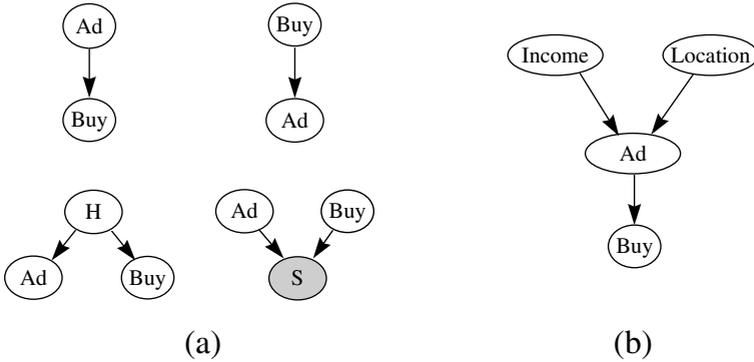


Fig. 3.9. (a) Causal graphs showing for explanations for an observed dependence between *Ad* and *Buy*. The node *H* corresponds to a hidden common cause of *Ad* and *Buy*. The shaded node *S* indicates that the case has been included in the database. (b) A Bayesian network for which *A* causes *B* is the only causal explanation, given the causal Markov condition.

equate $p(y|x)$ with $p(y|\hat{x})$, where \hat{x} denotes the intervention that forces X to be x . For purposes of discussion, we use this rule as an operational definition for cause. Pearl (1995) and Heckerman and Shachter (1995) discuss versions of this definition that are more complete and more precise.

In our example, knowledge that A causes B allows us to learn $p(b|\hat{a})$ and $p(b|\hat{\bar{a}})$ from observations alone—no randomized experiment is needed. But how are we to determine whether or not A causes B ? The answer lies in an assumption about the connection between causal and probabilistic dependence known as the *causal Markov condition*, described by Spirtes et al. (1993). We say that a directed acyclic graph \mathcal{C} is a *causal graph for variables \mathbf{X}* if the nodes in \mathcal{C} are in a one-to-one correspondence with \mathbf{X} , and there is an arc from node X to node Y in \mathcal{C} if and only if X is a direct cause of Y . The causal Markov condition says that if \mathcal{C} is a causal graph for \mathbf{X} , then \mathcal{C} is also a Bayesian-network structure for the joint physical probability distribution of \mathbf{X} . In Section 3.3, we described a method based on this condition for constructing Bayesian-network structure from causal assertions. Several researchers (e.g., Spirtes et al., 1993) have found that this condition holds in many applications.

Given the causal Markov condition, we can infer causal relationships from conditional-independence and conditional-dependence relationships that we learn from the data.¹⁷ Let us illustrate this process for the marketing example. Suppose we have learned (with high Bayesian probability) that the physical probabilities $p(b|a)$ and $p(b|\bar{a})$ are not equal. Given the causal Markov condition, there are four simple causal explanations for this dependence: (1) A is a cause for B , (2)

¹⁷ Spirtes et al. (1993) also require an assumption known as *faithfulness*. We do not need to make this assumption explicit, because it follows from our assumption that $p(\theta_s|S^h)$ is a probability density function.

B is a cause for A , (3) there is a hidden common cause of A and B (e.g., the person's income), and (4) A and B are causes for data selection. This last explanation is known as *selection bias*. Selection bias would occur, for example, if our database failed to include instances where A and B are false. These four causal explanations for the presence of the arcs are illustrated in Figure 3.9a. Of course, more complicated explanations—such as the presence of a hidden common cause and selection bias—are possible.

So far, the causal Markov condition has not told us whether or not A causes B . Suppose, however, that we observe two additional variables: *Income* (I) and *Location* (L), which represent the income and geographic location of the possible purchaser, respectively. Furthermore, suppose we learn (with high probability) the Bayesian network shown in Figure 3.9b. Given the causal Markov condition, the *only* causal explanation for the conditional-independence and conditional-dependence relationships encoded in this Bayesian network is that Ad is a cause for Buy . That is, none of the other explanations described in the previous paragraph, or combinations thereof, produce the probabilistic relationships encoded in Figure 3.9b. Based on this observation, Pearl and Verma (1991) and Spirtes et al. (1993) have created algorithms for inferring causal relationships from dependence relationships for more complicated situations.

3.16 A Case Study: College Plans

Real-world applications of techniques that we have discussed can be found in Madigan and Raftery (1994), Lauritzen et al. (1994), Singh and Provan (1995), and Friedman and Goldszmidt (1996). Here, we consider an application that comes from a study by Sewell and Shah (1968), who investigated factors that influence the intention of high school students to attend college. The data have been analyzed by several groups of statisticians, including Whittaker (1990) and Spirtes et al. (1993), all of whom have used non-Bayesian techniques.

Sewell and Shah (1968) measured the following variables for 10,318 Wisconsin high school seniors: *Sex* (SEX): male, female; *Socioeconomic Status* (SES): low, lower middle, upper middle, high; *Intelligence Quotient* (IQ): low, lower middle, upper middle, high; *Parental Encouragement* (PE): low, high; and *College Plans* (CP): yes, no. Our goal here is to understand the (possibly causal) relationships among these variables.

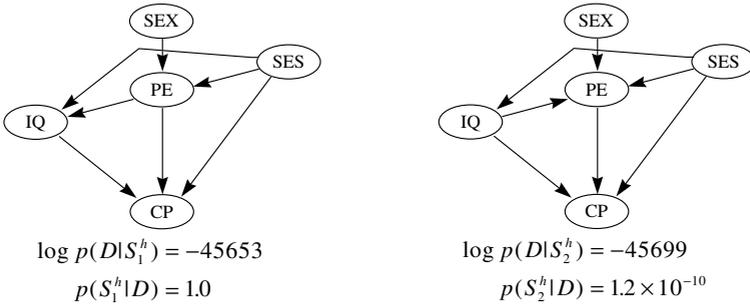
The data are described by the sufficient statistics in Table 3.2. Each entry denotes the number of cases in which the five variables take on some particular configuration. The first entry corresponds to the configuration SEX=male, SES=low, IQ=low, PE=low, and CP=yes. The remaining entries correspond to configurations obtained by cycling through the states of each variable such that the last variable (CP) varies most quickly. Thus, for example, the upper (lower) half of the table corresponds to male (female) students.

As a first pass, we analyzed the data assuming no hidden variables. To generate priors for network parameters, we used the method described in Section 3.10.1 with an equivalent sample size of 5 and a prior network where

Table 3.2. Sufficient statistics for the Sewall and Shah (1968) study

4	349	13	64	9	207	33	72	12	126	38	54	10	67	49	43
2	232	27	84	7	201	64	95	12	115	93	92	17	79	119	59
8	166	47	91	6	120	74	110	17	92	148	100	6	42	198	73
4	48	39	57	5	47	123	90	9	41	224	65	8	17	414	54
5	454	9	44	5	312	14	47	8	216	20	35	13	96	28	24
11	285	29	61	19	236	47	88	12	164	62	85	15	113	72	50
7	163	36	72	13	193	75	90	12	174	91	100	20	81	142	77
6	50	36	58	5	70	110	76	12	48	230	81	13	49	360	98

Reproduced by permission from the University of Chicago Press. ©1968 by The University of Chicago. All rights reserved.

**Fig. 3.10.** The a posteriori most likely network structures without hidden variables

$p(\mathbf{x}|S_c^h)$ is uniform. (The results were not sensitive to the choice of parameter priors. For example, none of the results reported in this section changed qualitatively for equivalent sample sizes ranging from 3 to 40.) For structure priors, we assumed that all network structures were equally likely, except we excluded structures where *SEX* and/or *SES* had parents, and/or *CP* had children. Because the data set was complete, we used Equations 3.34 and 3.35 to compute the posterior probabilities of network structures. The two most likely network structures that we found after an exhaustive search over all structures are shown in Figure 3.10. Note that the most likely graph has a posterior probability that is extremely close to one.

If we adopt the causal Markov assumption and also assume that there are no hidden variables, then the arcs in both graphs can be interpreted causally. Some results are not surprising—for example the causal influence of socioeconomic status and IQ on college plans. Other results are more interesting. For example, from either graph we conclude that sex influences college plans only indirectly through parental influence. Also, the two graphs differ only by the orientation

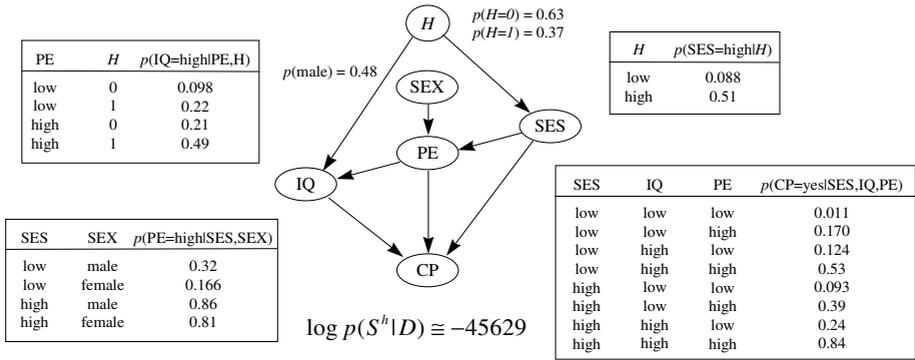


Fig. 3.11. The a posteriori most likely network structure with a hidden variable. Probabilities shown are MAP values. Some probabilities are omitted for lack of space.

of the arc between PE and IQ. Either causal relationship is plausible. We note that the second most likely graph was selected by Spirtes et al. (1993), who used a non-Bayesian approach with slightly different assumptions.

The most suspicious result is the suggestion that socioeconomic status has a direct influence on IQ. To question this result, we considered new models obtained from the models in Figure 3.10 by replacing this direct influence with a hidden variable pointing to both *SES* and *IQ*. We also considered models where the hidden variable pointed to *SES*, *IQ*, and *PE*, and none, one, or both of the connections *SES*—*PE* and *PE*—*IQ* were removed. For each structure, we varied the number of states of the hidden variable from two to six.

We computed the posterior probability of these models using the Cheeseman-Stutz (1995) variant of the Laplace approximation. To find the MAP $\tilde{\theta}_s$, we used the EM algorithm, taking the largest local maximum from among 100 runs with different random initializations of θ_s . Among the models we considered, the one with the highest posterior probability is shown in Figure 3.11. This model is $2 \cdot 10^{10}$ times more likely than the best model containing no hidden variable. The next most likely model containing a hidden variable, which has one additional arc from the hidden variable to *PE*, is $5 \cdot 10^{-9}$ times less likely than the best model. Thus, if we again adopt the causal Markov assumption and also assume that we have not omitted a reasonable model from consideration, then we have strong evidence that a hidden variable is influencing both socioeconomic status and IQ in this population—a sensible result. An examination of the probabilities in Figure 3.11 suggests that the hidden variable corresponds to some measure of “parent quality”.

3.17 Pointers to Literature and Software

Like all tutorials, this one is incomplete. For those readers interested in learning more about graphical models and methods for learning them, we offer the

following additional references and pointers to software. Buntine (1996) provides another guide to the literature.

Spirtes et al. (1993) and Pearl (1995) use methods based on large-sample approximations to learn Bayesian networks. In addition, as we have discussed, they describe methods for learning causal relationships from observational data.

In addition to directed models, researchers have explored network structures containing undirected edges as a knowledge representation. These representations are discussed (e.g.) in Lauritzen (1982), Verma and Pearl (1990), Frydenberg (1990), Whittaker (1990), and Richardson (1997). Bayesian methods for learning such models from data are described by Dawid and Lauritzen (1993) and Buntine (1994).

Finally, several research groups have developed software systems for learning graphical models. For example, Scheines et al. (1994) have developed a software program called TETRAD II for learning about cause and effect. Badsberg (1992) and Højsgaard et al. (1994) have built systems that can learn with mixed graphical models using a variety of criteria for model selection. Thomas, Spiegelhalter, and Gilks (1992) have created a system called BUGS that takes a learning problem specified as a Bayesian network and compiles this problem into a Gibbs-sampler computer program.

Acknowledgments

I thank Max Chickering, Usama Fayyad, Eric Horvitz, Chris Meek, Koos Rommelse, and Padhraic Smyth for their comments on earlier versions of this manuscript. I also thank Max Chickering for implementing the software used to analyze the Sewall and Shah (1968) data, and Chris Meek for bringing this data set to my attention.

Notation

X, Y, Z, \dots Variables or their corresponding nodes in a Bayesian network

$\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \dots$ Sets of variables or corresponding sets of nodes

$X = x$ Variable X is in state x

$\mathbf{X} = \mathbf{x}$ The set of variables \mathbf{X} is in configuration \mathbf{x}

$\mathbf{x}, \mathbf{y}, \mathbf{z}$ Typically refer to a complete case, an incomplete case, and missing data in a case, respectively

$\mathbf{X} \setminus \mathbf{Y}$ The variables in X that are not in Y

D A data set: a set of cases

D_l The first $l - 1$ cases in D

$p(\mathbf{x}|\mathbf{y})$ The probability that $\mathbf{X} = \mathbf{x}$ given $\mathbf{Y} = \mathbf{y}$
(also used to describe a probability density, probability distribution, and probability density)

$E_{p(\cdot)}(x)$ The expectation of x with respect to $p(\cdot)$

S A Bayesian network structure (a directed acyclic graph)

\mathbf{Pa}_i The variable or node corresponding to the parents of node X_i in a Bayesian network structure

\mathbf{pa}_i A configuration of the variables \mathbf{Pa}_i

r_i The number of states of discrete variable X_i

q_i The number of configurations of \mathbf{Pa}_i

S_c A complete network structure

S^h The hypothesis corresponding to network structure S

θ_{ijk} The multinomial parameter corresponding to the probability $p(X_i = x_i^k | \mathbf{Pa}_i = \mathbf{pa}_i^j)$

$\boldsymbol{\theta}_{ij} = (\theta_{ij2}, \dots, \theta_{ijr_i})$

$\boldsymbol{\theta}_i = (\theta_{i1}, \dots, \theta_{iq_i})$

$\boldsymbol{\theta}_s = (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_n)$

α An equivalent sample size

α_{ijk} The Dirichlet hyperparameter corresponding to θ_{ijk}

$\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$

N_{ijk} The number of cases in data set D where $X_i = x_i^k$ and $\mathbf{Pa}_i = \mathbf{pa}_i^j$

$N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$

References

- [Aliferis and Cooper, 1994] Aliferis, C., Cooper, G.: An evaluation of an algorithm for inductive learning of Bayesian belief networks using simulated data sets. In: Proceedings of Tenth Conference on Uncertainty in Artificial Intelligence, Seattle, WA, pp. 8–14. Morgan Kaufmann, San Francisco (1994)
- [Badsberg, 1992] Badsberg, J.: Model search in contingency tables by CoCo. In: Dodge, Y., Whittaker, J. (eds.) Computational Statistics, pp. 251–256. Physica Verlag, Heidelberg (1992)

- [Becker and LeCun, 1989] Becker, S., LeCun, Y.: Improving the convergence of back-propagation learning with second order methods. In: Proceedings of the 1988 Connectionist Models Summer School, pp. 29–37. Morgan Kaufmann, San Francisco (1989)
- [Beinlich et al., 1989] Beinlich, I., Suermondt, H., Chavez, R., Cooper, G.: The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In: Proceedings of the Second European Conference on Artificial Intelligence in Medicine, London, pp. 247–256. Springer, Berlin (1989)
- [Bernardo, 1979] Bernardo, J.: Expected information as expected utility. *Annals of Statistics* 7, 686–690 (1979)
- [Bernardo and Smith, 1994] Bernardo, J., Smith, A.: *Bayesian Theory*. John Wiley and Sons, New York (1994)
- [Buntine, 1991] Buntine, W.: Theory refinement on Bayesian networks. In: Proceedings of Seventh Conference on Uncertainty in Artificial Intelligence, Los Angeles, CA, pp. 52–60. Morgan Kaufmann, San Francisco (1991)
- [Buntine, 1993] Buntine, W.: Learning classification trees. In: *Artificial Intelligence Frontiers in Statistics: AI and statistics III*. Chapman and Hall, New York (1993)
- [Buntine, 1996] Buntine, W.: A guide to the literature on learning graphical models. *IEEE Transactions on Knowledge and Data Engineering* 8, 195–210 (1996)
- [Chaloner and Duncan, 1983] Chaloner, K., Duncan, G.: Assessment of a beta prior distribution: PM elicitation. *The Statistician* 32, 174–180 (1983)
- [Cheeseman and Stutz, 1995] Cheeseman, P., Stutz, J.: Bayesian classification (Auto-Class): Theory and results. In: Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R. (eds.) *Advances in Knowledge Discovery and Data Mining*, pp. 153–180. AAAI Press, Menlo Park (1995)
- [Chib, 1995] Chib, S.: Marginal likelihood from the Gibbs output. *Journal of the American Statistical Association* 90, 1313–1321 (1995)
- [Chickering, 1995] Chickering, D.: A transformational characterization of equivalent Bayesian network structures. In: Proceedings of Eleventh Conference on Uncertainty in Artificial Intelligence, Montreal, QU, pp. 87–98. Morgan Kaufmann, San Francisco (1995)
- [Chickering, 1996b] Chickering, D.: Learning equivalence classes of Bayesian-network structures. In: Proceedings of Twelfth Conference on Uncertainty in Artificial Intelligence, Portland, OR. Morgan Kaufmann, San Francisco (1996)
- [Chickering et al., 1995] Chickering, D., Geiger, D., Heckerman, D.: Learning Bayesian networks: Search methods and experimental results. In: Proceedings of Fifth Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, FL. Society for Artificial Intelligence in Statistics, pp. 112–128 (1995)
- [Chickering and Heckerman, 1996] Chickering, D., Heckerman, D.: Efficient approximations for the marginal likelihood of incomplete data given a Bayesian network. Technical Report MSR-TR-96-08, Microsoft Research, Redmond, WA (Revised, November 1996)
- [Cooper, 1990] Cooper, G.: Computational complexity of probabilistic inference using Bayesian belief networks (Research note). *Artificial Intelligence* 42, 393–405 (1990)
- [Cooper and Herskovits, 1992] Cooper, G., Herskovits, E.: A Bayesian method for the induction of probabilistic networks from data. *Machine Learning* 9, 309–347 (1992)
- [Cooper and Herskovits, 1991] Cooper, G., Herskovits, E.: A Bayesian method for the induction of probabilistic networks from data. Technical Report SMI-91-1, Section on Medical Informatics, Stanford University (January 1991)
- [Cox, 1946] Cox, R.: Probability, frequency and reasonable expectation. *American Journal of Physics* 14, 1–13 (1946)

- [Dagum and Luby, 1993] Dagum, P., Luby, M.: Approximating probabilistic inference in bayesian belief networks is np-hard. *Artificial Intelligence* 60, 141–153 (1993)
- [D’Ambrosio, 1991] D’Ambrosio, B.: Local expression languages for probabilistic dependence. In: *Proceedings of Seventh Conference on Uncertainty in Artificial Intelligence*, Los Angeles, CA, pp. 95–102. Morgan Kaufmann, San Francisco (1991)
- [Darwiche and Provan, 1996] Darwiche, A., Provan, G.: Query DAGs: A practical paradigm for implementing belief-network inference. In: *Proceedings of Twelfth Conference on Uncertainty in Artificial Intelligence*, Portland, OR, pp. 203–210. Morgan Kaufmann, San Francisco (1996)
- [Dawid, 1984] Dawid, P.: Statistical theory. The prequential approach (with discussion). *Journal of the Royal Statistical Society A* 147, 178–292 (1984)
- [Dawid, 1992] Dawid, P.: Applications of a general propagation algorithm for probabilistic expert systems. *Statistics and Computing* 2, 25–36 (1992)
- [de Finetti, 1970] de Finetti, B.: *Theory of Probability*. Wiley and Sons, New York (1970)
- [Dempster et al., 1977] Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, B* 39, 1–38 (1977)
- [DiCiccio et al., 1995] DiCiccio, T., Kass, R., Raftery, A., Wasserman, L.: Computing Bayes factors by combining simulation and asymptotic approximations. Technical Report 630, Department of Statistics, Carnegie Mellon University, PA (July 1995)
- [Friedman, 1995] Friedman, J.: Introduction to computational learning and statistical prediction. Technical report, Department of Statistics, Stanford University (1995)
- [Friedman, 1996] Friedman, J.: On bias, variance, 0/1-loss, and the curse of dimensionality. *Data Mining and Knowledge Discovery*, 1 (1996)
- [Friedman and Goldszmidt, 1996] Friedman, N., Goldszmidt, M.: Building classifiers using Bayesian networks. In: *Proceedings AAAI 1996 Thirteenth National Conference on Artificial Intelligence*, Portland, OR, pp. 1277–1284. AAAI Press, Menlo Park (1996)
- [Frydenberg, 1990] Frydenberg, M.: The chain graph Markov property. *Scandinavian Journal of Statistics* 17, 333–353 (1990)
- [Geiger and Heckerman, 1995] Geiger, D., Heckerman, D.: A characterization of the Dirichlet distribution applicable to learning Bayesian networks. Technical Report MSR-TR-94-16, Microsoft Research, Redmond, WA (Revised, February 1995)
- [Geiger et al., 1996] Geiger, D., Heckerman, D., Meek, C.: Asymptotic model selection for directed networks with hidden variables. In: *Proceedings of Twelfth Conference on Uncertainty in Artificial Intelligence*, Portland, OR, pp. 283–290. Morgan Kaufmann, San Francisco (1996)
- [Geman and Geman, 1984] Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6, 721–742 (1984)
- [Gilks et al., 1996] Gilks, W., Richardson, S., Spiegelhalter, D.: *Markov Chain Monte Carlo in Practice*. Chapman and Hall, Boca Raton (1996)
- [Good, 1950] Good, I.: *Probability and the Weighing of Evidence*. Hafners, New York (1950)
- [Heckerman, 1989] Heckerman, D.: A tractable algorithm for diagnosing multiple diseases. In: *Proceedings of the Fifth Workshop on Uncertainty in Artificial Intelligence*, Windsor, ON, pp. 174–181. Association for Uncertainty in Artificial Intelligence, Mountain View, CA (1989); Also In: Henrion, M., Shachter, R., Kanal, L., Lemmer, J. (eds.) *Uncertainty in Artificial Intelligence* 5, pp. 163–171. North-Holland, New York (1990)

- [Heckerman, 1995] Heckerman, D.: A Bayesian approach for learning causal networks. In: Proceedings of Eleventh Conference on Uncertainty in Artificial Intelligence, Montreal, QU, pp. 285–295. Morgan Kaufmann, San Francisco (1995)
- [Heckerman and Geiger, 1996] Heckerman, D., Geiger, D.: Likelihoods and priors for Bayesian networks. Technical Report MSR-TR-95-54, Microsoft Research, Redmond, WA (Revised, November 1996)
- [Heckerman et al., 1995a] Heckerman, D., Geiger, D., Chickering, D.: Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning* 20, 197–243 (1995a)
- [Heckerman et al., 1995b] Heckerman, D., Mamdani, A., Wellman, M.: Real-world applications of Bayesian networks. *Communications of the ACM* 38 (1995b)
- [Heckerman and Shachter, 1995] Heckerman, D., Shachter, R.: Decision-theoretic foundations for causal reasoning. *Journal of Artificial Intelligence Research* 3, 405–430 (1995)
- [Højsgaard et al., 1994] Højsgaard, S., Skjøth, F., Thiesson, B.: User’s guide to BIOFROST. Technical report, Department of Mathematics and Computer Science, Aalborg, Denmark (1994)
- [Howard, 1970] Howard, R.: Decision analysis: Perspectives on inference, decision, and experimentation. *Proceedings of the IEEE* 58, 632–643 (1970)
- [Howard and Matheson, 1981] Howard, R., Matheson, J.: Influence diagrams. In: Howard, R., Matheson, J. (eds.) *Readings on the Principles and Applications of Decision Analysis*, Strategic Decisions Group, Menlo Park, CA, vol. II, pp. 721–762 (1981)
- [Howard and Matheson, 1983] Howard, R., Matheson, J. (eds.): *The Principles and Applications of Decision Analysis*, Strategic Decisions Group, Menlo Park, CA (1983)
- [Humphreys and Freedman, 1996] Humphreys, P., Freedman, D.: The grand leap. *British Journal for the Philosophy of Science* 47, 113–118 (1996)
- [Jaakkola and Jordan, 1996] Jaakkola, T., Jordan, M.: Computing upper and lower bounds on likelihoods in intractable networks. In: Proceedings of Twelfth Conference on Uncertainty in Artificial Intelligence, Portland, OR, pp. 340–348. Morgan Kaufmann, San Francisco (1996)
- [Jensen, 1996] Jensen, F.: *An Introduction to Bayesian Networks*. Springer, Heidelberg (1996)
- [Jensen and Andersen, 1990] Jensen, F., Andersen, S.: Approximations in Bayesian belief universes for knowledge based systems. Technical report, Institute of Electronic Systems, Aalborg University, Aalborg, Denmark (1990)
- [Jensen et al., 1990] Jensen, F., Lauritzen, S., Olesen, K.: Bayesian updating in recursive graphical models by local computations. *Computational Statistics Quarterly* 4, 269–282 (1990)
- [Kass and Raftery, 1995] Kass, R., Raftery, A.: Bayes factors. *Journal of the American Statistical Association* 90, 773–795 (1995)
- [Kass et al., 1988] Kass, R., Tierney, L., Kadane, J.: Asymptotics in Bayesian computation. In: Bernardo, J., DeGroot, M., Lindley, D., Smith, A. (eds.) *Bayesian Statistics*, vol. 3, pp. 261–278. Oxford University Press, Oxford (1988)
- [Koopman, 1936] Koopman, B.: On distributions admitting a sufficient statistic. *Transactions of the American Mathematical Society* 39, 399–409 (1936)
- [Korf, 1993] Korf, R.: Linear-space best-first search. *Artificial Intelligence* 62, 41–78 (1993)
- [Lauritzen, 1982] Lauritzen, S.: *Lectures on Contingency Tables*. University of Aalborg Press, Aalborg (1982)

- [Lauritzen, 1992] Lauritzen, S.: Propagation of probabilities, means, and variances in mixed graphical association models. *Journal of the American Statistical Association* 87, 1098–1108 (1992)
- [Lauritzen and Spiegelhalter, 1988] Lauritzen, S., Spiegelhalter, D.: Local computations with probabilities on graphical structures and their application to expert systems. *J. Royal Statistical Society B* 50, 157–224 (1988)
- [Lauritzen et al., 1994] Lauritzen, S., Thiesson, B., Spiegelhalter, D.: Diagnostic systems created by model selection methods: A case study. In: Cheeseman, P., Oldford, R. (eds.) *AI and Statistics IV*. *Lecture Notes in Statistics*, vol. 89, pp. 143–152. Springer, New York (1994)
- [MacKay, 1992a] MacKay, D.: Bayesian interpolation. *Neural Computation* 4, 415–447 (1992a)
- [MacKay, 1992b] MacKay, D.: A practical Bayesian framework for backpropagation networks. *Neural Computation* 4, 448–472 (1992b)
- [MacKay, 1996] MacKay, D.: Choice of basis for the Laplace approximation. Technical report, Cavendish Laboratory, Cambridge, UK (1996)
- [Madigan et al., 1995] Madigan, D., Garvin, J., Raftery, A.: Eliciting prior information to enhance the predictive performance of Bayesian graphical models. *Communications in Statistics: Theory and Methods* 24, 2271–2292 (1995)
- [Madigan and Raftery, 1994] Madigan, D., Raftery, A.: Model selection and accounting for model uncertainty in graphical models using Occam’s window. *Journal of the American Statistical Association* 89, 1535–1546 (1994)
- [Madigan et al., 1996] Madigan, D., Raftery, A., Volinsky, C., Hoeting, J.: Bayesian model averaging. In: *Proceedings of the AAAI Workshop on Integrating Multiple Learned Models*, Portland, OR (1996)
- [Madigan and York, 1995] Madigan, D., York, J.: Bayesian graphical models for discrete data. *International Statistical Review* 63, 215–232 (1995)
- [Martin and VanLehn, 1995] Martin, J., VanLehn, K.: Discrete factor analysis: Learning hidden variables in bayesian networks. Technical report, Department of Computer Science, University of Pittsburgh, PA. (1995), <http://bert.cs.pitt.edu/vanlehn>
- [Meng and Rubin, 1991] Meng, X., Rubin, D.: Using EM to obtain asymptotic variance-covariance matrices: The SEM algorithm. *Journal of the American Statistical Association* 86, 899–909 (1991)
- [Neal, 1993] Neal, R.: Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto (1993)
- [Olmsted, 1983] Olmsted, S.: On representing and solving decision problems. PhD thesis, Department of Engineering-Economic Systems, Stanford University (1983)
- [Pearl, 1986] Pearl, J.: Fusion, propagation, and structuring in belief networks. *Artificial Intelligence* 29, 241–288 (1986)
- [Pearl, 1995] Pearl, J.: Causal diagrams for empirical research. *Biometrika* 82, 669–710 (1995)
- [Pearl and Verma, 1991] Pearl, J., Verma, T.: A theory of inferred causation. In: Allen, J., Fikes, R., Sandewall, E. (eds.) *Knowledge Representation and Reasoning: Proceedings of the Second International Conference*, pp. 441–452. Morgan Kaufmann, New York (1991)
- [Pitman, 1936] Pitman, E.: Sufficient statistics and intrinsic accuracy. *Proceedings of the Cambridge Philosophy Society* 32, 567–579 (1936)
- [Raftery, 1995] Raftery, A.: Bayesian model selection in social research. In: Marsden, P. (ed.) *Sociological Methodology*. Blackwells, Cambridge (1995)

- [Raftery, 1996] Raftery, A.: Hypothesis testing and model selection, ch. 10. Chapman and Hall, Boca Raton (1996)
- [Ramamurthi and Agogino, 1988] Ramamurthi, K., Agogino, A.: Real time expert system for fault tolerant supervisory control. In: Tipnis, V., Patton, E. (eds.) *Computers in Engineering*, American Society of Mechanical Engineers, Corte Madera, CA, pp. 333–339 (1988)
- [Ramsey, 1931] Ramsey, F.: Truth and probability. In: Braithwaite, R. (ed.) *The Foundations of Mathematics and other Logical Essays*. Humanities Press, London (1931); (Reprinted in Kyburg and Smokler, 1964)
- [Richardson, 1997] Richardson, T.: Extensions of undirected and acyclic, directed graphical models. In: *Proceedings of Sixth Conference on Artificial Intelligence and Statistics*, Ft. Lauderdale, FL, pp. 407–419. Society for Artificial Intelligence in Statistics (1997)
- [Rissanen, 1987] Rissanen, J.: Stochastic complexity (with discussion). *Journal of the Royal Statistical Society, Series B* 49, 223–239, 253–265 (1987)
- [Robins, 1986] Robins, J.: A new approach to causal inference in mortality studies with sustained exposure results. *Mathematical Modelling* 7, 1393–1512 (1986)
- [Rubin, 1978] Rubin, D.: Bayesian inference for causal effects: The role of randomization. *Annals of Statistics* 6, 34–58 (1978)
- [Russell et al., 1995] Russell, S., Binder, J., Koller, D., Kanazawa, K.: Local learning in probabilistic networks with hidden variables. In: *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, Montreal, QU, pp. 1146–1152. Morgan Kaufmann, San Mateo (1995)
- [Saul et al., 1996] Saul, L., Jaakkola, T., Jordan, M.: Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research* 4, 61–76 (1996)
- [Savage, 1954] Savage, L.: *The Foundations of Statistics*. Dover, New York (1954)
- [Schervish, 1995] Schervish, M.: *Theory of Statistics*. Springer, Heidelberg (1995)
- [Schwarz, 1978] Schwarz, G.: Estimating the dimension of a model. *Annals of Statistics* 6, 461–464 (1978)
- [Sewell and Shah, 1968] Sewell, W., Shah, V.: Social class, parental encouragement, and educational aspirations. *American Journal of Sociology* 73, 559–572 (1968)
- [Shachter, 1988] Shachter, R.: Probabilistic inference and influence diagrams. *Operations Research* 36, 589–604 (1988)
- [Shachter et al., 1990] Shachter, R., Andersen, S., Poh, K.: Directed reduction algorithms and decomposable graphs. In: *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence*, Boston, MA, pp. 237–244. Association for Uncertainty in Artificial Intelligence, Mountain View, CA (1990)
- [Shachter and Kenley, 1989] Shachter, R., Kenley, C.: Gaussian influence diagrams. *Management Science* 35, 527–550 (1989)
- [Silverman, 1986] Silverman, B.: *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, New York (1986)
- [Singh and Provan, 1995] Singh, M., Provan, G.: Efficient learning of selective Bayesian network classifiers. Technical Report MS-CIS-95-36, Computer and Information Science Department, University of Pennsylvania, Philadelphia, PA (November 1995)
- [Spetzler and Stael von Holstein, 1975] Spetzler, C., Stael von Holstein, C.: Probability encoding in decision analysis. *Management Science* 22, 340–358 (1975)
- [Spiegelhalter et al., 1993] Spiegelhalter, D., Dawid, A., Lauritzen, S., Cowell, R.: Bayesian analysis in expert systems. *Statistical Science* 8, 219–282 (1993)

- [Spiegelhalter and Lauritzen, 1990] Spiegelhalter, D., Lauritzen, S.: Sequential updating of conditional probabilities on directed graphical structures. *Networks* 20, 579–605 (1990)
- [Spirtes et al., 1993] Spirtes, P., Glymour, C., Scheines, R.: *Causation, Prediction, and Search*. Springer, New York (1993)
- [Spirtes and Meek, 1995] Spirtes, P., Meek, C.: Learning Bayesian networks with discrete variables from data. In: *Proceedings of First International Conference on Knowledge Discovery and Data Mining*, Montreal, QU. Morgan Kaufmann, San Francisco (1995)
- [Suermondt and Cooper, 1991] Suermondt, H., Cooper, G.: A combination of exact algorithms for inference on Bayesian belief networks. *International Journal of Approximate Reasoning* 5, 521–542 (1991)
- [Thiesson, 1995a] Thiesson, B.: Accelerated quantification of Bayesian networks with incomplete data. In: *Proceedings of First International Conference on Knowledge Discovery and Data Mining*, Montreal, QU, pp. 306–311. Morgan Kaufmann, San Francisco (1995a)
- [Thiesson, 1995b] Thiesson, B.: Score and information for recursive exponential models with incomplete data. Technical report, Institute of Electronic Systems, Aalborg University, Aalborg, Denmark (1995b)
- [Thomas et al., 1992] Thomas, A., Spiegelhalter, D., Gilks, W.: Bugs: A program to perform Bayesian inference using Gibbs sampling. In: Bernardo, J., Berger, J., Dawid, A., Smith, A. (eds.) *Bayesian Statistics*, vol. 4, pp. 837–842. Oxford University Press, Oxford (1992)
- [Tukey, 1977] Tukey, J.: *Exploratory Data Analysis*. Addison-Wesley, Reading (1977)
- [Tversky and Kahneman, 1974] Tversky, A., Kahneman, D.: Judgment under uncertainty: Heuristics and biases. *Science* 185, 1124–1131 (1974)
- [Verma and Pearl, 1990] Verma, T., Pearl, J.: Equivalence and synthesis of causal models. In: *Proceedings of Sixth Conference on Uncertainty in Artificial Intelligence*, Boston, MA, pp. 220–227. Morgan Kaufmann, San Francisco (1990)
- [Whittaker, 1990] Whittaker, J.: *Graphical Models in Applied Multivariate Statistics*. John Wiley and Sons, Chichester (1990)
- [Winkler, 1967] Winkler, R.: The assessment of prior distributions in Bayesian analysis. *American Statistical Association Journal* 62, 776–800 (1967)

The Causal Interpretation of Bayesian Networks

Kevin B. Korb and Ann E. Nicholson

Clayton School of Information Technology
Monash University
Clayton, Victoria 3800, Australia
{kevin.korb,ann.nicholson}@infotech.monash.edu.au

Summary. The common interpretation of Bayesian networks is that they are vehicles for representing probability distributions, in a graphical form supportive of human understanding and with computational mechanisms supportive of probabilistic reasoning (updating). But the interpretation of Bayesian networks assumed by causal discovery algorithms is causal: the links in the graphs specifically represent direct causal connections between variables. However, there is some tension between these two interpretations. The philosophy of probabilistic causation posits a particular connection between the two, namely that causal relations of certain kinds give rise to probabilistic relations of certain kinds. Causal discovery algorithms take advantage of this kind of connection by ruling out some Bayesian networks given observational data not supported by the posited probability-causality relation. But the discovered (remaining) Bayesian networks are then specifically causal, and not simply arbitrary representations of probability.

There are multiple contentious issues underlying any causal interpretation of Bayesian networks. We will address the following questions:

- Since Bayesian net construction rules allow the construction of multiple distinct networks to represent the very same probability distribution, how can we come to prefer any specific one as “the” causal network?
- Since Bayesian nets within a Verma-Pearl pattern are strongly indistinguishable, how can causal discovery ever come to select exactly one network as “the” causal network?
- Causal discovery assumes faithfulness (that d-connections in the model are accompanied by probabilistic dependency in the system modeled). However, some physical systems cannot be modeled faithfully under a causal interpretation. How can causal discovery cope with that?

Here we introduce a causal interpretation of Bayesian networks by way of answering these questions and then apply this interpretation to answering further questions about causal power, explanation and responsibility.

Keywords: causal discovery, faithfulness, Bayesian networks, probabilistic causality, intervention, causal power, causal responsibility.

4.1 Introduction

In the last decade Bayesian networks have risen to prominence as the preferred technology for probabilistic reasoning in artificial intelligence, with a

proliferation of techniques for fast and approximate updating and also for their automated discovery from data (causal discovery, or “data mining” of Bayesian networks). Philosophers of science have also begun to adopt the technology for reasoning about causality and methodology (e.g., [2]; [23]). Both the causal discovery and the philosophical analysis depend upon the propriety of a causal interpretation of the Bayesian nets in use. However, the standard semantics for Bayesian networks are purely probabilistic (see, e.g., [39]), and various of their properties — such as the statistical indistinguishability of distinct Bayesian networks [53, 7] — seem to undermine any causal interpretation. As a result, skeptics of causal interpretation are growing along with the technology itself.

4.2 Bayesian Networks

We begin with a perfectly orthodox (we hope) introduction to the concepts and notation used in Bayesian networks (for a more detailed introduction see [28, Chap 2]); readers familiar with these are advised to skip to the next section. A **Bayesian network** is a directed acyclic graph (dag), M , over a set of variables with associated conditional probabilities θ which together represent a probability distribution over the joint states of the variables. The fully parameterized model will be designated $M(\theta)$. For a simple (unparameterized) example, see Figure 4.1. *Rain* and *Sprinkler* are the **root** nodes (equivalently, the **exogenous** variables), which report whether there is rain overnight and whether the automatic sprinkler system comes on. The **endogenous** (non-root) variables are *Lawn*, which describes whether or not the lawn is wet, and *Newspaper* and *Carpet*, which respectively describe the resultant soggy and muddiness when the dog retrieves the newspaper. Note that we shall vary between talk of variables and their values and talk of **event types** (e.g., rain) and their corresponding **token events** (e.g., last night’s rain) without much ado.

Probabilistic reasoning is computationally intractable (specifically, NP-hard; [10]). The substantial advantage Bayesian networks offer for probabilistic reasoning is that, if the probability distribution can be represented with a sparse

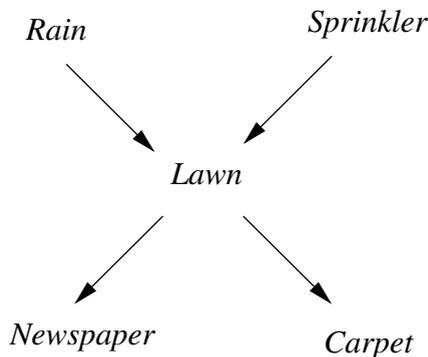


Fig. 4.1. A simple Bayesian network

network (i.e., with few arcs), the computations become practicable. And, most networks of actual interest to us are sparse. Given a sparse network, the probabilistic implications of observations of a subset of the variables can be readily computed using any of a large number of available algorithms [28]. In order for the computational savings afforded by low arc density to be realized, the lack of an arc between two variables must be reflected in a probabilistic independence in the system being modeled. Thus, in a simple two-variable model with nodes X and Y , a missing arc implies that X and Y are probabilistically independent. If they are not, then the Bayesian network simply fails to be an appropriate model. Thus, in our example, *Rain* and *Sprinkler* must be independent of each other; if the sprinkler system is turned off in rainy weather, then Figure 4.1 is simply the wrong model (which could be made right by then adding $Rain \rightarrow Sprinkler$).

X and Y being probabilistically independent just means that $P(X = x_i | Y = y_j) = P(X = x_i)$ for any two states x_i and y_j . **Conditional independence** generalizes this to cases where observations of a third variable may induce an independence between the first two variables, which may otherwise be dependent. Philosophers, following [43], have tended to call this relationship **screening off**. For example, *Rain* and *Newspaper* are presumably dependent in Figure 4.1; however, if we hold fixed the state of the lawn — say, we already know it is wet — then they are no longer probabilistically related: $P(Newspaper | Lawn, Rain) = P(Newspaper | Lawn)$, which we will commonly abbreviate as $Newspaper \perp\!\!\!\perp Rain | Lawn$. Given these and like facts for other variables, Figure 4.1 is said to have the **Markov property**: that is, all of the conditional independencies implied by the Bayesian network are true of the actual system (or, equivalently, it is said to be an **independence map (I-map)** of the system). We shall often assume that our Bayesian networks are Markov, since, as we have indicated, when they are not, this is simply because we have chosen an incorrect model for our problem.

In the opposite condition, where all apparent dependencies in the network are realized in the system, the network is said to be **faithful** to the system (or, the network is called a **dependence-map (D-map)** of the system). A network which both satisfies the Markov property and is faithful is said to be a **perfect map** of the system.¹ There is no general requirement for a Bayesian network to be faithful in order to be considered an adequate probabilistic model. In particular, arcs can always be added which do nothing — they can be parameterized so that no additional probabilistic influence between variables is implied. Of course, there is a computational cost to doing so, but there is no misrepresentation of the probability distribution. What we are normally interested in, however, are I-maps that are **minimal**: i.e., I-maps such that if any arc is deleted, the model is no longer an I-map for the system of interest. A minimal I-map need not necessarily also be a perfect map, although they typically are; in particular, there are some systems which have multiple distinct minimal I-maps.

¹ The concept of “faithfulness” comes from the “faithfulness condition” of Spirtes, et al. [47]; they also, somewhat confusingly, talk about graphs and distributions being “faithful to each other”, when we prefer to talk about perfect maps.

We shall often be interested in probabilistic dependencies carried by particular paths across a network. A **path** is a sequence of nodes which can be visited by traversing arcs in the model (disregarding their directions) in which no node is visited twice. A **directed path** proceeds entirely in the direction of the arcs traversed. A fundamental graph-theoretic concept is that of two nodes X and Y being **d-separated** by a set of nodes \mathbf{Z} , which we will abbreviate $X \perp Y | \mathbf{Z}$. Formally,

Definition 1 (D-separation)

X and Y are **d-separated** given \mathbf{Z} (for any subset \mathbf{Z} of variables not including X or Y) if and only if each distinct path Φ between them is cut by one of the graph-theoretic conditions:

1. Φ contains a chain $X_1 \rightarrow X_2 \rightarrow X_3$ and $X_2 \in \mathbf{Z}$.
2. Φ contains a common causal structure $X_1 \leftarrow X_2 \rightarrow X_3$ and $X_2 \in \mathbf{Z}$.
3. Φ contains a common effect structure $X_1 \rightarrow X_2 \leftarrow X_3$ (i.e., an **uncovered collision** with X_1 and X_3 not directly connected) and neither X_2 nor any descendant of X_2 is in \mathbf{Z} .

This is readily generalized to sets of variables \mathbf{X} and \mathbf{Y} .

The idea of d-separation is simply that dependencies can be cut by observing intermediate variables (1) or common causes (2), on the one hand, and induced by observing common effects (or their descendants), on the other (3). As for the latter, if we assume as above that the automated sprinkler and rain are independent in Figure 4.1, they will not remain so if we presuppose knowledge of the state of the lawn. For example, if the lawn is wet, something must explain that, so if we learn that there was no rain overnight, we must increase our belief that the sprinkler system came on.

A more formal version of the **Markov property** is then: a model has the Markov property relative to a system if the system has a conditional independence corresponding to every d-separation in the model. I.e.,

$$\forall X, Y, \mathbf{Z} (X \perp Y | \mathbf{Z} \Rightarrow X \perp\!\!\!\perp Y | \mathbf{Z})$$

The opposite condition to d-separation is **d-connection**, when some path between X and Y is *not* blocked by \mathbf{Z} , which we will write $X \not\perp Y | \mathbf{Z}$. Faithfulness of a graph is equivalent to

$$\forall X, Y, \mathbf{Z} (X \not\perp Y | \mathbf{Z} \Rightarrow X \not\perp\!\!\!\perp Y | \mathbf{Z})$$

A concept related to d-connection, but not the same, is that of an active path. We use active paths to consider the probabilistic impact of observations of some variables upon others: a path between X and Y is an **active path** in case an observation of X can induce a change in the probability distribution of Y . The concepts of d-connected paths and active paths are not equivalent because a d-connected path may be inactive due only to its parameterization.

4.3 Are Bayesian Networks Causal Models?

Such are Bayesian networks and some of their associated concepts. The illustrative network of Figure 4.1 is itself clearly a causal model: its arcs identify direct causal relationships between event types corresponding to its nodes — rain *causes* lawns to get wet, etc. But it is clear that there is no necessity in this. We could represent the very same probabilistic facts with a very different network, one which does not respect the causal story, reversing some of the arcs. Indeed, Chickering [7] introduced a transformation rule which allows us to reverse any number of the arcs in a Bayesian network:

Rule 1 (Chickering’s Transformation Rule). *The transformation of M to M' , where $M = M'$ except that, for some variables C and E , $C \rightarrow E \in M$ and $C \leftarrow E \in M'$ (and excepting any arc introduced below), will allow the probability distribution induced by M to be represented via M' so long as:*

if any uncovered collision is introduced or eliminated, then a covering arc is added (e.g., if $A \rightarrow C \rightarrow E \in M$ then A and E must be directly connected in M').

Given any probability distribution, and any causal Bayesian network representing it, we can use Chickering’s rule to find other, *anti-causal*, Bayesian networks. Clearly, this rule only *introduces* arcs and never eliminates any. Thus, when starting with a causal model and applying a sequence of Chickering transformations to find additional models capable of representing the original probability distribution, we can only start from a simpler model and reach (monotonically) ever more complex models. For example, we can apply this rule to our *Sprinkler* model in order to find that the alternative of Figure 4.2 can represent the probabilities just as well. Except that, this model clearly does not represent the probabilities *just as well* — it is far more, needlessly, complex. Since, in general, parameter complexity is exponential in the number of arcs, this complexification is highly undesirable computationally. And the new complexity introduced by Chickering transformations fail to represent the very same causal structure, and under a causal interpretation they will imply falsehoods about the consequences of interventions on its variables. The model arrived at in Figure 4.2, for example, has a wet carpet watering the lawn.

The models in a sequence of Chickering transformations progress monotonically from simpler to more complex; the set of probability distributions representable by those models (by re-parameterizing them) form a sequence of monotonically increasing supersets, with the final model capable of representing all of (and usually more than) the distributions representable by its predecessors.

Intuitively, we might come to a preference for causal models then on simplicity grounds: causal models are the simplest of Bayesian networks capable of representing the probabilistic facts (such nets we will call **admissible models**), whereas the transformed, non-causal networks are inherently more complex. While we believe that this is largely true, and a sufficient motivation to prefer causal over non-causal models, it is not universally true. There are true causal

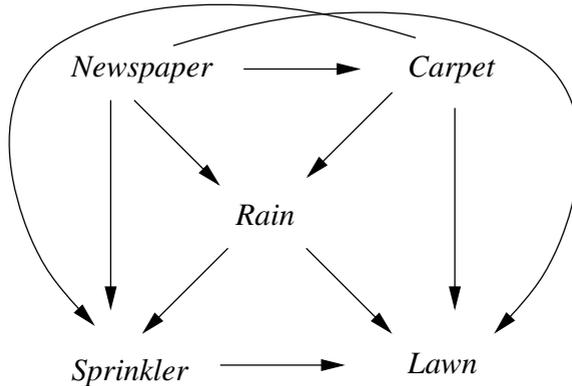


Fig. 4.2. A less simple Bayesian network

models which are more complex than alternative admissible models, although the latter cannot be arrived at by Chickering's rule. We shall see an example of that below. Our real reason for preferring causal models to their probabilistic imitators is that most of our interest in Bayesian networks arises from an interest in understanding the stochastic universe around us, which comes in the form of numerous (partially) independent causal systems. We generate models to explain, predict and manipulate these systems, which models map directly onto those systems — that is what it means to have a true (causal) theory of any such system. Alternative models which arise from arbitrary redirections of causal arcs may have some computational interest, but they have little or no interest in scientific explanation or engineering application. The causal models are metaphysically and epistemologically primary; the alternative models arise from mathematical games.

So, to answer the question heading this section: most Bayesian networks are not causal models; however, most *interesting* Bayesian networks are.

4.4 Causal Discovery and Statistical Indistinguishability

Causal discovery algorithms aim to find the causal model responsible for generating some available data, usually in the form of a set of joint observations over the variables being modeled. Since 1990, when the first general discovery algorithms were invented, many more have been developed. Causal discovery algorithms include the original IC [53], PC [47], K2 [11], BDe/BGe [21], GES [8], and CaMML [54].

Automated causal discovery is analogous to scientific discovery. It is typical for scientific discovery to aim at finding some unique theory to explain some data, only to find instead some larger set of theories, all of which can equally well explain the available data. Philosophers of science call this the problem of **underdetermination**: any finite set of data can be explained by some large

group of theories; adding some new data may well rule out a subset of the theories, while the set of theories remaining consistent with the expanded data is still large, perhaps infinitely large. Karl Popper [41] based his theory of scientific method on this observation, calling it Falsificationism: rather than verifying theories as true, he suggested that science progresses by falsifying theories that fail to accommodate new data; science progresses in a way vaguely similar to evolution, with unfit theories dying out and multiple fit theories remaining.²

Causal discovery can proceed in much the same way. We can begin with some hypothetical set of dags (the model space) that could provide causal explanations of reality and the real probability distribution, P_R , to be explained.³ The discovery process then repeatedly finds probabilistic dependencies in P_R , crossing off all of those dags which cannot model the dependencies. In a simplified form, the original Verma-Pearl IC algorithm can be expressed as:

1. **Step I.** Put an undirected link between any two variables X and Y if and only if

for every set of variables \mathbf{S} s.t. $X, Y \notin \mathbf{S}$

$$X \not\perp\!\!\!\perp Y | \mathbf{S}$$

I.e., X and Y are directly connected if and only if they are always conditionally dependent.

2. **Step II.** For every undirected structure $X - Z - Y$ (where X and Y are not themselves directly connected) orient the arcs $X \rightarrow Z \leftarrow Y$ if and only if

$$X \not\perp\!\!\!\perp Y | \mathbf{S}$$

for every \mathbf{S} s.t. $X, Y \notin \mathbf{S}$ and $Z \in \mathbf{S}$.

I.e., we have an uncovered collision if and only if the ends are always conditionally dependent upon the middle.

Following Steps I and II, a **Step III** checks for arc directions forced by further considerations, such as avoiding the introduction of cycles and uncovered collisions not revealed by P_R . Collectively, these steps suffice to determine what Verma and Pearl called a **pattern**, namely, a set of dags all of which share the same skeleton (arc structure, disregarding orientations) and uncovered collisions.

The IC algorithm is both the first causal discovery algorithm and the simplest to understand; it is not, however, practical in that it relies upon direct access to P_R via some oracle. Spirtes et al. [47] redesigned it to be more practical, in their PC algorithm, by finding some algorithmic efficiencies and, more importantly, by replacing the oracle with statistical significance tests for dependencies. Because

² In later years Popper made this analogy more explicit, developing what he called evolutionary epistemology [42].

³ Of course, we usually have access to P_R only through statistical samples; we are simplifying the description here.

of its simplicity, PC is now widely available, for example, in the machine learning toolbox, Weka [55]. An alternative approach to the discovery problem is to generate a global Bayesian or information-theoretic score for candidate models and then to search the model space attempting to optimize the score. Examples of this are the K2, BDe/BGe and CaMML algorithms cited above. The main difference between these two approaches, generally dubbed constraint-based and metric discovery respectively, is that the constraint learners test for dependencies in isolation, whereas the metric learners score the models based upon their overall ability to represent a pattern of dependencies.

Considerations arising from the Verma-Pearl algorithm led directly to a theory of the observational equivalence, or statistical indistinguishability, of models. It turns out that, given the assumption of a dependency oracle, the IC algorithm is in some sense optimal: Verma and Pearl proved that, if we restrict ourselves to observational data, no algorithm can improve upon its power to use the oracle's answers to determine causal structure. We can formalize (strong) statistical indistinguishability so:

Definition 2 (Strong Indistinguishability)

$$\forall \theta_1 \exists \theta_2 [P_{M_1(\theta_1)} = P_{M_2(\theta_2)}]$$

and vice versa

That is, any probability distribution representable by M_1 , via some parameterization θ_1 , is also representable by M_2 via some other parameterization, and vice versa. What Verma and Pearl [53] proved then is that two models are strongly indistinguishable if and only if they are in the same pattern. The set of patterns over the model space form a partition, with patterns being equivalence classes of dags. In consequence of these results, most researchers have focused upon the learning of patterns (or, equivalently, Markov equivalence classes), dismissing the idea of discovering the causal models themselves as quixotic. Thus, for example, the GES [8] stands for ‘‘Greedy Equivalence Search’’; it explicitly eschews any attempt to determine which dag within the best equivalence class of dags might be the true causal model.

By strong indistinguishability, the dags within each equivalence class exactly share the set of probability distributions which they are capable of representing. So, in other words, in the hands of most researchers, causal discovery as been turned from the search for causal models into, once again, the weaker and easier search for probability distributions. Of course, not attempting the impossible is generally good advice, so perhaps these researchers are simply to be commended for their good sense. If the problem of underdetermination is unsolvable, there is no point in attempting to solve it.

On the other hand, we might consider what human scientists do when observational data alone fail to help us, when multiple theories clash, but *not* over the observational data. Suppose, for example, we were confronted with a largely

isolated system that could equally well be one of these two causal models (which are in a common pattern):

$$\begin{aligned} \textit{Smoking} &\rightarrow \textit{Cancer} \\ \textit{Smoking} &\leftarrow \textit{Cancer} \end{aligned}$$

Underdetermination may halt scientific discovery, but not so near to the starting point as this! Commonly, of course, we would have more to go on than joint observations of these two variables. For example, we would know that *Smoking* generally precedes *Cancer* rather than the other way around. Or, we would expand our observations, looking, say, for mutagenic mechanisms articulating smoking and cancer. But, what would we do if we had no such background knowledge and our technology allowed for no such mechanistic search? We would *not* simply throw up our hands in despair and describe both hypotheses as equally good. We would experiment by intervening upon *Smoking* (disregarding ethical or practical issues).

Merely finding representations for our probability distributions does not exhaust our scientific ambitions, not by a long way. Probabilities suffice for prediction in the light of evidence (probabilistic updating); but if we wish to understand and explain our physical systems, or predict the impact of interventions upon them, nothing short of the causal model itself will do.

4.5 A Loss of Faith

Before exploring the value of interventional data for the discovery process, we shall consider another difficulty arising for causal modeling and for causal discovery in particular, namely a lack of faith. This has become one of the central arguments against the causal discovery research program.

The probabilistic causality research program in philosophy of science aims to find probabilistic criteria for causal claims [50, 46, 52]. The underlying intuition is that, contrary to much of the philosophical tradition, causal processes are fundamentally probabilistic, rather than deterministic. The probabilistic dependencies we measure by collecting sample observations over related variables are produced not merely by our ignorance but also, in some cases, directly by the causal processes under study. A variety of powerful arguments have been brought in support of this approach to understanding causality. Perhaps the most basic point is that our philosophy of science must at least *allow* the world to be indeterministic. The question is, after all, synthetic, rather than analytic: indeterminism is logically possible, so deciding whether our world is indeterministic requires empirical inquiry beyond any philosophizing we may do. Purely analytic philosophical argument cannot establish the truth of determinism. Turning to empirical means to resolve the question, we will immediately notice that our best fundamental theory of physics, quantum physics, is indeterministic, at least in its most direct interpretation.

If, then, causal processes give rise to probabilistic structure, we should be able to learn what causal processes are active in the world by an inverse inference from the probabilistic structures to the underlying causal structure. In

other words, the probabilistic causality theory underwrites the causal discovery research program. This is one reason why the causal discovery of Bayesian networks has attracted the attention of philosophers of science, both supporters and detractors of probabilistic causality.

Patrick Suppes' account of probabilistic causation begins with what he calls prima facie causation [50]. C is a **prima facie cause** of E if the event type of C is positively related to the event type of E (ignoring an additional temporal precedence requirement); i.e., those C such that $P(C|E) - P(C) > 0$. We are then to filter out spurious causes, which are any that can be screened off by a common ancestor of the prima facie cause and the purported effect. What remains are the genuine causes of E . The essence of this is the identification of potential causes by way of probabilistic dependence.⁴ Of course, this is also how the IC algorithm operates: both Step I and Step II posit causal structure to explain observed dependencies. And every other method of automated causal discovery also takes probabilistic dependence as its starting point; and they all assume that once all probabilistic dependencies have been causally explained, then there is no more work for causality to do. In other words, they assume the model to be discovered is *faithful*: corresponding to every d-connection in a causal model there must be a probabilistic dependence. This assumption is precisely what is wrong with causal discovery, according to Nancy Cartwright [4] and other skeptics.

In the simplest case, where $C \rightarrow E$ is the *only* causal process posited by a model, it is hard to see how a presupposition of faithfulness can be contested. Such a simple causal process which left no probabilistic reflection would be as supernatural as a probabilistic reflection produced by nothing at all. In any case, insisting upon the possibility of an unfaithful structure between understanding and reality *here* leaves inexplicable the ability to infer the causal structure in any circumstance.

But there are many situations where we should and must prefer an unfaithful model. One kind of unfaithfulness is where transitivity of probabilistic dependence fails. If causality is somehow based upon the kind of causal processes investigated by Salmon [45] and Dowe [12], processes which are capable of carrying information from one space-time region to another (Salmon-Dowe processes for short), then it seems somehow causality *ought* to be transitive. Salmon-Dowe processes are clearly composable: when ball A strikes B and B strikes C, the subprocesses composed form a larger process from A to C. No doubt this kind of Newtonian example lies behind the widespread intuition that causality must be transitive. We share the intuition that causal processes are somehow foundational for causal structure, and that they can be composed transitively; unfortunately for any simple analysis, causal structure itself is not transitive. One of many

⁴ Suppes, as do many other advocates of probabilistic causality, directs attention to causes which *raise* the probability of their effects and away from those which *lower* that probability — or, as Hausman [20] puts it, to the positive (promoting) *causal role* of the candidate cause. We are not much concerned with causal roles (whether promotion or inhibition) here, as such matters are primarily aimed at accounting for ordinary language behavior.

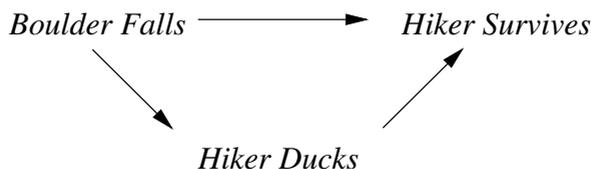


Fig. 4.3. The hiker surviving

examples from Hitchcock [23] will make this clear: suppose there is a hiker on a mountain side and at just the wrong time a boulder dislodges and comes flying towards her; however, observing the boulder, she ducks at the right moment, and the boulder sails harmlessly past; the hiker survives. This is represented graphically in Figure 4.3. We are to suppose that if the boulder dislodges, the hiker will duck and survive, and that if the boulder doesn't dislodge, she will again survive. In this case, there is no sense in which the boulder makes a difference to survival. It would be perverse to say that the boulder has caused the hiker to survive, or to generalize and assert that in this and relevantly similar cases falling boulders cause hikers to survive. While causal processes, and their inherent transitivity, are one part of the story of causality, making a difference, probabilistic dependence, is equally part of that story, a part which here fails dramatically.

Hiddleston [22] claims that intransitivity in all cases can be attributed to the fact that there are multiple causal paths; when we look at component causal effects in isolation, such things cannot happen. He is mistaken. An example of Richard Neapolitan [36] makes this clear: finesteride reduces DHT (a kind of testosterone) levels in rats; and low DHT can cause erectile dysfunction. However, finesteride doesn't reduce DHT levels sufficiently for erectile dysfunction to ensue (in at least one study); in other words, there is a threshold above which variations in DHT have no effect on dysfunction. Graphically, this is simply represented by:

$$\textit{Finesteride} \rightarrow \textit{DHT} \rightarrow \textit{Dysfunction}$$

Since, there is no probabilistic dependency between finesteride and erectile dysfunction, we have a failure of transitivity in a simple chain. We can equally have failures of transitivity in simple collisions [38].⁵

⁵ Hiddleston's mistake lies in an overly-simple account of causal power, for in linear models, and the small generalization thereof that Hiddleston addresses, causality is indeed transitive. Such models are incapable of representing threshold effects, as is required for the finesteride case.

We note also that some would claim that all cases of intransitive causation will be eliminated in some future state of scientific understanding: by further investigation of the causal mechanisms, and consequent increase in detail in the causal model, all apparently intransitive causal chains will turn into (sets of) transitive causal chains. This may or may not be so. Regardless, it is an a posteriori claim, and one which a theory of causality should not presuppose.

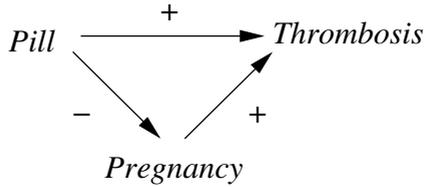


Fig. 4.4. Neutral Hesslow

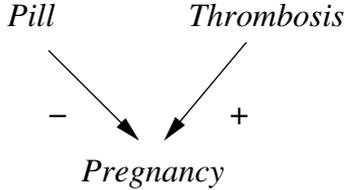


Fig. 4.5. Faithful Hesslow

Another kind of unfaithfulness does indeed arise by multiple paths, where *individual arcs* can fail the faithfulness test. These include “Simpson’s paradox” type cases, where two variables are directly related, but also indirectly related through a third variable. The difficulty is most easily seen in linear models, but generalizes to discrete models. Take a linear version of Hesslow’s example of the relation between the *Pill*, *Pregnancy* and *Thrombosis* (Figure 4.4). In particular, suppose that the causal strengths along the two paths from *Pill* to *Thrombosis* exactly balance, so that there is no net correlation between *Pill* and *Thrombosis*. And yet, the above model, by stipulation, is the true causal model. Well, in that case we have a failure of faithfulness, since we have a direct causal arc from *Pill* to *Thrombosis* without any correlation wanting to be explained by it. In fact, causal discovery algorithms in this case will generally not return Figure 4.4, but rather the simpler model (assuming no temporal information is provided!) of Figure 4.5. This simpler model has all and only the probabilistic dependencies of the original, given the scenario described.

A plausible response to this kind of example, the response of Spirtes et al. [48], is to point out that it depends upon a precise parameterization of the model. If the parameters were even slightly different, a non-zero correlation would result, and faithfulness would be saved. In measure theory (which provides the set-theoretic foundations for probability theory) such a circumstance is described as having *measure zero* — with the implication that the probability of this circumstance arising randomly is zero (see [48], Theorem 3.2). Accepting the possibility of zero-probability Simpson-type cases implies simply that we *can* go awry, that causal discovery is fallible. But no advocate of causal discovery can

reasonably be construed as having claimed infallibility.⁶ The question at issue cannot be the metaphysical claim that faithfulness *must* always be maintained — otherwise, how could we ever come to admit that it had been violated? Rather, it is a methodological proposal that, until we have good reason to come to doubt that we can find a faithful model, we should assume that we can. In the thrombosis case, with precisely counterbalancing paths, we begin with knowledge that the true model is not faithful, so we are obliged to abandon faithfulness.

Cartwright objects to this kind of saving maneuver. She claims that the “measure zero” cases are far more common than this argument suggests. In particular, she points out that many systems we wish to understand are artificial, rather than natural, and that in many of these we specifically want to cancel out deleterious effects. In such cases we can anticipate that the canceling out will be done by introducing third variables associated with both cause and effect, and so introducing *by design* a “measure-zero” case. In addition, there are many natural cases involving negative feedback where we might expect an equilibrium to be reached in which an approximate probabilistic independency is achieved. For example, suppose that in some community the use of sun screen is observed to be unrelated to skin cancer. Yet the possible causal explanation that sun screen is simply ineffective may be implausible. A more likely causal explanation could be that there is a feedback process such that the people using the sun screen expose themselves to more sunlight, since their skin takes longer to burn. If the people modulate their use of sun screen according to their exposure to the sun, then their total UV exposure would remain the same. Again, Steel [49] has pointed out that there are many cases of biological redundancy in DNA, such that if the allele at one locus is mutated, the genetic character will still be expressed due to a backup allele; in all such cases the mutation and the genetic expression will fail the faithfulness test. As Steel emphasizes, the point of all these cases is that the measure-zero premise fails to imply the probability zero conclusion: the system parameters have not been generated “at random” but as a result of intelligent or evolutionary design, leading to unfaithfulness.

If these cases posed some insurmountable burden for causal discovery algorithms, this would surely justify skepticism about automating causal discovery, for clearly we humans have no insurmountable difficulties in learning that the sun causes skin cancer, etc., even if these relations are also not easy to learn. However, it turns out there are equally clear, if again practically difficult, means available for machines to discover these same facts.

4.6 Intervention

So, it is time to see what interventions can do for causal discovery.

The concept of causal intervention and its uses in making sense of causal models have been receiving more attention recently, as, for example, in [56].

⁶ Cartwright notwithstanding: “Bayes-net methods. . . will bootstrap from facts about dependencies and independencies to causal hypotheses—and, claim the advocates, *never get it wrong*” [4, p. 254] (italics ours). Here, Cartwright’s straw-man has it wrong.

Indeed, both Pearl [40] and Spirtes et al. [48] treat intervention in some detail and provide a foundation for much of our work, and yet they have not applied interventions to resolve the problems raised by faithlessness and statistical indistinguishability. We now indicate how these are resolvable using interventions, first dealing with loss of faith. In empirical science, when observational data fail to differentiate between competing hypotheses, a likely response is to go beyond observation and experimentally intervene in nature: if we hold fixed known alternative causes of cancer and apply and withhold a candidate carcinogen to experimental and control groups respectively, we can be in a position to resolve the issue of what the true causal model is, whether or not it is faithful to observable dependency structures. Intervention and experiment would seem to have the power to resolve our conflict between truth, on the one hand, and simpler, faithless models, on the other.

In order to explore the epistemological power of intervention, we will consider a particular kind of intervention, with some ideal features. Much of the literature idealizes every possible feature (e.g., the do-calculus [40], and the manipulation theorem of [48]): interventions are themselves uncaused (they are root nodes in a causal model), and so multiple interventions are uncorrelated; interventions impact upon exactly one variable in the original model; interventions are deterministic, definitely resulting in the intervened upon variable adopting a unique state. Such extreme idealization is not a promising starting point for a *general* theory of causal modeling, and the actual interventions available to us often fall well short of the ideal (cf. [27]). For our purposes here, however, we shall adopt all of them except the last: by default our interventions influence their target variables, but do not to the extreme of cutting off all influence of their prior parents (but, should they do so, we shall indicate this by calling the interventions *perfect*).⁷ The extreme, perfect interventions can be represented in Bayesian networks simply by setting the target variable to the desired state and cutting all of its inbound arcs. Our less perfect interventions cannot be so represented; existing parents retain their arcs to the target variable. So, the natural representation of our interventions is by augmenting our dags with new intervention variables.

It is worth noting, as an aside, that this approach makes clear the difference between intervention and observation. Since most existing tools *only* provide explicit means for probabilistic updating under observation,⁸ some have attempted to understand the effect of causal interventions upon a variable X simply by setting X to some desired value in a Bayesian network and updating. Such updates, however, can radically mis-estimate the effects of interventions. For example, consider Figure 4.6. *HT40* describes blood pressure (hypertension) at age 40, *HT50* the same at age 50 and *CHD50+* the coronary status of a subject from ages 50 to 60. We may be contemplating an intervention to reduce blood pressure at age 50, which may well reduce the chances of a heart attack over then next ten years. If, however, we were hypothetically to *observe* the same reduced level of

⁷ We will assume that our interventions have *some* positive probability of affecting the target variable; indeed, we shall assume the same for every parent variable.

⁸ For a description of a tool, *The Causal Reckoner*, that does more, see [27].

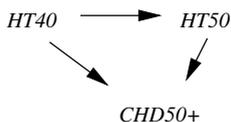


Fig. 4.6. Hypertension and coronary heart disease

blood pressure at age 50 as we might achieve by intervention, the resulting risk of heart attack would be even lower. Technically, that is because there are two d-connecting paths from $HT50$ to $CHD50+$, whereas with a perfect intervention the link from $HT40$ to $HT50$ is cut, so there is only one path from $HT50$ to $CHD50+$.⁹ What this means non-technically is that should we observe someone at 50 with lower blood pressure, that implies the she or he also had lower blood pressure at age 40, which, entirely independently of blood pressure at 50, has an effect on $CHD50+$; whereas if we *intervene* at 50 we are not thereby gaining any new information about $HT40$. In short, in order to determine the effects of intervention, simply using observations and updating with standard Bayesian net tools is radically wrong.

So, we prefer to represent interventions quite explicitly, by putting new intervention variables into our graphs. Furthermore, in order to test the limits of what we can learn from intervention we consider *fully* augmented models, meaning those where *every* original variable X gets a new intervention parent I_X , doubling the number of variables. In the case of the two Hesslow models, faithful (true) and faithful (false), full augmentation results in Figure 4.7.

What we suggest, then, is that the argument over whether faithfulness is an acceptable assumption for causal discovery is simply misdirected, ignoring the power of interventions. Faithfulness is not the issue; the real issue is **admissibility under augmentation**: A causal model M is admissible under augmentation if and only if its fully augmented model M' is capable of representing the system's fully augmented probability distribution.

Spirtes, Glymour and Scheines (SGS) [48] proved a theorem about augmentation which already suggests the value of interventions for distinguishing between causal models:

SGS Theorem 4.6. *No two distinct causal models that are strongly indistinguishable remain so under intervention.*¹⁰

This has a trivial corollary:

Corollary 1. *Under intervention, no two distinct causal models are strongly indistinguishable.*

⁹ And with an imperfect intervention, although the link from $HT40$ to $HT50$ is not cut, the influence of $HT40$ on $CHD50+$ is reduced.

¹⁰ Note that where we write of augmenting models with intervention variables, Sprites et al. [48] talk about “rigid indistinguishability”, which amounts to the same thing.

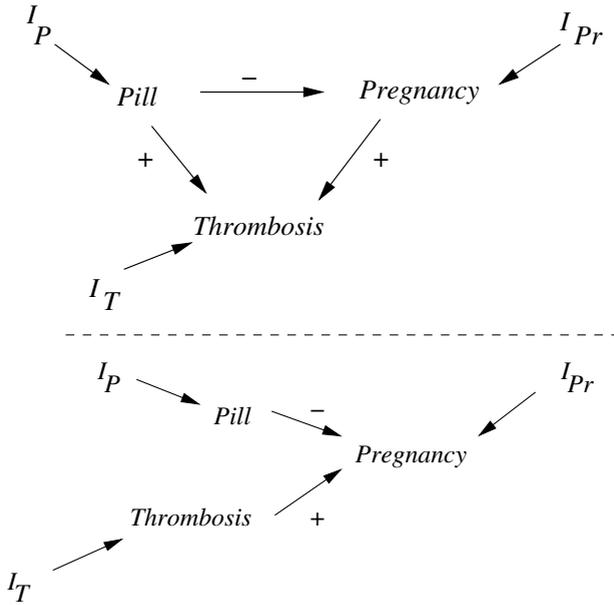


Fig. 4.7. Augmented Hesslow models: the faithless, but true, causal model (top); the faithful, but false, model (bottom)

These results tell us that when we augment models, even strongly indistinguishable models, we can then find *some* probability distribution that will distinguish them. Unfortunately, this does not provide much practical guidance. In particular, it tells us nothing about the value of intervention when we are dealing with a specific physical system and its *actual* probability distribution, as we typically are in science. But we can also apply intervention and augmentation to answering questions about particular physical systems rather than the set of all possible physical systems.

First, we can show that the neutral Hesslow system M_1 of Figure 4.4 can be distinguished from its faithless imposter M_2 of Figure 4.5. Under these circumstances, the probability distributions asserted by these models are identical; i.e., $P_{M_1(\theta_1)} = P_{M_2(\theta_2)}$. Then:

Theorem 2 (Distinguishability under Imperfect Intervention; [30])

If $P_{M_1(\theta_1)} = P_{M_2(\theta_2)}$, then under imperfect interventions $P_{M_1(\theta_1)} \neq P_{M_2(\theta_2)}$ (where M_1 and M_2 are the respective structures of Figures 4.4 and 4.5).

The proof is in [30] and is related to that of SGS Theorem 4.6 (we also proved there the same result for perfect interventions). The notable difference is that we begin with a particular probability distribution, that of the true Hesslow model,

and a particular inability to distinguish two models using that distribution.¹¹ Using Wright's path modeling rules [57] we are able to find the needed difference between the augmented models, when there is none in their unaugmented originals. This difference arises from the introduction of new uncovered collisions under augmentation: every intervention variable induces a collision with its target variable and any pre-existing parent variable. As the Verma-Pearl algorithm is sensitive to such dependency structures (and likewise every other causal discovery algorithm), the dependency structures of M_1 and M_2 , while parameterized to be identical under observation, cannot be so parameterized under intervention.

Although this theorem is particular to cases isomorphic to the neutral (and linear) Hesslow case, the result is of much wider interest than that suggests, since the neutral Hesslow structure is the only way in which a true linear causal model can be unfaithful to the probability distribution which it generates, through balancing multiple causal paths of influence. Perhaps of more interest will be interventions upon discrete causal models, which are more commonly the center of attention in causal discovery and which certainly introduce more complexity than do linear models. In particular, in non-linear cases there are many more opportunities for intransitivities to arise, both across multiple paths and, unlike linear models, across isolated paths. Nevertheless, similar theoretical results are available for discrete causal models, although, as their description and proof are more complicated, we skip them here (see instead [38]).

But the power of intervention to reveal causal structure goes well beyond these preliminary theorems [29]. It turns out that a comprehensive regime of interventions has the power to eliminate all but the true causal model from consideration. In other words, despite the fact that strong indistinguishability can leave a very large number of causal models equally capable of accommodating observational data, interventional data can guarantee the elimination of all models but one, the truth (this is the combined effect of Theorems 5 and 6 of [29]). Again, these results are generalizable to discrete models, with some additional restrictions upon the interventions needed to cope with the more complex environment [38]. Current research, at both Carnegie Mellon University and Monash University, is aimed at determining optimal approaches to gathering interventional data to aid the causal discovery process. None of these results are really surprising: they are, after all, inspired by the observation that human scientists dig beneath the readily available observations by similar means.

Neither the presumption of faithfulness by causal discovery algorithms nor their inability to penetrate beneath the surface of strong statistical indistinguishability offer any reason to dismiss the causal interpretation of Bayesian networks nor their automated discovery. The common view to the contrary is plausible only when ignoring the possibility of extending causal modeling and causal discovery by intervening in physical systems. Even if such interventions

¹¹ In SGS terminology, this is an application of the idea of rigid distinguishability to models which are weakly indistinguishable, that is, having some probability distribution which they can both represent.

are expensive, or even presently technologically impossible, that is no principled reason for rejecting causal interpretation. The history of human science is replete with examples of competing theories remaining indistinguishable for centuries, or millenia, before technological advances have decisively found in favor of one over another. Witness Copernican and Ptolemaic astronomy, continental drift versus static geological theory, evolution theory versus static species theory. All of these are decisively settled today. In no case was the truth of no interest prior to the technological advances which allowed their settlement: on the contrary, interest in settling the issues has typically driven those very technological advances.

4.7 Causal Explanation

So far, we find that the causal interpretation of Bayesian nets is a reasonable one. At least, the arguments thrown up against it have failed to show otherwise, which is less than a clear justification for adopting a causal interpretation, but more than nothing. A principled justification would perhaps rely upon a serious exploration of the metaphysics of causation. We shall not attempt that, although in the next sections we will direct interested readers to some of the relevant current discussions in the philosophy of causation. A pragmatic justification is more to our taste: having dispelled the known objections, the success of causal modeling and causal discovery in developing an application technology for AI that is usable and used, and which presupposes the causal interpretation, should settle the argument. Norsys Corp., developer of the Bayesian network tool *Netica*, lists a considerable variety of real-world applications:

- *Agricultural Yield* for predicting the results of agricultural interventions [3]
- *PROCAM* model for predicting coronary heart disease risk [51]
- *Wildlife Viability* for predicting species viability in response to resource management decisions [34]
- *Risk Assessment Fusion* combining multiple expert opinions into one risk assessment [1]

There is much more work that can be done in making sense of Bayesian networks in causal terms. One such effort is to provide an account of the explanatory power of a cause for some effect. For example, to what extent can we attribute the current prevalence of lung cancer to smoking? This question about type causality can also be particularized in a question about token causality, as in: Was Rolah McCabe's terminal lung cancer due to her smoking cigarettes produced by British American Tobacco? We defer such questions about token causation to the section §4.9.

The type causality question is an old issue in the philosophy of science; what makes it also a new issue is the prospect of tying such an account to Bayesian networks, so that we might have computer assisted causal reasoning. Such an attempt is already on offer in the work of Cheng and Glymour [6, 15]. They define a concept of *causal power* for binomial variables, which, in the case of

causes which promote their effects (rather than inhibit them, which takes a different equation), is:

$$p_c = \frac{P(e|c) - P(e|\neg c)}{1 - P(e|\neg c)}$$

The numerator corresponds to the probabilistic dependence used as a criterion by Suppes of prima facie causation. Cheng and Glymour get the effect of Suppes' filtering out of spurious cases of causation by imposing structural restrictions on the Bayesian networks which source the probabilities required to define p_c . So, the result is a causal power measure for genuine causation which is proportional to positive probabilistic dependence, relative to the background frequency of $\neg e$ when the cause is absent.

This is a plausible start on explanatory power in Bayesian networks; unfortunately there is no finish. Cheng and Glymour's definition is restricted to binomial networks with parent variables which fail to interact (as, for example, an XOR interacts).¹² Furthermore, as Glymour [14] notes, the restrictions they impose on allowable Bayesian networks are equivalent to requiring them to be noisy-OR networks. The result is that for these, and also for the extension of Hiddleston [22], causal power is *transitive*, whereas we have already seen in Section §4.5 that probabilistic dependence over causal links (and, hence, a key ingredient for any useful definition of causal power) is *intransitive*. A different account is wanted.

We prefer to identify causal power with an information-theoretic measure related to mutual information [25]. The mutual information of C for E (or vice versa) is a measure of how much you learn about one variable when the other variable is observed. Hope and Korb's causal power differs from mutual information per se in a number of ways. First, it is asymmetric; it is required that it be attributed to the cause, rather than the effect, according to the ancestral relations indicated by the causal model. Also, the relevant probabilities are relativized to a context of interest. That is, any causal question is raised in some context and a measure of causal power needs to be sensitive to that context. For example, the causal power of smoking for lung cancer may be fairly high in some populations, but it is non-existent amongst those who already suffer from lung cancer. Finally, the causal power of C is measured according to a hypothetical perfect intervention upon C and not based upon mutual information computed by observations of C . Thus, if Sir Ronald Fisher's [13] speculative defence of smoking were actually true, i.e., that the true model were

$$Smoking \leftarrow Gene \rightarrow Cancer$$

rather than

$$Smoking \rightarrow Cancer$$

then the causal power of *Smoking* for *Cancer* would be nil, whereas the mutual information between the two is unaffected by the change of structure.

¹² However, Novick and Cheng [37] relax this restriction to some extent by considering pairwise parental interactions.

This measure of causal power takes full advantage of Bayesian networks: all relevant interactions between causal variables are automatically taken into account, since the computation of the information-theoretic measure depends upon the underlying computations of the Bayesian network. As a result, for example, all cases of intransitivity found by Hitchcock and others yield end-to-end causal powers of zero, as is desirable.

This causal power theory also shows considerable promise for answering a pressing practical need in the application of Bayesian network technology, namely making the networks easier to interpret. By providing a means to query any such network about the implications of proposed interventions (of any type) it is no longer necessary for users themselves to follow and account for causal influences across multiple paths. Furthermore, by separating questions of causal power from those of probabilistic updating we can reduce the temptation to attempt to find causal explanations using tools which only answer questions about probabilistic updating, confusing any causal story rather than illuminating it.

The concept of causal power is a necessary ingredient for a full account of type causality, i.e., causal relations between types of events, rather than particular evens (token causality, treated in section §4.9). In some sense, causal Bayesian networks without any variables instantiated provide as full an account of the type causal relations between its variables as could be imagined (assuming they are true models of reality, of course). However, there remains analytical work to do beyond such an observation. In Hesslow's neutral model, for example, does the *Pill* cause *Thrombosis*? The net effect — the net causal power — is, as we have noted, nil. However, type causal questions are typically aimed at determining whether there is *some* way (consistent with any explicitly provided context) for the cause to have an impact on the effect. In order to answer such a question, (type) causal paths need to be considered in isolation, for example the *Pill* \rightarrow *Thrombosis* path isolated from the path through *Pregnancy*, by fixing the value of the latter variable [23].¹³ The type causal question can be answered affirmatively if any such isolated path has a non-zero causal power. Of course, we are commonly interested also in knowing *how important* a cause may be: for that we need the non-zero causal power itself.

4.8 Causal Processes

Now we will consider how Bayesian networks can help us make sense of token causality: claims about the particular responsibility of particular causal happenings for particular outcomes. We begin by sketching some relevant philosophical background, for we believe the philosophy of causal processes is now needed.

There are two distinct approaches which in recent times have dominated attempts to come to grips with the notion of causality. One is the probabilistic

¹³ In addition to that, type causal questions need to be relativized to an objectively homogeneous context, so that the causal power being computed is *not* an average of disparate powers in distinct contexts, as Cartwright [5] has effectively argued (see also [52]).

causality research program, already introduced. The other is the attempt to locate a supervenience base for causal relationships in an underlying metaphysics of process, initiated by Salmon [45] and furthered by Dowe [12]. Processes are contiguous regions of space extended through some time interval — i.e., spacetime “worms”. Of course, they can’t be just any such slice of spacetime; most such slices are causal junk [26]. The Salmon-Dowe research program is largely aimed at coming up with clear criteria that rule out junk, but rule in processes which can sustain causal relationships. Intuitively, we can say legitimate processes are those which can carry information from one spacetime region to another (“mark transmission” is what Salmon called this; Dowe calls it “conserving physical quantities”). Examples are ordinary objects (balls carry around their scratches) and ordinary processes (recipes carry their mistakes through to the end). Non-examples are pseudo-processes and pseudo-objects (e.g., Platonic objects, shadows, the Void of Lewis [33]). Hitchcock [24] rightly points out that, thus far, this account leaves the metaphysics of causal processes unclear. The Salmon-Dowe research program is incomplete. But we know of no reason to believe it is not completable, so for the purposes of our discussion we shall describe as “Salmon-Dowe processes” those which fall under some future completed analysis of this type.

If it is causal processes which ground the probabilistic dependencies between variables, then it must be possible to put the variables within a single model into relation with one another via such processes. This suggests a natural criterion of relevance to require of variables within a single causal model: namely, if two variables appear in a causal model, there must be a sequence of possible or actual causal processes connecting them. This makes precise Hitchcock [23], who vaguely requires that pairs of variables not be “too remote” from each other. Note that we do not demand a possible sequence of causal processes between any two variables, but a sequence of possible processes: it may be, for example, that two events are spacewise separated, yet mediated by a common third event. Nor, of course, do we demand *actual* processes between any event types in the model. Probabilistic dependency is founded upon possibilities, realized and unrealized.¹⁴

The two approaches to understanding causality, dependency and process, have disparate strengths and weaknesses. This disparity has led many to suggest that there is no one concept of causality and that attempts to provide a unified account are confused.¹⁵ While we agree that there may well be various distinct concepts of causality, we are unconvinced that the particular disparity between the dependence and process analyses argues for two concepts of causality. Instead,

¹⁴ That there are causal processes behind the arcs of causal models suggests the answer to one of the concerns about causal modeling that Nancy Cartwright raises, namely that causal reality may not be made up of discrete token events, but perhaps continuous processes instead [4]. Well, we expect that reality is made up of token processes, whether discrete or continuous. Discrete Bayesian networks are a convenient way of modeling them, and the variables we choose are convenient and useful abstractions. They need to be tied to the underlying reality in certain ways, but they certainly do not need to be exhaustive descriptions of that reality.

¹⁵ Hitchcock has suggested this, e.g., in Hitchcock (2004a, b); see also [16].

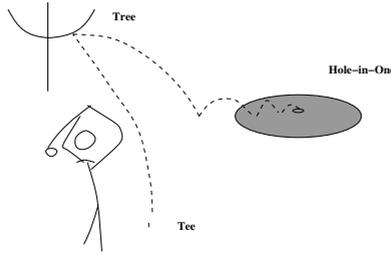


Fig. 4.8. Rosen's hole-in-one

we propose a causal unification program: that we develop a unifying account that uses the strengths of the one to combat the weaknesses of the other.

Dependency accounts characteristically have difficulties dealing with negative relevance, that is, causes (promoters, in role language) which in some token cases are negatively relevant to the effect (i.e., prevent it), or vice versa. Deborah Rosen [44] produced a nice example of this in response to Suppes [50]. In Figure 4.8 Rosen has struck a hole-in-one, but in an abnormal way. In particular, by hooking into the tree, she has *lowered* her chance of holing the ball, and yet this very chance-lowering event is the proximal cause of her getting the hole-in-one. The only hope of salvaging probability-raising here, something which all of the dependency accounts mentioned above wanted, is to refine the reference class from that of simply striking the tree to something like striking the tree with a particular spin, momentum, with the tree surface at some exact angle, with such-and-such wind conditions, etc. But the idea that we can always refine this reference class in enough detail to recover a chance-raising reference class is far-fetched. It is what Salmon [46] described as *pseudo-deterministic faith*.¹⁶ In any case, as Salmon also pointed out, we can always generate chance-lowering causes in games, or find them in quantum-mechanical scenarios, where there is no option for refinement. Take Salmon's "cascade" [46], where a quantum-mechanical system undergoes state changes with the probabilities indicated in Figure 4.9. Every time such a system reaches state d via state c rather than state b, it has done so through a chance-lowering event of transition into state c. By construction (in case this is a game, by nature otherwise) there is no refinement of the intermediate state which would make the final transition to state d more probable than the untaken alternative path through b; hence, probability-raising alone cannot account for causality here.

¹⁶ Note that the escape by contrasting striking the tree with missing it fails on at least two counts. Of course, missing the tree, given the hook, is a contrast class with a *lower* probability of success than hitting the tree. But we are attempting to understand causality *relative* to a given causal model. And this maneuver introduces a new variable, namely *how* the ball is hit (or, perhaps, its general direction), so the maneuver is strictly evasive. Secondly, if we are going to countenance new variables, we can just introduce a local rule for this hole: just behind the tree is a large net; landing in the net also counts as holing the ball.

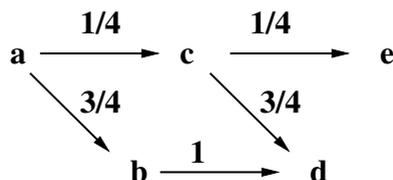


Fig. 4.9. Salmon's quantum-mechanical cascade

Salmon's way out was to bite the bullet: he asserted that the best we can do in such cases is to locate the event we wish to explain causally (transition to d) in the causal nexus.¹⁷ If the transition to d occurs through b, then everyone is happy to make the causal attribution; but if it has occurred through c, then it is no less caused, it is simply caused in a less probable way. Insisting on the *universal* availability of promoting causes (probability raising) is tantamount to the pseudo-deterministic faith he denounced [45, chapter 4]. Instead of reliance upon the universal availability of promoters, Salmon asked us to rely upon the universal availability of Salmon-Dowe causal processes leading from each state to the next. This seems the only available move for retaining irreducible state transitions within the causal order.

Assuming, per above, that the metaphysics of process has been completed, the problem remains for Salmon's move that it is insufficient. For one thing, as we saw above, causal processes are composable: if we can carry information from one end to the other along two processes, then if we connect the processes, we can carry the (or, at any rate, some) information from the composite beginning to the composite end. But the many cases of end-to-end probabilistic independency need to be accommodated; the possibility of causal intransitivity needs to be compatible with our criteria of causality. Hence, invoking causal processes cannot suffice.

A pure causal process account has other problems as well. Whereas probability raising clearly itself is too strong a criterion, missing minimally every least probable outcome within a non-trivial range of outcomes, simply invoking causal process is clearly too weak a criterion. In some sense the Holists are right that everything is connected to everything else; at any rate, everything within a lightcone of something else is likely to have a causal process or potential process relating the two. But while it makes sense to assert that the sun causes skin cancer, it makes little sense to say that the sun causes recovery from skin cancer. Yet from the sun stream causal processes to all such events, indeed to every event on earth. Salmon's account of 1984 lacked distinction.

It is only in adding back probabilistic dependencies that we can find the lacking distinction. Positive dependencies, of course, have difficulties dealing with negative relevance; processes do not. Processes alone cannot distinguish relevant from irrelevant connections; probabilistic dependencies can. Plausibly what is

¹⁷ In Salmon's terms, within an objectively homogeneous reference class.

wanted is an account of causal relevance in terms of processes-which-make-a-relevant-probabilistic-difference. Two accounts which provide this are those of Menzies [35] and Twardy and Korb [52]. What we will do here, however, is apply these two concepts of causal process and difference making to making sense of causal responsibility.

4.9 Causal Responsibility

Many of the more pressing questions that arise about causality concern questions of responsibility, legal or moral. These are questions about particular facts, that is, particular events and their particular causal relationships. Or, in other words, questions about token causality (“actual causality”) rather than type causality.¹⁸ Incidentally, much of the philosophical literature on causality focuses on token causality and stories told about token causality; we shall treat some of them here.

It has always been clear that type and token causality, while distinct, are related, but the relationship is itself not clear. Causal modeling with Bayesian networks provides an opportunity for getting that relationship clear, by providing an opportunity to establish criteria for both based upon the same model. Our analysis aims at putting type-token causality into a kind of general-to-particular relationship. And what we will do here is to outline a plausible account of token causality, one that needs work to be complete, but appears to us to be aimed in the right direction.

Our treatment is based upon a presumption that we have in hand the right causal model. That is, what our analysis asserts is, or is not, a cause of some particular event, depends upon the causal model assumed to be true; the token causality analysis itself does not provide guidance in finding that true model. We will instead rely upon certain principles of model building which arise from the prior discussion, although we do not defend them explicitly (for a more complete defence see [30]):

Principle 1 (Intervention): *Variables in a causal model must be intervenable.*

Principle 2 (Distinction): *Every pair of variables in a causal model must have a physically possible intervention which, in some physically possible context, affects the distribution of one variable without affecting that of the other.*

Principle 3 (Process): *If two variables appear in a causal model, there must be a sequence of possible or actual causal processes connecting them.*

The first efforts to provide a Bayesian-net based analysis of token causality were those of Hitchcock [23] and Halpern and Pearl [17, 18]. Hitchcock’s is a simplification of [17], which is arguably superior in some ways but more complex

¹⁸ To be sure, any satisfying account of either legal or moral responsibility goes beyond a satisfying account of token causality alone, since it will have to incorporate a treatment of legal or moral principles and their application to causal questions. We will not enter into such matters here.

than we care to deal with here. The difficulties we will point out with Hitchcock’s treatment carry through transitively to that of Halpern and Pearl.

Consider again the case of Hitchcock’s hiker (Figure 4.3). Clearly, what we want to say is that boulders do cause death in such circumstances, if only because human responses are fallible, so the type relations are right in that model — each arc corresponds to a type causal relation that manifests itself in a probabilistic dependency under the right circumstances.¹⁹ But in the particular case — to be sure, idealistically (deterministically) described — the boulder’s fall does not affect survival in any way, because there is no probabilistic dependency between the two.

Hitchcock [23] describes two plausible criteria for token (actual) causality. Both of them look at component effects, by isolating some causal path of interest. The first is very simple. Let’s call it **H1** (following [22]).

H1: $C = c$ **actually caused** $E = e$ if and only if both $C = c$ and $E = e$ occurred and when we iterate through all $\Phi_i \in \text{Paths}(C, E)$, for some such Φ_i if we block all the alternative paths by fixing them at their actually observed values, there is a probabilistic dependence between C and E .

In application to Hitchcock’s hiker’s survival, this works perfectly. Considering the direct path *Boulder* \rightarrow *Survival*, we must fix *Duck* at true, when there is no probabilistic dependency. The second path (through *Duck*) doesn’t need to be considered, since there is no variable mediating the path alternative to it, so there is no question of blocking it.

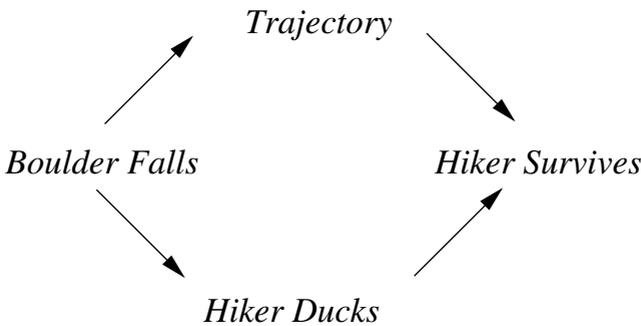


Fig. 4.10. The hiker surviving some more

The second path could, of course, be considered if we embed the model of Figure 4.3 in a larger model with a variable that mediates *Boulder* and *Survival*. We could model the trajectory of the boulder, giving us Figure 4.10. In this case, H1 gets the wrong answer, since we are now obliged to fix *Trajectory* and

¹⁹ Of course, we would never say “Boulders falling cause survival.” But that’s because in our speech acts causal role ordinarily leaks into causal attributions. We are not here interested in a theory of ordinary language utterances about causality.

discover that there is now a probabilistic dependency between *Boulder* and *Survival*. In particular, if the boulder *doesn't* fall, but somehow regardless achieves its original trajectory, then the hiker won't have ducked and will end up dead. Hitchcock's response to this possibility is to say that the introduction of *Trajectory* requires a "sophisticated philosophical imagination" — we have to be able to imagine the boulder miraculously appearing on collision course without any of the usual preliminaries — and so an account of actual causation for the ordinary world needn't be concerned with it. Hiddleston objects to this as an ad hoc maneuver: he suspects that variables will be called miraculous when and only when they cause trouble for our analysis. However, he is mistaken. Our Intervention Principle makes perfectly good sense of Hitchcock's response. Either *Trajectory* is intervenable (independently of *Boulder*) or it is not. If it is not, then modeling it is a mistake, and H1's verdict in that case is irrelevant. If it is intervenable, then there must be a possible causal process for manipulating its value. A possible example would be: build a shunt aimed at the hiker through which we can let fly another boulder. For the purposes of the story, we can keep it camouflaged, so the hiker has no chance to react to it. All of this is possible, or near enough. But in order to introduce this variable, and render it intervenable, we have to make the original story unrecognizable. Hitchcock's criterion, just as much as ours to follow, is model-relative. The fact that it gives different answers to different models is unsurprising; the only relevant question is what answer it gives to the right model.

This reveals some of the useful work our model-building principles do in accounting for actual causation, even before considering the details of any explicit criterion.

H1 handles a variety of examples without difficulty. For example, it copes with the ordinary cases of pre-emption which cause problems for dependency theories. Thus, in Figure 4.11 if a supervisor assassin fires at the victim if and only if the trainee assassin doesn't fire and, idealistically again, neither the trainee nor supervisor can miss, then an account requiring end-to-end dependency, such as Lewis's original counterfactual analysis of causation [31], fails. In particular, should the trainee fire, this action will not be considered the cause of the victim's death, since there is no dependency. In the face of such counterexamples, Lewis adopted a step-wise dependency of states of the bullet as it traverses the distance to the victim. Although there is no end-to-end dependency, if we take the transitive closure of step-by-step dependencies, we find end-to-end causation. We find this objectionable on two counts: first, as we have seen, causation is not transitive; second, finding the intermediate dependencies requires generating intermediate variables, and so altering the causal story in unacceptable ways. Hitchcock's H1, on the other hand, has it easy here: we simply observe that the supervisor did not fire and that under this circumstance there is a dependency between the trainee's action and the victim's health.

This is an example of pre-emption by "early cutting". Pre-emption can also occur through late cutting. If Billy and Suzy are each throwing a rock at a bottle (and, as usual, they cannot miss) and if Suzy throws slightly earlier than Billy,

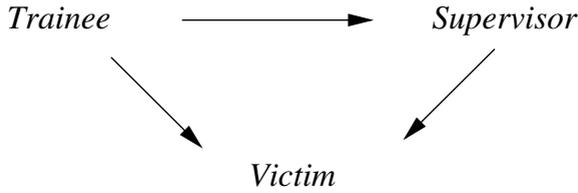


Fig. 4.11. Pre-emptive assassination

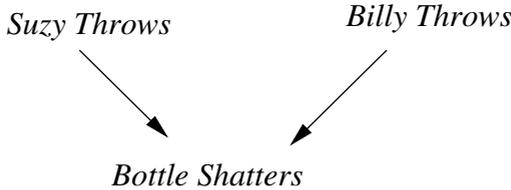


Fig. 4.12. Pre-emptive bottle smashing

then Suzy causes the bottle to shatter and Billy does not (see Figure 4.12), again despite the fact that there is no end-to-end dependency. In this case, however, there is also no step-wise dependency for Lewis to draw upon: at the very last step, where the bottle shatters, the dependency will always fail, because Billy’s rock is on its way.

Hitchcock’s H1 fails to accommodate Suzy’s throw, because the end-to-end dependency fails under the actual circumstances. So, Hitchcock resorts to counterfactual contexts to cope (in this he is following the account of Halpern and Pearl [17]).²⁰ For these contexts Hitchcock only allows counterfactual circumstances which would not change the values of any of the variables on the causal path under consideration. Any variable off that path will have a range of values which have no impact on the causal path, minimally that value which it actually took. Such values are said to be in the **redundancy range** (RR) for that path. Then the new criterion, **H2**, is:

H2: $C = c$ **actually caused** $E = e$ if and only if both $C = c$ and $E = e$ occurred and when we iterate through all $\Phi_i \in \text{Paths}(C, E)$, for some such Φ_i there is a set of variables \mathbf{W} s.t., when fixed at values in their redundancy ranges relative to Φ_i , there is a probabilistic dependence between C and E .

Since actual values are trivially within the RR, the prior (positive) successes of H1 remain successes for H2. With Suzy and Billy, it’s clear that Billy’s throwing or not are both within the redundancy range, and the dependency upon Suzy’s

²⁰ Lewis [32] also resorted to counterfactuality, replacing sequences of dependencies with sequences of hypothetical dependencies (“quasi-dependencies”). Incidentally, Halpern and Pearl [17] analyse this case using a temporally expanded (dynamic) network; however, the complexities involved would not reward our treating it here.

throw reappears when we consider what happens when Billy’s throw is absent. This seems a very tidy solution.

However, Hiddleston [22] offers an example which H2 cannot handle, as usual an example concerning potential violent death. Suppose the king’s guard, fearing an attack, pours an antidote to poison in the king’s coffee. The assassin, however, fails to make an appearance; there is no poison in the coffee. The king drinks his coffee and survives. Did the antidote cause the king to survive? That is no more plausible than the claim that the boulder falling has caused the hiker to survive; however, H2 makes this claim, since *Poison* being true is in the redundancy range.²¹ Interestingly, H1 gets this story right, since the poison is then forced to be absent, when the dependency of survival on antidote goes away. Hiddleston concludes that H1 was just the right criterion all along, but needed to be supplemented with Patricia Cheng’s (and Clark Glymour’s) theory of causal models and process theory [6, 15, 14]. We agree with his general idea: examining dependencies under actual instantiations of context variables is the right way to approach actual causality. Cheng’s causal model theory, however, is far too restrictive, as we noted above.

4.9.1 An Algorithm for Assessing Token Causation

So, we now present our alternative account of actual causation in the form of an “algorithm” for assessing whether $C = c$ actually caused $E = e$, given that both events occurred. Our steps are hardly computationally primitive, but opportunities for refining them and making them clearer are surely available.

Step 1

Build the right causal model M .

Of course, this is a complicated step, possibly involving causal discovery, expert consultation, advancing the science of relevant domains, and so forth. All of our model-building rules apply. This (hopefully) leads to the right causal model, describing the right type causal relationships. So, this account of token causality starts from the type causal model and gets more specific from there.

This step, applying the model-building principles, circumvents a number of problems that have arisen in the literature. For example, we know that *Time* should not be invoked as a variable (it violates the Intervention Principle) and that problem-defining constraints should likewise be excluded (because of the Distinction Principle). We also know that the imaginative introduction of intermediate events to save some kind of step-wise dependency across a causal chain is (normally) illegitimate. So, despite being open-ended, this “step” is not vacuous.

²¹ It might be pointed out that the model here is incomplete, and an intermediate node which registers the combined state of poison and antidote would push *Poison=true* out of the redundancy range. But that’s an ineffective response, unless in fact *no* model of the structure offered by Hiddleston is possible. However, we can always construct such a model, as a game, for example.

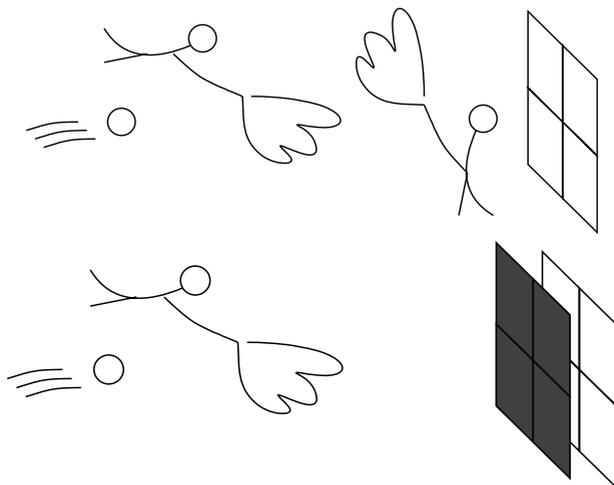


Fig. 4.13. Backup fielders, imperfect and perfect

Step 2

Select and instantiate an actual context O (we will designate the model M in context O by M/O).

Typically, this involves selecting a set O of variables in M and fixing them at their observed values. Often the causal question itself sets the context for us, when selecting the context is trivial. For example, someone might ask, “Given that no poison was added to the coffee, did the antidote cause the king’s survival?” Indeed, that appears to be exactly what Hiddleston was asking in the above example.

A striking example of context-setting is this (from [9]). Suzy and Billy are playing left and center field in a baseball game. The batter hits a long drive between the two, directly at a kitchen window. Suzy races for the ball. Billy races for the ball. Suzy’s in front and Billy’s behind. Suzy catches the ball. Did Suzy prevent the window from being smashed? Not a hard question. However, now suppose we replace Billy with a movable metal wall, the size of a house. We position the wall in front of the window. The ball is hit, Suzy races and catches it in front of the wall. Did Suzy prevent the window from being smashed? This is no harder than the first question, but produces the opposite answer. The question, of course, is how can our criterion for actual causation reproduce this switch, when the two cases are such close parallels of each other. What changes is the context in which the causal question gets asked. Here the changed context is not reflected in the values of the variables which get observed — neither the wall nor Billy are allowed to stop the ball; rather, the changed context is in

the probability structure.²² Billy is a fallible outfielder; the wall, for practical purposes, is an infallible outfielder. The infallibility of the wall leaves no possible connecting process between Suzy's fielding and the window.²³

The idea of a connecting process arose already in the Process Principle, in that there must be a sequence of possible connecting processes between any two variables in the model. Specifically, by **connecting process** we mean any Salmon-Dowe process such that under the relevant circumstances (e.g., O) C makes a probabilistic difference to E . We suggest beyond the Process Principle a somewhat more demanding model building principle for individual arcs is in order:

Principle 4 (Connecting Process): *For every arc $C \rightarrow E$ in M/O there must be a possible value for C such that there is a connecting process between C and E .*

The baseball example is a case where an *individual* arc fails to have a corresponding connecting process for any value of its causal variable. Such arcs we call **wounded** and remove them:²⁴

Step 3

Delete wounded arcs, producing M^* .

In the second baseball case the arc *Suzy Catches* \rightarrow *Window Shatters* starts out wounded: it's an arc that should never have been added and which no causal discovery algorithm would add. But many cases of wounding arise only when a specific context is instantiated. For example, in bottle smashing (Figure 4.12), if Suzy doesn't throw, there's nothing wrong with the causal process carrying influence from Billy's throw to the bottle. If we ask about Billy's throw specifically in the context of Suzy having thrown first, however, then there is no connecting process. The arc *Billy Throws* \rightarrow *Bottle Shatters* is **vulnerable** to that specific context, and, given the context, is wounded.

Until now, in Bayesian network modeling two kinds of relationship between pairs of variables $\langle C, E \rangle$ have been acknowledged: those for which there is always a probabilistic dependency regardless of context set, when a direct arc

²² The causal question being relative to an observational context presupposes that it is also relative to the causal model, including parameters, in which the context is set.

²³ An alternative analysis of this would be to say that since the wall *is* infallible, it effectively has only one state, and so is not a variable at all. Whether we really should remove the variable depends upon whether or not we wish to take seriously the possibility of it altering state, even if only by an explicit intervention. Regardless of whether we deal with the wall in parameters or structure, there will remain no possible dependency between Suzy's catch and the window.

²⁴ For a more careful discussion of the metaphysics of process and wounding we refer the reader to [19].

must be added between them;²⁵ and those pairs which are screened off from each other by some context set (possibly empty). But vulnerable arcs are those which are sometimes needed, they connect pairs of the first type above, but also they are sometimes not needed; when they are wounded, the arcs can mediate no possible probabilistic dependency, since they cannot participate in any active path. We might say, the arcs flicker on and off, depending upon the actual context.

Strictly speaking, deleting wounded arcs is not necessary for the assessment of token causality, since the next step applies a dependency test which is already sensitive to wounding. Removing the wounded arc simply makes the independence graphic, as faithful models do.

Step 4

Determine whether intervening upon C can make a probabilistic difference to E in the given circumstances M^*/O .

Note that we make no attempt here to isolate any path connecting C and E (in contrast with our treatment of type causality above). In token causation our interest is in identifying whether C actually *does* make a difference within context O ; to answer this question we must allow that alternative strands of influence may nullify the affect. Thus, in questions of token causation we should allow for neutralizing alternatives. In neutral Hesslow, for example, we would not attribute token causality to the pill, whether or not thrombosis ensued — unless, of course, the woman's state of pregnancy were fixed as part of the context. Allowing neutralization for a type question is not an option, however, since the type question indicates a desire to know whether there is some possible extension to context which yields a difference-making intervention, which there is in neutral Hesslow. So, in this way, token causality differs from a straightforward particularization of type causality, by being bound to the specific context M^*/O .

Since we are not focused here on causal role, the probabilistic difference identified in Step 4 might well be to *reduce* the probability of E ; hence, rather than saying that C actually caused E , it might be better simply to say that C is actually causally relevant to E . As the context in which the token causal question gets raised, O , is enlarged, this criterion becomes more particular to the historical circumstances; as the context O shrinks, this criterion more closely resembles type causal relevance, with the exception noted above.

4.10 Conclusion

Bayesian network technology and its philosophy have reached an important juncture. The representational and inferential methods have proven themselves in

²⁵ This means, for every possible set of context variables O there is *some* instantiation of the context $O = o$ such that $P(E|C, O = o) \neq P(E|O = o)$. This is not to be confused with requiring that for all possible context sets, and *all possible instantiations thereof*, C and E are probabilistically dependent, which is a mistake Cartwright [4] makes.

academic practice and are beginning to be taken up widely in industry and scientific research. This, and the difficulty in building them by expert elicitation, has fueled considerable work in the automation of causal discovery, which in turn has prompted a reconsideration of the causal interpretation of the models discovered, by both supporters and skeptics. The friction between the two subcommunities has sparked a variety of ideas, both philosophical disputes and initiatives potentially valuable in application, such as the measures of causal power introduced above. This is a lively and productive time for research in causal modeling, both theoretical and applied.

Acknowledgements

We thank Erik Nyberg, Charles Twardy, Toby Handfield, Graham Oppy and Luke Hope for contributing to work and discussions that we have drawn upon here.

References

1. Bouissou, M., Thuy, N.: Decision making based on expert assessments: Use of belief networks to take into account uncertainty, bias, and weak signals. In: Decision-making aid and control of the risks: Lambda-Mu 13/ESREL 2002 Conference, Lyon, France (2002)
2. Bovens, L., Hartmann, S.: Bayesian networks and the problem of unreliable instruments. *Philosophy of Science* 69, 29–72 (2002)
3. Cain, J.: Planning improvements in natural resources management. Technical report, Centre for Ecology and Hydrology (2001)
4. Cartwright, N.: What is wrong with Bayes nets? *The Monist* 84, 242–264 (2001)
5. Cartwright, N.: *Nature's Capacities and their Measurement*. Clarendon Press, Oxford (1989)
6. Cheng, P.W.: From covariation to causation: A causal power theory. *Psychological Review* 104, 367–405 (1997)
7. Chickering, D.M.: A transformational characterization of equivalent Bayesian network structures. In: Besnard, P., Hanks, S. (eds.) *Proc. of the 11th Conf. on Uncertainty in AI*, San Francisco, pp. 87–98 (1995)
8. Max Chickering, D.: Optimal structure identification with greedy search. *Machine Learning Research* 3, 507–559 (2002)
9. Collins, J.: Preemptive prevention. *Journal of Philosophy* 97, 223–234 (2000)
10. Cooper, G.F.: The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence* 42, 393–405 (1990)
11. Cooper, G.F., Herskovits, E.: A Bayesian method for constructing Bayesian belief networks from databases. In: D'Ambrosio, S., Bonissone (eds.) *uai1991*, pp. 86–94 (1991)
12. Dowe, P.: *Physical Causation*. Cambridge University, New York (2000)
13. Fisher, R.A.: Letter. *British Medical Journal*, 297–298 (August 3, 1957)
14. Glymour, C.: *The Mind's Arrows: Bayes Nets and Graphical Causal Models in Psychology*. MIT, Cambridge (2001)

15. Glymour, C., Cheng, P.W.: Causal mechanism and probability: A normative approach. In: Oaksford, M., Chater, N. (eds.) *Rational models of cognition*, Oxford (1998)
16. Hall, N.: Two concepts of causation. In: Collins, J., Hall, N., Paul, L.A. (eds.) *Causation and Counterfactuals*, pp. 225–276. MIT Press, Cambridge (2004)
17. Halpern, J.Y., Pearl, J.: Causes and explanations, part I. *British Journal for the Philosophy of Science* 56, 843–887 (2005)
18. Halpern, J.Y., Pearl, J.: Causes and explanations, part II. *British Journal for the Philosophy of Science* 56, 889–911 (2005)
19. Handfield, T., Oppy, G., Twardy, C., Korb, K.B.: Probabilistic process causality (2005); Under Submission
20. Hausman, D.M.: Probabilistic causality and causal generalizations. In: Eells, E., Fetzer, J.H. (eds.) *The Place of Probability in Science*, Open Court (2005)
21. Heckerman, D., Geiger, D., Chickering, D.M.: Learning Bayesian networks: the combination of knowledge and statistical data. In: de Mantras, L., Poole, D. (eds.) *Proc. of the 10th Conf. on Uncertainty in AI*, San Francisco, pp. 293–301 (1994)
22. Hiddleston, E.: Causal powers. *British Journal for the Philosophy of Science* 56, 27–59 (2005)
23. Hitchcock, C.R.: The intransitivity of causation revealed in equations and graphs. *Journal of Philosophy* 158(6), 273–299 (2001)
24. Hitchcock, C.R.: Routes, processes and chance-lowering causes. In: Dowe, P., Noordhof (eds.) *Cause and Chance*, Routledge, pp. 138–151 (2004)
25. Hope, L.R., Korb, K.B.: An information-theoretic causal power theory. In: *Proc. of the 18th Australian Joint Conference on AI*, Sydney, NSW, pp. 805–811. Springer, Heidelberg (2005)
26. Kitcher, P.: Explanatory unification and the causal structure of the world. In: Kitcher, P., Salmon, W.C. (eds.) *Minnesota Studies in the Philosophy of Science*, Univ. of Minnesota, vol. XIII, pp. 410–505 (1989)
27. Korb, K.B., Hope, L.R., Nicholson, A.E., Axnick, K.: Varieties of causal intervention. In: *Pacific Rim International Conference on AI*, pp. 322–231 (2004)
28. Korb, K.B., Nicholson, A.E.: *Bayesian Artificial Intelligence*. CRC/Chapman and Hall, Boca Raton, Fl (2004)
29. Korb, K.B., Nyberg, E.: The power of intervention. *Minds and Machines* 16, 289–302 (2006)
30. Korb, K.B., Twardy, C.R., Handfield, T., Oppy, G.: Causal reasoning with causal models. Technical Report 2005/183, School of Computer Science and Software Engineering, Monash University (2005)
31. Lewis, D.: Causation. *Journal of Philosophy* 70, 556–567 (1973)
32. Lewis, D.: *Philosophical Papers*, vol. II. Oxford Univ., Oxford (1986)
33. Lewis, D.: Void and object. In: Collins, J., Hall, N., Paul, L.A. (eds.) *Causation and Counterfactuals*, pp. 277–290. MIT Press, Cambridge (2004)
34. Marcot, B.G., Holthausen, R.S., Raphael, M.G., Rowland, M.M., Wisdom, M.J.: Using Bayesian belief networks to evaluate fish and wildlife population viability under land management alternatives from an environmental impact statement. *Forest Ecology and Management* 153, 29–42 (2001)
35. Menzies, P.: Difference making in context. In: Collins, J., Hall, N., Paul, L. (eds.) *Counterfactuals and Causation*, pp. 139–180. MIT Press, Cambridge (2004)
36. Neapolitan, R.E.: *Learning Bayesian Networks*. Prentice Hall, Englewood Cliffs (2003)
37. Novick, L.R., Cheng, P.W.: Assessing interactive causal influence. *Psychological Review* 111, 455–485 (2004)

38. Nyberg, E.P., Korb, K.B.: Informative interventions. Technical Report 2006/204, School of Information Technology, Monash University (2006)
39. Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann, San Mateo (1988)
40. Pearl, J.: Causality: Models, reasoning and inference, Cambridge (2000)
41. Popper, K.R.: The Logic of Scientific Discovery. Basic Books, New York (1959); Translation, with new appendices, of Logik der Forschung Vienna (1934)
42. Popper, K.R.: Objective Knowledge: An Evolutionary Approach. Oxford University, Oxford (1972)
43. Reichenbach, H.: The Direction of Time. Univ of California, Berkeley (1956)
44. Rosen, D.: In defense of a probabilistic theory of causality. *Philosophy of Science* 45, 604–613 (1978)
45. Salmon, W.C.: Scientific Explanation and the Causal Structure of the World, Princeton (1984)
46. Salmon, W.: Probabilistic causality. *Pacific Philosophical Quarterly*, 50–74 (1980)
47. Spirtes, P., Glymour, C., Scheines, R.: Causation, Prediction and Search. Lecture Notes in Statistics, vol. 81. Springer, Heidelberg (1993)
48. Spirtes, P., Glymour, C., Scheines, R.: Causation, Prediction and Search, 2nd edn. MIT Press, Cambridge (2000)
49. Steel, D.: Homogeneity, selection and the faithfulness condition. *Minds and Machines* 16, 303–317 (2006)
50. Suppes, P.: A Probabilistic Theory of Causality, North Holland, Amsterdam (1970)
51. Twardy, C.R., Nicholson, A.E., Korb, K.B.: Knowledge engineering cardiovascular Bayesian networks from the literature. Technical Report 2005/170, Clayton School of IT, Monash University (2005)
52. Twardy, C., Korb, K.B.: A criterion of probabilistic causation. *Philosophy of Science* 71, 241–262 (2004)
53. Verma, T.S., Pearl, J.: Equivalence and synthesis of causal models. In: D'Ambrosio, S., Bonissone (eds.) *Proc. of the Sixth Conference on Uncertainty in AI*, pp. 255–268 (1991)
54. Wallace, C.S., Korb, K.B.: Learning linear causal models by MML sampling. In: Gammerman, A. (ed.) *Causal Models and Intelligent Data Management*. Springer, Heidelberg (1999)
55. Witten, I.H., Frank, E.: Data mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)
56. Woodward, J.: Making Things Happen. Oxford Univ., Oxford (2003)
57. Wright, S.: The method of path coefficients. *Annals of Mathematical Statistics* 5(3), 161–215 (1934)

An Introduction to Bayesian Networks and Their Contemporary Applications

Daryle Niedermayer, I.S.P.*

College of the North Atlantic-Qatar, Doha, Qatar

Abstract. Bayesian Networks are an important area of research and application within the domain of Artificial Intelligence. This paper explores the nature and implications for Bayesian Networks beginning with an overview and comparison of inferential statistics with Bayes' Theorem. The nature, relevance and applicability of Bayesian Network theory for issues of advanced computability form the core of the current discussion. A number of current applications using Bayesian networks are examined. The paper concludes with a brief discussion of the appropriateness and limitations of Bayesian Networks for human-computer interaction and automated learning.

5.1 Introduction

Inferential statistics is a branch of statistics that attempts to make valid predictions based on only a sample of all possible observations. For example, imagine a bag of 10,000 marbles. Some are black and some white, but the exact proportion of these colours is unknown. It is unnecessary to count all the marbles in order to make some statement about this proportion. A randomly acquired sample of 1,000 marbles may be sufficient to make an inference about the proportion of black and white marbles in the entire population. If 40% of our sample is white, then we may be able to infer that about 40% of the population is also white.

To the layperson, this process seems rather straight forward. In fact, it might seem that there is no need to even acquire a sample of 1,000 marbles. A sample of 100 or even 10 marbles might do.

This assumption is not necessarily correct. As the sample size becomes smaller, the potential for error grows. For this reason, inferential statistics has developed numerous techniques for stating the level of confidence that can be placed on these inferences.

If we took ten samples of 100 marbles each, we might find the following results:

* This paper is revised from an earlier work dated December 1, 1998. ©1998, 2008 by Daryle Niedermayer. All Rights Reserved.

Table 5.1. Relative proportions of 10 samples from a population of 10,000

Sample Number (n)	Number of White Marbles (X)	Number of Black Marbles
1	40	60
2	35	65
3	47	53
4	50	50
5	31	69
6	25	75
7	36	64
8	20	80
9	45	55
10	55	45

We are then in a position to calculate the mean and standard deviation of these samples. Our sample mean of white marbles from our sample is denoted by \bar{X} to distinguish it from the actual proportion of white and black marbles in the bag which we refer to as μ . The standard deviation denoted by the symbol sigma (σ) is found using equations 1 and 2:

$$\sigma = \sqrt{\frac{\sum x^2}{n-1}} \tag{eq. 5.1}$$

where x^2 is the sum of the squares and n is the number of samples taken, so that the equation is expanded to:

$$\sigma = \sqrt{\frac{(X_1 - \bar{X})^2 + (X_2 - \bar{X})^2 \dots (X_n - \bar{X})^2}{n-1}} \tag{eq. 5.2}$$

The value of \bar{X} derived from our 10 samples is 38.4. We might be tempted to say that about 40% of the marbles are white, but how sure are we that our samples are representative of the actual population of black and white marbles in the bag, μ ? Determining this level of confidence requires some analysis concerning the variability in our samples.

Using equation 2 above, we determine that $\sigma=11.15$. We must then determine the ‘‘Sample Error of the Mean’’ (denoted by ζ):

$$\zeta_{\bar{X}} = \frac{\sigma}{\sqrt{n}} \tag{eq. 5.3}$$

In our example, $\zeta_{\bar{X}}$ is 3.53.

The confidence we can put on our hypothesis that 40% of the marbles are white is then found using a standard statistical test called a ‘‘z-test’’:

$$z = \frac{\bar{X} - \mu}{\zeta_{\bar{X}}} \quad (\text{eq. 5.4})$$

Using our hypothesized value of 40% for μ , and entering our values for \bar{X} and $\zeta_{\bar{X}}$, we obtain a z-value of 0.4535. Looking this value up in a z-test table or using a z-test computer algorithm, we would find that 32% of the area of the normal curve falls below this “z” value.ⁱ

Based on the inference that the more we sample our bag of marbles, the more we will find the number of white marbles in our samples (\bar{X}) clustering around the actual number of white marbles (μ), the z-test helps us determine how confident we can be in our hypothesis.

Figure 5.1a) illustrates this confidence. If μ actually is 40%, then only 32% of any samples we draw (represented by the shaded area) should have less than 38.4% white marbles.

However, we are assuming that our sample mean is less than our actual mean, μ . This isn't necessarily the case. Because the z-curve is symmetrical, we can see that there is also a 32% chance that a sample will have more than 41.6% white balls if μ is 40%. This possibility is shown in Figure 5.1b). For this reason, we say that our sample distribution is two-sided or “two-tailed”.

In summary, our series of ten samples can only give us a 36% confidence level that the actual percentage of white marbles in the bag is $40\% \pm 1.6\%$. Clearly, the confidence we can place in our conclusion is not as good as it was on first glance. This lack of confidence is due to the high variability among the samples. If we took more samples or larger samples, our confidence in our conclusion might increase.

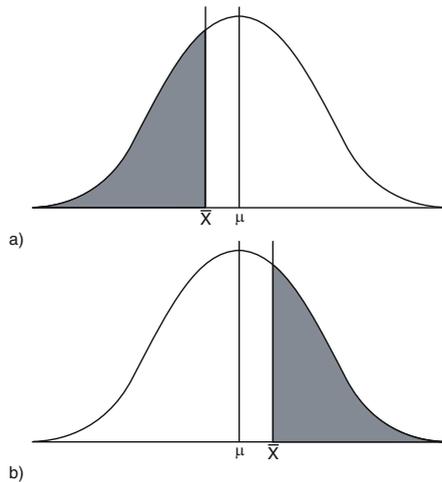


Fig. 5.1. Areas of the distribution curve used in z-tests

ⁱ An example of such a table can be obtained at: http://instruct.uwo.ca/geog/500/z_test.htm (accessed November 2007).

5.2 An Introduction to Bayesian Inference

Classical inferential models do not permit the introduction of prior knowledge into the calculations. For the rigours of the scientific method, this is an appropriate response to prevent the introduction of extraneous data that might skew the experimental results. However, there are times when the use of prior knowledge would be a useful contribution to the evaluation process.

Assume a situation where an investor is considering purchasing some sort of exclusive franchise for a given geographic territory. Her business plan suggests that she must achieve 25% market saturation for the enterprise to be profitable. Using some of her investment funds, she hires a polling company to conduct a randomized survey. The results conclude that from a random sample of 20 consumers, 25% of the population would indeed be prepared to purchase her services. Is this sufficient evidence to proceed with the investment?

If this is all the investor has to go on, she could find herself on the break-even point or could just as easily turn a loss instead of a profit. On the basis of this data alone, she may not have enough confidence to proceed with her plan.

Fortunately, the franchising company has a wealth of experience in exploiting new markets. Their results show that in 20% of cases, new franchises only achieve 25% market saturation, while in 40% of cases, new franchises achieve 30% market saturation. The entire table of their findings appears below:

Table 5.2. Percent of New Franchises achieving a given Market Saturation

Market Saturation (Proportion) (p)	Percent of Franchises (Relative Frequency)
0.10	0.05
0.15	0.05
0.20	0.20
0.25	0.20
0.30	0.40
0.35	0.10
	Total = 1.00

Our investor's question is simple: "What is the probability that my population will achieve a market saturation of greater than 25% given the poll conducted and the results found in other places?" In effect, she needs to determine the probability that her population will one of the 70% of cases where market saturation is greater than or equal to 25%. She now has the information she needs to make a Bayesian inference of her situation.

5.2.1 Bayes Theorem

Bayes' Theorem, developed by the Rev. Thomas Bayes, an 18th century mathematician and theologian, was first published in 1763¹. Mathematically it is expressed as:

$$P(H | E, c) = \frac{P(H | c) \times P(E | H, c)}{P(E | c)} \quad (\text{eq. 5.5})$$

Essentially, we can update our belief in hypothesis H given the additional evidence E and the background context c . The left-hand term, $P(H|E,c)$ is known as the “*posterior probability*”, or the probability of H after considering the effect of E on c . The term $P(H|c)$ is called the “*prior probability of H given c alone.*” The term $P(E|H,c)$ is called the “*likelihood*” and gives the probability of the evidence assuming the hypothesis H and the background information c is true. Finally, the last term $P(E|c)$ is independent of H and can be regarded as a normalizing or scaling factor.

In the case of our hypothetical investor, the franchiser already knows that 50% of prior franchisees are profitable, so $P(H|c)$ is already known:

$$P(H | E, c) = \frac{0.50 \times P(E | H, c)}{P(E | c)} \quad (\text{eq. 5.6})$$

It is important to note that all of these probabilities are conditional. They specify the degree of belief in some proposition or propositions based on the assumption that some other propositions are true. As such, the theory has no meaning without prior resolution of the probability of these antecedent propositions.

5.2.2 Bayes Theorem Applied

Let us return the example of the investor. From theory of binomial distributions, if the probability of some event occurring on any one trial is p , then the probability of x such events occurring out of n trials is expressed as:

$$P(x) = \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x} \quad (\text{eq. 5.7})^2$$

For example, the likelihood that 5 out of 20 people will support her enterprise should her location actually fall into the category where 20% of franchises actually achieve 25% saturation is:

$$P(x = 5 | p_{.20}) = \frac{20!}{5!(20-5)!} (0.25)^5 (0.75)^{15} = 0.20233 \quad (\text{eq. 5.8})$$

The likelihood of the other situations can also be determined:

The sum of all the Joint Probabilities provides the scaling factor found in the denominator of Bayes Theorem and is ultimately related to the size of the sample. Had the sample been greater than 20, the relative weighting between prior knowledge and current evidence would be weighted more heavily in favour of the latter. The Posterior Probability column of Table 3 shows the results of the Bayesian theorem for this case. By adding up the relative posterior probabilities for market shares $\geq 25\%$ and those $< 25\%$, our investor will see that there is a 75% probability that her franchise will make money—definitely a more attractive situation on which to base an investment decision.

Table 5.3. Likelihood of An Investor Finding herself in each situation given $x=5$ and $n=20$

Event (Market Saturation) (p_i)	Prior Probability $P_0(p_i)$	Likelihood of Situation $P(x=5 p_i)$	Joint Probability of Situation $P(x=5 p_i)P_0(p_i)$	Posterior Probability ⁱⁱ
0.10	0.05	0.03192	0.001596	0.00959
0.15	0.05	0.10285	0.005142	0.00309
0.20	0.20	0.17456	0.034912	0.20983
0.25	0.20	0.20233	0.040466	0.24321
0.30	0.40	0.17886	0.071544	0.43000
0.35	0.10	0.12720	0.012720	0.07645
	1.00	0.81772	0.166381	0.99997

5.3 Bayesian Networks

5.3.1 Introduction

The concept of conditional probability is a useful one. There are countless real world examples where the probability of one event is conditional on the probability of a previous one. While the sum and product rules of probability theory can anticipate this factor of conditionality, in many cases such calculations are “NP-hard” meaning that they are computationally difficult and cannot be determined in polynomial time. The prospect of managing a scenario with 5 discrete random variables ($2^5-1=31$ discrete parameters) might be manageable. An expert system for monitoring patients with 37 variables resulting in a joint distribution of over 2^{37} parameters would not be manageable.³

5.3.2 Definition

Consider a domain U of n variables, x_1, \dots, x_n . Each variable may be discrete having a finite or countable number of states, or continuous. Given a subset X of variables x_i where $x_i \in U$, if one can observe the state of every variable in X , then this observation is called an instance of X and is denoted as:

$$X = p(x_i | x_1, \dots, x_{i-1}, \xi) = p(x_i | \Pi_i, \xi) \vec{k}_X \text{ for the observations } x_i = k_i, x_i \in X \tag{eq. 9}$$

The “joint space” of U is the set of all instances of U . $p(X = \vec{k}_X | Y = \vec{k}_Y, \xi)$ denotes the “generalized probability density” that $X = p(x_i | x_1, \dots, x_{i-1}, \xi) = p(x_i | \Pi_i, \xi) \vec{k}_X$ given $Y = \vec{k}_Y$ for a person with

ⁱⁱ The Posterior Probability can be expressed as:

$$P_1(p_i) = P(p_i | x = 5) = \frac{P(x = 5 | p_i)P_0(p_i)}{P(x = 5)}$$

current state information ξ , $p(X|Y, \xi)$ then denotes the “*Generalized Probability Density Function*” (gpdf) for X , given all possible observations of Y . The joint gpdf over U is the gpdf for U .

A Bayesian network for domain U represents a joint gpdf over U . This representation consists of a set of local conditional gpdfs combined with a set of conditional independence assertions that allow the construction of a global gpdf from the local gpdfs. As shown previously, the chain rule of probability can be used to ascertain these values:

$$p(x_1, \dots, x_n | \xi) = \prod_{i=1}^n p(x_i | x_1, \dots, x_{i-1}, \xi) \quad (\text{eq. 5.10})$$

One assumption imposed by Bayesian Network theory (and indirectly by the Product Rule of probability theory) is that each variable x_i , $\Pi_i \subseteq \{x_1, \dots, x_{i-1}\}$ must be a set of variables that renders x_i and $\{x_1, \dots, x_{i-1}\}$ conditionally independent. In this way:

$$p(x_i | x_1, \dots, x_{i-1}, \xi) = p(x_i | \Pi_i, \xi) \quad (\text{eq. 5.11})^4$$

A Bayesian Network Structure then encodes the assertions of conditional independence in equation 10 above. Essentially then, a Bayesian Network Structure B_s “is a directed acyclic graph such that (1) each variable in U corresponds to a node in B_s , and (2) the parents of the node corresponding to x_i are the nodes corresponding to the variables in Π_i .”⁵

A Bayesian-network gpdf set B_p is the collection of local gpdfs: $p(x_i | \Pi_i, \xi)$ for each node in the domain.

5.3.3 Bayesian Networks Illustrated

Given a situation where it might rain today, and might rain tomorrow, what is the probability that it will rain on both days? Rain on two consecutive days are not independent events with isolated probabilities. If it rains on one day, it is more likely to rain the next. Solving such a problem involves determining the chances that it will rain today, and then determining the chance that it will rain tomorrow conditional on the probability that it will rain today. These are known as “joint probabilities.” Suppose that $P(\text{rain today}) = 0.20$ and $P(\text{rain tomorrow given that it rains today}) = 0.70$. The probability of such joint events is determined by:

$$P(E_1, E_2) = P(E_1)P(E_2 | E_1) \quad (\text{eq. 5.12})$$

This can also be expressed as:

$$P(E_2 | E_1) = \frac{P(E_1, E_2)}{P(E_1)} \quad (\text{eq. 5.13})^6$$

Working out the joint probabilities for all eventualities, the results can be expressed in a table format.

From the table, it is evident that the joint probability of rain over both days is 0.14, but there is a great deal of other information that had to be brought into the calculations before such a determination was possible. With only two discrete, binary variables, four calculations were required.

Table 5.4. Marginal and Joint Probabilities for rain both today and tomorrow

	Rain Tomorrow	No Rain Tomorrow	Marginal Probability of Rain Today
Rain Today	0.14	0.06	0.20
No Rain Today	0.16	0.64	0.80
Marginal Probability of Rain Tomorrow	0.30	0.70	

This same scenario can be expressed using a Bayesian Network Diagram as shown in Figure 5.2.

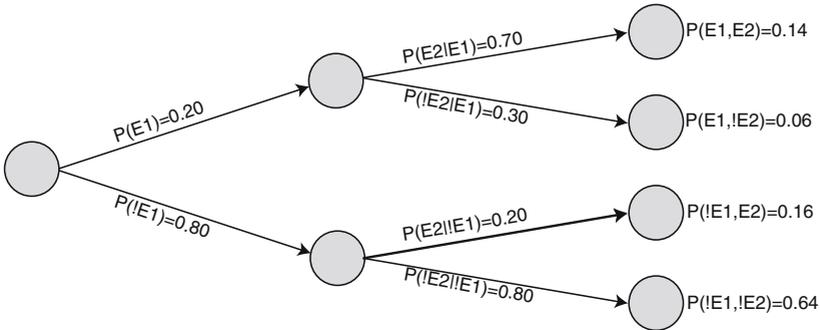


Fig. 5.2. A Bayesian Network showing the probability of rain

One attraction of Bayesian Networks are their efficiency owing to the fact that only one branch of the tree needs to be traversed. We are really only concerned with $P(E_1)$, $P(E_2|E_1)$ and $P(E_1, E_2)$.

We can also utilize the graph both visually and algorithmically to determine which parameters are independent of each other. Instead of calculating four joint probabilities, we can use the independence of the parameters to limit our calculations to two. It is self-evident that the probabilities of rain on the second day having rained on the first are completely autonomous from the probabilities of rain on the second day having not rained on the first.

At the same time as emphasizing parametric indifference, Bayesian Networks also provide a parsimonious representation of conditionality among parametric relationships. While the probability of rain today and the probability of rain tomorrow are two discrete events (it cannot rain both today and tomorrow at the same time), there is a conditional relationship between them (if it rains today, the lingering weather systems and residual moisture are more likely to result in rain tomorrow). For this reason, the directed edges of the graph are connected to show this dependency.

Friedman and Goldszmidt suggest looking at Bayesian Networks as a story. They offer the example of a story containing five random variables: “Burglary”, “Earthquake”, “Alarm”, “Neighbour Call”, and “Radio Announcement”.⁷ In such a story, “Burglary” and “Earthquake” are independent, and “Burglary” and “Radio Announcement” are independent given “Earthquake.” This is to say that there is no event

which effects both burglaries and earthquakes. As well, “Burglary” and “Radio Announcements” are independent given “Earthquake”—meaning that while a radio announcement might result from an earthquake, it will not result as a repercussion from a burglary.

Because of the independence among these variables, the probability of $P(A,R,E,B)$ (The joint probability of an alarm, radio announcement, earthquake and burglary) can be reduced from:

$$P(A,R,E,B) = P(A|R,E,B) \times P(R|E,B) \times P(E|B) \times P(B)$$

involving 15 parameters to 8:

$$P(A,R,E,B) = P(A|E,B) \times P(R|E) \times P(E) \times P(B)$$

This significantly reduced the number of joint probabilities involved. This can be represented as a Bayesian Network:

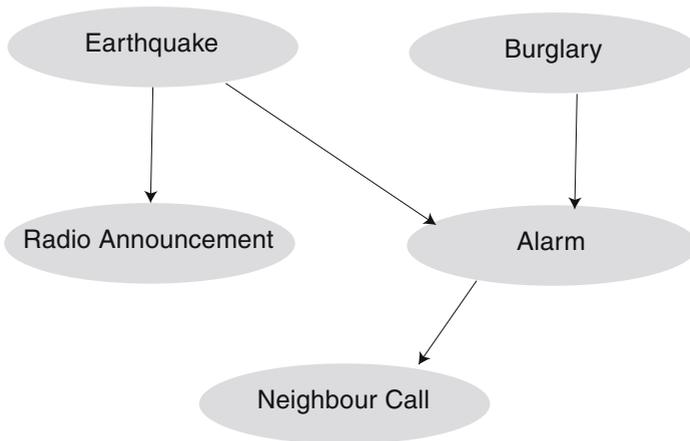


Fig. 5.3. The conditional probabilities of an alarm given the independent events of a burglary and earthquake

Using a Bayesian Network offers many advantages over traditional methods of determining causal relationships. Independence among variables is easy to recognize and isolate while conditional relationships are clearly delimited by a directed graph edge: two variables are independent if all the paths between them are blocked (given that edges are directional). Not all the joint probabilities need to be calculated to make a decision; extraneous branches and relationships can be ignored (One can make a prediction of a radio announcement regardless of whether an alarm sounds). By optimizing the graph, every node can be shown to have at most k parents. The algorithmic routines required can then be run in $O(2^k n)$ instead of $O(2^n)$ time. In essence, the algorithm can run in linear time (based on the number of edges) instead of exponential time (based on the number of parameters).

Associated with each node is a set of conditional probability distributions. For example, the “Alarm” node might have the following probability distribution:

Table 5.5. Probability Distribution for the Alarm Node given the events of “Earthquakes” (E) and “Burglaries” (B)

Earthquakes	Burglaries	$P(A E,B)$	$P(!A E,B)$
Earthquake	Burglary	0.90	0.10
Earthquake	No Burglary	0.20	0.80
No Earthquake	Burglary	0.90	0.10
No Earthquake	No Burglary	0.01	0.99

For example, should there be both an earthquake and a burglary, the alarm has a 90% chance of sounding. With only an earthquake and no burglary, it would only sound in 20% of the cases. A burglary unaccompanied by an earthquake would set off the alarm 90% of the time, and the chance of a false alarm given no antecedent event should only have a probability of 0.1% of the time. Obviously, these values would have to be determined through observation.

5.4 Algorithmic Implications of Bayesian Networks

Bayesian networks are useful for both inferential exploration of previously undetermined relationships among variables as well as descriptions of these relationships upon discovery. In the former case, raw computational power can be brought to bear upon a problem. In the case of determining the likelihood of rain the next day following a rainy day, raw meteorological data can be input into the computer and the computer can determine the resultant probability network. This process of network discovery is discussed by Friedman & Goldszmidt. Such a process is computationally intensive and NP-hard in its algorithmic implications. The benefit of such a process is evident in the ability to describe the discovered network in the future. The calculation of any probability branch of the network can then be computed in linear time.

5.5 Practical Uses for Bayesian Networks

5.5.1 AutoClass

The National Aeronautic and Space Administration has a large investment in Bayesian research. NASA's Ames Research Center is interested in deep-space exploration and knowledge acquisition. In gathering data from deep-space observatories and planetary probes, an *a priori* imposition of structure or pattern expectations is inappropriate. Researchers do not always know what to expect or even have hypotheses to test when gathering such data. Bayesian inference is useful because it allows the inference system to construct its own potential systems of meaning upon the data. Once any implicit network is discovered within the data, the juxtaposition of this network against other data sets allows for quick and efficient testing of new theories and hypotheses.

The AutoClass project is an attempt to create Bayesian applications that can automatically interpolate raw data from interplanetary probes, and deep space explorations.⁸ A graphical example of AutoClass's capabilities is displayed in Figure 5.4.

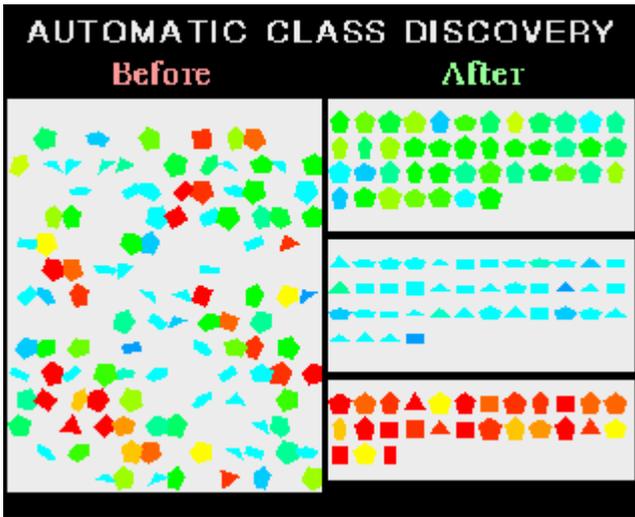


Fig. 5.4. An AutoClass interpolation of raw data with no predefined categories. Sorted data is grouped by colour and shape. The top area is sorted into green-blue shapes, the middle into blues, and the bottom into red-orange-yellow shapes. (Image courtesy of NASA, used with permission).

Incidentally, the source code for AutoClass is available in both LISP and C on an Open Source basis.

An applied example of AutoClass's capabilities was the input of infrared spectra. Although no differences among these spectra were initially suspected, AutoClass successfully distinguished two subgroups of stars.⁹

5.5.2 Introduction of Search Heuristics

Searching for a solution to a problem is usually an NP-hard problem resulting in a combinatorial explosion of possible solutions to investigate. This problem is often ameliorated through the use of heuristics, or sub-routines to make “intelligent” choices along the decision tree. An appropriately defined heuristic can quicken the search by eliminating obviously unsuccessful paths from the search tree. However, an inappropriately defined heuristic might eliminate the successful solutions and result in no evident solution.

Bayesian networks can replace heuristic methods by introducing a method where the probabilities are updated continually during search.

One class of search algorithms called “*Stochastic searching*” utilizes what are known as “Monte-Carlo” procedures. These procedures are non-deterministic and do not guarantee a solution to a problem. As such they are very fast, and repeated use of these algorithms will add evidence that a solution does not exist even though they never prove that such a solution is non-existent.

Combining procedures with knowledge of properties of the distribution from which problem instances are drawn can help extend the utility of these algorithms by focusing in on areas of the search tree not previously studied.

5.5.3 Lumiere

Microsoft began work in 1993 on Lumiere, its project to create software that could automatically and intelligently interact with software users by anticipating the goals and needs of these users.

This research was in turn based on earlier research on pilot-aircraft interaction.¹⁰ The concern of this investigation was the ability of a system to supply a pilot with information congruent with the pilot's current focus of attention. Extraneous information or information not related to the pilot's current task list was suppressed.

"This ability to identify a pilot's focus of attention at any moment during a flight can provide an essential link to the provision of effective decision support. In particular, understanding the current goals of a pilot decision maker can be applied to select the presentation of alternative systems and displays."¹¹

The Lumiere project at Microsoft eventually resulted in the "Office Assistant" with the introduction of the Office 95 suite of desktop products.¹²

5.6 Limitations of Bayesian Networks

In spite of their remarkable power and potential to address inferential processes, there are some inherent limitations and liabilities to Bayesian networks.

In reviewing the Lumiere project, one potential problem that is seldom recognized is the remote possibility that a system user might wish to violate the distribution of probabilities upon which the system is built. While an automated help desk system that is unable to embrace unusual or unanticipated requests is merely frustrating, an automated navigation system that is unable to respond to some previously unforeseen event might put an aircraft and its occupants in mortal peril. While these systems can update their goals and objectives based on prior distributions of goals and objectives among sample groups, the possibility that a user will make a novel request for information in a previously unanticipated way must be accommodated.

Two other problems also warrant consideration. The first is the computational difficulty of exploring a previously unknown network. To calculate the probability of any branch of the network, all branches must be calculated. While the resulting ability to describe the network can be performed in linear time, this process of network discovery is an NP-hard task which might either be too costly to perform, or impossible given the number and combination of variables.

The second problem centers on the quality and extent of the prior beliefs used in Bayesian inference processing. A Bayesian network is only as useful as this prior knowledge is reliable. Either an excessively optimistic or pessimistic expectation of the quality of these prior beliefs will distort the entire network and invalidate the results. Related to this concern is the selection of the statistical distribution induced in modeling the data. Selecting the proper distribution model to describe the data has a notable effect on the quality of the resulting network.

5.7 Conclusion

These concerns aside, Bayesian networks have incredible power to offer assistance in a wide range of endeavours. They support the use of probabilistic inference to update and revise belief values. Bayesian networks readily permit qualitative inferences

without the computational inefficiencies of traditional joint probability determinations. In doing so, they support complex inference modeling including rational decision making systems, value of information and sensitivity analysis. As such, they are useful for causality analysis and through statistical induction they support a form of automated learning. This learning can involve parametric discovery, network discovery, and causal relationship discovery.

In this paper, we discussed the premises of Bayesian networks from Bayes' Theorem and how such Bayesian inference differs from classical treatments of statistical inference. The reasons, implications and emerging potential of Bayesian networks in the area of Artificial Intelligence were then explored with an applied focus profiling some current areas where Bayesian networks and models are being employed to address real-life problems. Finally, we examined some of the limitations of Bayesian networks.

At best, such a paper can only be a snapshot of the state of Bayesian research at a given time and place. The breadth and eclectic foci of the many individuals, groups and corporations researching this topic makes it one of the truly dynamic areas within the discipline of Artificial Intelligence.

References

1. Stutz, J., Cheeseman, P.: A Short Exposition on Bayesian Inference and Probability. National Aeronautic and Space Administration Ames Research Centre: Computational Sciences Division, Data Learning Group (June 1994)
2. Morgan, B.W.: An Introduction to Bayesian Statistical Decision Processes, p. 15. Prentice-Hall Inc., Englewood Cliffs (1968)
3. Friedman, N., Goldszmidt, M.: Learning Bayesian Networks from Data, <http://robotics.stanford.edu/~nir/tutorial/Tutorial.ps.gz> (accessed November 2007)
4. Heckerman, D., Geiger, D., Chickering, D.: Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning* 20, 197–243 (1995), http://research.microsoft.com/research/pubs/view.aspx?tr_id=81 (accessed November 2007). This online version will be used for citation references throughout this paper
5. Heckerman, D., Geiger, D., Chickering, D., p. 6
6. Winkler, R.L.: An Introduction to Bayesian Inference and Decision. Holt, Rinehart and Winston. Toronto (1972)
7. Friedman, N., Goldszmidt, M.
8. Stutz, J., Taylor, W., Cheeseman, P.: AutoClass C - General Information. NASA, Ames Research Center (1998), <http://ic-www.arc.nasa.gov/ic/projects/bayes-group/autoclass/autoclass-c-program.html#AutoClassC> (accessed November 2007)
9. <http://ic-www.arc.nasa.gov/ic/projects/bayes-group/autoclass/index.html> (accessed November 2007)
10. Cooper, G., Horvitz, E., Curry R.: Conceptual Design of Goal Understanding Systems: Investigation of Temporal Reasoning Under Uncertainty. Decision Theory & Adaptive Systems Group, Microsoft Research. Microsoft Corp. Redmond, WA (1998), <http://research.microsoft.com/research/dtg/horvitz/goal.htm> (accessed November 2007)

11. Horvitz, E.: Lumiere Project: Bayesian Reasoning for Automated Assistance. Decision Theory & Adaptive Systems Group, Microsoft Research. Microsoft Corp., Redmond, WA (1998), <http://research.microsoft.com/research/dtg/horvitz/lum.htm> (accessed November 2007)
12. Heckerman, D., Horvitz, E.: Inferring Informational Goals from Free-Text Queries: A Bayesian Approach. Decision Theory & Adaptive Systems Group, Microsoft Research. Microsoft Corp., Redmond, WA (1998), <http://research.microsoft.com/research/dtg/horvitz/aw.htm> (accessed November 2007)

Objective Bayesian Nets for Systems Modelling and Prognosis in Breast Cancer

Sylvia Nagl¹, Matt Williams², and Jon Williamson³

¹ Department of Oncology, University College London
s.nagl@medsch.ucl.ac.uk

² Advanced Computation Laboratory, Cancer Research UK, and Computer Science,
University College London
m.williams@cs.ucl.ac.uk

³ Department of Philosophy, University of Kent
j.williamson@kent.ac.uk

Summary. Cancer treatment decisions should be based on all available evidence. But this evidence is complex and varied: it includes not only the patient's symptoms and expert knowledge of the relevant causal processes, but also clinical databases relating to past patients, databases of observations made at the molecular level, and evidence encapsulated in scientific papers and medical informatics systems. Objective Bayesian nets offer a principled path to knowledge integration, and we show in this chapter how they can be applied to integrate various kinds of evidence in the cancer domain. This is important from the systems biology perspective, which needs to integrate data that concern different levels of analysis, and is also important from the point of view of medical informatics.

In this chapter we propose the use of objective Bayesian nets for knowledge integration in the context of cancer systems biology. In Part I we discuss this context in some detail. Part II introduces the machinery that is to be applied, objective Bayesian nets. Then a proof-of-principle application is presented in Part III. Finally, in Part IV, we discuss avenues for further research.

Part I: Cancer Systems Biology and Knowledge Integration

6.1 Cancer Systems Biology

Cancer systems biology seeks to elucidate complex cell and tumour behaviour through the integration of many different types of knowledge. Information is obtained from scientific and clinical measurements made across biological scale, ranging from molecular components to systems, and from the genome to the whole patient. Integration of this information into predictive computational models, and their use in research and clinical settings, is expected to improve prevention, diagnostic and prognostic prediction, and treatment of cancer.

Systems biology addresses the complexity of cancer by drawing on a conceptual framework based on the current understanding of complex adaptive systems.¹ Complex systems are composed of a huge number of components that can interact simultaneously in a sufficiently rich number of parallel ways so that the system shows spontaneous self-organisation and produces global, emergent structures and behaviours.² Self-organisation concerns the emergence of higher-level order from the local interactions of system components in the absence of external forces or a pre-programmed plan embedded in any individual component.³

The challenges posed by the complex-systems properties of cancer are several-fold and can be thought about in terms of a taxonomy of complexity put forward by Mitchell:⁴

- Structural complexity;
- Dynamic complexity—complexity in functional processes;
- Evolved complexity—complex systems can generate alternative evolutionary solutions to adaptive problems; these are historically contingent.

Decisions need to be made in the face of great uncertainty regarding all three aspects of the complexity that is exhibited by the cancer systems in which one seeks to intervene.⁵ This is true both for therapeutic decisions for individual patients and also for design strategies leading to new anti-cancer therapies. Although our ability to collect ever more detailed quantitative molecular data on cells and cell populations in tumours is growing exponentially, and clinical investigations are becoming more and more sophisticated, our understanding of system complexity advances more slowly for the following reasons.

It is very difficult to directly observe and measure dynamic processes in complex systems, and this is particularly challenging in biomedicine where human subjects are involved. Research relies on data generated from tissue samples by high throughput technologies mainly directed at the ‘omic’ levels of the *genome*, *transcriptome* (gene transcripts) and *proteome* (proteins).⁶ However, data sampling is highly uneven and incomplete, and the data themselves are noisy and often hard to replicate. The molecular data that are being gathered typically only illuminate ‘single-plane’ omic slices ‘dissected’ out of entire systems which are characterized by highly integrated multi-scale organization and non-linear behaviour (Fig. 6.1). Furthermore, due to technological and economic constraints, current techniques can only capture a few time points out of the continuous systems dynamics, and are not yet able to address the ‘complexity explosion’ of control at the proteomic level. This situation is likely to persist for some time to come.

¹ (Nagl, 2006.)

² (Holland, 1995; Depew and Weber, 1996.)

³ (Holland, 1995, 1998; Mitchell, 2003.)

⁴ (Mitchell, 2003, pp. 4–7.)

⁵ (Nagl, 2005).

⁶ (Abramovitz and Leyland-Jones, 2006).

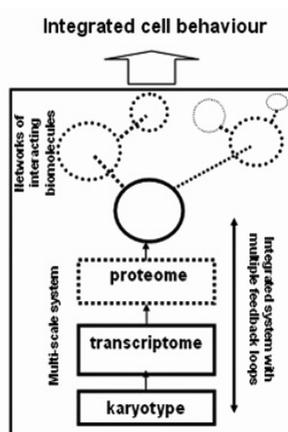


Fig. 6.1. Multi-scale integrated cell systems

A further challenge is posed by the need to relate molecular data to clinical measurements, notably those obtained in clinical trials, for identification of molecular parameters underlying physiological (dys)function. This integration task spans spatial and temporal scales of several orders of magnitude and complexity. Ultimately, cancer systems analysis needs to cut across all biological levels—genome, transcriptome, proteome, cell, and beyond to tissue, organ and patient (and the environment, but this is outside the present discussion). Toyoda and Wada (2004) have coined the term *omic space* and presented a hierarchical conceptual model linking different omic planes. They showed that this structuring of omic space helps to integrate biological findings and data comprehensively into hypotheses or models combining higher-order phenomena and lower-order mechanisms through a comprehensive ranking of correspondences among interactions in omic space. The key idea behind the concept of omic space may also serve as a general organising principle for *multi-scale systems*, and may be extended beyond cells to the tissue, organ and patient level. Below, we discuss how objective Bayesian nets can make significant contributions to the elucidation of multi-scale relationships in cancer systems (see §§6.4, 6.20).

6.2 Unstable Genomes and Complexity in Cancer

Tumours maintain their survival and proliferative potential against a wide range of anticancer therapies and immunological responses of the patient. Their robustness is seen as an emergent property arising through the interplay of genomic instability and selective pressure driven by host-tumour dynamics.⁷

⁷ (Kitano, 2004).

Progression from normal tissue to malignancy is associated with the evolution of neoplastic cell lineages with multiple genomic lesions (abnormal karyotypes).⁸ Most cancer cells do not have a significantly higher mutation rate at the nucleotide level compared to their normal counterparts, whereas extensive gross chromosomal changes are observed in liquid, and nearly all, solid tumours. The most common mutation class among the known cancer genes is chromosomal.

In cancer, dynamic large-scale changes of genome structure occur at dramatically increased frequency and tumour cell–microenvironment interactions drive selection of abnormal karyotypes. Copy-number changes, such as gene amplification and deletion, can affect several megabases of DNA and include many genes. These extensive changes in genome content can be advantageous to the cancer cell by simultaneous activation of oncogenes and elimination of tumour suppressors. Due to the magnitude of the observed genomic rearrangements, it is not always clear which gene, or set of genes, is the crucial target of the rearrangement on the basis of genetic evidence alone.

These changes encompass both directly cancer-causing and epiphenomenal changes (bystander mutations) which can nevertheless contribute significantly to the malignant phenotype and can modulate treatment resistance in a complex fashion. The ability of abnormal karyotypes to change autocatalytically in response to challenge, the masking of specific cancer-promoting constellations by collateral variation (any chromosomal combination that is specific for a selected function is also specific for many unselected functions), and the common phenomenon of several alternative cell pathways able to generate the same phenotype, limits the usefulness of context-independent predictions from karyotypic data. Interestingly, several researchers have put forward the theory that the control of phenotype is distributed to various extents among all the genetic components of a complex system.⁹

Mathematical modelling, adapted from Metabolic Control Analysis, suggests that it may in fact be the large *fraction* of the genome undergoing differential expression as a result of changes in gene dose (due to chromosomal rearrangements) that leads to highly non-linear changes in the physiology of cancer cells.

6.3 A Systems View of Cancer Genomes and Bayesian Networks

Genomes are dynamic molecular systems, and selection acts on cancer karyotypes as integrated wholes, not just on individual oncogenes or tumour suppressors. Given the irreversible nature of evolutionary processes, the randomness of mutations and rearrangements relative to those processes, and the modularity and redundancy of complex systems, there potentially exists a multitude of ways to ‘solve’ the problems of achieving a survival advantage in cancer cells.¹⁰ Since each patient’s cancer cells evolve through an independent set of genomic lesions and

⁸ (Nygren and Larsson, 2003; Vogelstein and Kinzler, 2004).

⁹ (Rasnick and Duesberg, 1999, and references therein).

¹⁰ (Mitchell, 2003, p. 7).

selective environments, the resulting heterogeneity of cell populations within the same tumour, and of tumours from different patients, is a fundamental reason for differences in survival and treatment response.

Since the discovery of oncogenes and tumour suppressors, a reductionist focus on single, or a small number of, mutations has resulted in cancer being conceptualized as a ‘genetic’ disease. More recently, cancer has been recast as a ‘genomic’ or ‘systems’ disease.¹¹ In the work presented in this chapter, we apply a systems framework to karyotype evolution and employ Bayesian networks to generate models of non-independent rearrangements at chromosomal locations from comparative genome hybridisation (CGH) data.¹² Furthermore, we present a method for integration of genomic Bayesian network models with nets learnt from clinical data. The method enables the construction of multi-scale nets from Bayesian nets learnt from independent datasets, with each of the nets representing the joint probability distributions of parameter values obtained from different levels of the biological hierarchy, i.e., the genomic and tumour level in the application presented here (together with treatment and outcome data). Bayesian network integration allows one to capture ‘more of the physiological system’ and to study dependency relationships across scales.¹³

Some of the questions one may address by application of our approach include

- utilising genomic (karyotype) data from patients:
 - Can we identify probabilistic dependency networks in large sample sets of karyotypes from individual tumours? If so, under which conditions may these be interpreted as causal networks?
 - Can we discover key features of the ‘evolutionary logic’ embodied in gene copy number changes of individual karyotypes?
 - Can we characterise the evolutionary ‘solution space’ explored by unstable cancer genomes? Is there a discernible dependence on cancer types?
- utilising omic and other molecular data together with clinical measurements:
 - Can we identify probabilistic dependency networks involving molecular and clinical levels?
 - How may such probabilistic dependencies aid diagnostic and prognostic prediction and design of personalised therapies?

6.4 Objective Bayesianism and Knowledge Integration

Bayesian networks are well suited to problems that require integration of data from various sources or data with different temporal or spatial resolutions. They can model complex non-linear relationships, and are also very robust to missing information. Bayesian network learning has already been successfully applied to data gathered at the transcriptomic and proteomic level for predictions regarding structure and function of gene regulatory, metabolic and signalling

¹¹ (Khalil and Hill, 2005; Lupski and Stankiewicz, 2006).

¹² (Reis-Filho et al., 2005).

¹³ (Nagl et al., 2006).

networks.¹⁴ Bulashevskaya and colleagues have applied Bayesian network analysis to allelotyping data in urothelial cancer.¹⁵ However, these studies also demonstrate persistent limits—only very partial answers have so far been obtained concerning the organization and dynamic function of whole biological systems which are by definition multi-scale and integrated by multiple feedback loops (see §6.1).

Employing objective Bayesianism as our methodology, we present a multi-scale approach to knowledge integration which utilises more fully the very considerable scope of available data. Our method enables integration of ‘omic’ data types and quantitative physiological and clinical measurements. These data combined offer rich, and as yet largely unexplored, opportunities for the discovery of probabilistic dependencies involving system features situated at multiple levels of biological organisation.

The technique supports progressive integration of Bayesian networks learnt from independently conducted studies and diverse data types, e.g., mRNA or proteomic expression, SNP, epigenetic, tissue microarray, and clinical data. New knowledge and new data types can be integrated as they become available over time. The application of our knowledge discovery method is envisaged to be valuable in the clinical trials arena which is undergoing far-reaching changes with steadily increasing incorporation of molecular profiling. It is our aim to assess the potential of our technique for integrating different types of clinical trial datasets (with and without molecular data). The methods described here are highly complementary to ongoing research initiatives, such as the Cancergrid project (www.cancergrid.org) and caBIG (cabig.nci.nih.gov) which are already addressing pressing informatics requirements that result from these changes in clinical study design.

6.5 Complementary Data Integration Initiatives

We are currently not in a position to make maximal use of existing data sets for Bayesian network analysis, since data have not yet been standardised in terms of experimental and clinical data capture (protocols, annotation, data reproducibility and quality), and computational data management (data formats, vocabularies, ontologies, metadata, exchange standards). Basic requirements are the generation of validated high-quality datasets and the existence of the various data sources in a form that is suitable for computational analysis and data integration. This has been well recognised, as is amply demonstrated by the aims and activities of a collaborative network of several large initiatives for data integration within the cancer domain which work towards shared aims in a coordinated fashion (the initiatives mentioned below are meant to serve as example projects and do not represent the sum total of these efforts on an international scale).

The National Cancer Institute Center for Bioinformatics (NCICB) in the United States has developed caCORE which provides an open-source suite of

¹⁴ (Xia et al., 2004).

¹⁵ (Bulashevskaya et al., 2004).

common resources for cancer vocabulary, metadata and data management needs (biological and clinical), and, from Version 3.0, achieves semantic interoperability across disparate biomedical information systems (for detailed information and access to the caCORE components, see ncicb.nci.nih.gov/NCICB/infrastructure/cacore_overview). caCORE plays an essential integrative role for the cancer Biomedical Informatics Grid (caBIG), a voluntary network connecting individuals and institutions to enable the sharing of data and tools, creating a ‘World Wide Web of cancer research’ whose goal is to speed up the delivery of innovative approaches for the prevention and treatment of cancer (cabig.nci.nih.gov/).

In the United Kingdom, the National Cancer Research Institute (NCRI) is developing the NCRI Strategic Framework for the Development of Cancer Research Informatics in the UK (www.cancerinformatics.org.uk). The ultimate aim is the creation of an internationally compatible informatics platform that would facilitate data access and analysis. CancerGRID develops open standards and information management systems (XML, ontologies and data objects, web services, GRID technology) for clinical cancer informatics, clinical trials, integration of molecular profiles with clinical data, and effective translation of clinical trials data to bioinformatics and genomics research (www.cancergrid.org).

Part II: Objective Bayesian Nets

6.6 Integrating Evidence Via Belief

In this information-rich age we are bombarded with evidence from a multiplicity of sources. This is evidence in a defeasible sense: items of evidence may not be true—indeed different items of evidence often contradict each other—but we take such evidence on trust until we learn that it is flawed or until something better comes along. In the case of breast cancer prognosis we have databases of molecular and clinical observations of varying reliability, current causal knowledge about the domain, knowledge encapsulated in medical informatics systems (e.g. argumentation systems, medical ontologies), and knowledge about the patient’s symptoms, treatment, and medical history. The key question is how we represent this eclectic body of evidence and render it coherent.

Knowledge impinges on belief, and one way in which we try to make sense of conflicting evidence is by finding a coherent set of beliefs that best fits this knowledge. We try to find beliefs that are consistent with undefeated items of evidence where we can, and where two items conflict we try to find some compromise beliefs. But this is vaguely put, and in this Part we shall describe a way of making this idea more precise.

Objective Bayesianism offers a formalism for determining the beliefs that best fit evidence; §6.7 offers a brief introduction to this theory. While this provides a useful theoretical framework, further machinery is required in order to find these beliefs and reason with them in practice—this is the machinery of objective Bayesian nets outlined in §6.8. In §6.9 we sketch a general procedure for constructing these nets, then in §6.10 we see how objective Bayesian nets can be used to

integrate qualitative evidence with quantitative evidence. Finally §6.11 discusses objective Bayesian nets in the context of the problem at hand, breast cancer.¹⁶

6.7 Objective Bayesianism

According to Bayesian theory, an agent's degrees of belief should behave like probabilities. Thus you should believe that a particular patient's cancer will recur to some degree representable by a real number x between 0 and 1 inclusive; you should believe that the patient's cancer will not recur to degree $1 - x$. Many Bayesians go further by adopting empirical constraints on degrees of belief. Arguably, for instance, degrees of belief should be calibrated with known frequencies: if you know just that 40% of similar patients have cancers that recur then you should believe that this patient's cancer will recur to degree 0.4. Objective Bayesians go further still, accepting not only empirical constraints on degrees of belief but also logical constraints: in the absence of empirical evidence concerning cancer recurrence you should equivocate on the question of this patient's cancer recurring—i.e. you should believe the cancer will recur to the same degree that you should believe it will not recur, 0.5.¹⁷

From a formal point of view the objective Bayesian position can be summed up as follows.¹⁸ Applying Bayesian theory, the agent's degrees of belief should be representable by a probability function p . Suppose that the agent has empirical evidence that takes the form of a set of quantitative constraints on p . Then she should adopt the probability function p , from all those that satisfy these constraints, that is maximally equivocal, i.e. that maximises entropy $H = -\sum_v p(v) \log p(v)$, where the sum is taken over all assignments $v = v_1 \cdots v_n$ to the variables V_1, \dots, V_n in the domain. This is known as the *maximum entropy principle*.¹⁹

Note that two items of empirical evidence may conflict—for example, the agent might be told that the frequency of recurrence is 0.4, but might also be told on another occasion that the frequency of recurrence is 0.3, with neither of the two reports known to be more reliable than the other and neither more pertinent to the patient in question. Arguably, the agent's degree of belief that the patient's cancer will recur will be constrained to lie within the closed interval $[0.3, 0.4]$. More generally, empirical constraints will constrain an agent's belief function to lie within a *closed convex* set of probability functions, and consequently there will be a unique function p that maximises entropy.²⁰ Thus the agent's rational belief function p is objectively determined by her evidence (hence the name *objective Bayesianism*).²¹

¹⁶ See Williamson (2002); Williamson (2005a, §5.5–5.8) and Williamson (2005b) for more detailed descriptions of the theory behind objective Bayesian nets.

¹⁷ (Russo and Williamson, 2007).

¹⁸ (Williamson, 2005a, Chapter 5).

¹⁹ (Jaynes, 1957).

²⁰ (Williamson, 2005a, §5.3).

²¹ See Williamson (2007b) for a general justification of objective Bayesianism, and Russo and Williamson (2007) for a justification within the cancer context.

We see then that objective Bayesianism provides a way of integrating evidence. The maximum entropy probability function p commits to the extent warranted by evidence: it satisfies constraints imposed by evidence but is non-committal where there is insufficient evidence. In that respect, objective Bayesian degrees of belief can be thought of as representative of evidence.

6.8 Obnets

Finding a maximum entropy probability function by searching for the parameters $p(v)$ that maximise the entropy equation is a computationally complex process and impractical for most real applications. This is because for a domain of n two-valued variables there are 2^n parameters $p(v)$ to calculate; as n increases the calculation gets out of hand. But more efficient methods are available. Bayesian nets, in particular, can be used to reduce the complexity of representing a probability function and drawing inferences from it.

A *Bayesian net* is a graphical representation of a probability function. The variables in the domain form the nodes of the graph. The graph also contains arrows between nodes, but must contain no cycles. Moreover, to each node is attached a probability table, containing the probability distribution of that variable conditional on its parents in the graph. As long as the *Markov condition* holds—i.e. each variable is probabilistically independent of its non-descendants in the graph conditional on its parents, written $V_i \perp\!\!\!\perp ND_i \mid Par_i$ —the net suffices to determine a probability function over the whole domain, via the identity

$$p(v) \stackrel{\text{df}}{=} p(v_1 \cdots v_n) = \prod_{i=1}^n p(v_i | par_i).$$

Thus the probability of an assignment to all the variables in the domain is the product of the probabilities of the variables conditional on their parents. These latter probabilities can be found in the probability tables. Depending on the sparsity of the graph, a Bayesian net can offer a much smaller representation of a probability function than that obtained by listing all the 2^n probability values $p(v)$. Furthermore, the Bayesian net can be used to efficiently draw inferences from the probability function and there is a wide variety of software available for handling these nets.²² The other chapters in this volume are testament to the importance of Bayesian nets for probabilistic reasoning.

An *objective Bayesian net*, or *obnet*, is a Bayesian net that represents objective Bayesian degrees of belief, i.e. that represents an agent's entropy-maximising probability function. Because the objective Bayesian belief function is determined in a special way (via the maximum entropy principle) there are special methods for constructing an objective Bayesian net, detailed in §6.9. These methods are more efficient to carry out than the more direct maximisation of the parameters $p(v)$ in the entropy equation.

²² (Neapolitan, 1990; Korb and Nicholson, 2003).

Given an objective Bayesian net, standard Bayesian net algorithms can be used to calculate probabilities, e.g. the probability of cancer recurrence given the characteristics of a particular patient. Thus an obnet can help with the task in hand, breast cancer prognosis. But an obnet can address other tasks too, for example the problem of knowledge discovery. An objective Bayesian net can suggest new relationships between variables: for instance if two variables are found to be strongly dependent in the obnet but there is no known connection between the variables that accounts for this dependence, then one might posit a causal connection to explain that link (§§6.16, 6.20). An obnet can also be used to determine new arguments to add to an argumentation framework: if one variable significantly raises the probability of another then the former is an argument for the latter (§6.18). Thus an obnet is a versatile beast that can assist with a range of tasks.

6.9 Constructing Obnets

One can build an objective Bayesian net by following a 3-step procedure. Given evidence, first determine conditional independencies that the entropy maximising probability function will satisfy. With this information about independencies one can then construct a directed acyclic graph for which the Markov condition holds. Finally, add the probability tables by finding the probability parameter $p(v_i | par_i)$ that maximise entropy.

The first step—finding the conditional independencies that p must satisfy—can be performed as follows. As before, we suppose that background knowledge imposes a set of quantitative constraints on p . Build an undirected *constraint graph* by taking variables as nodes and linking two variables if they occur together in some constraint. We can then read off probabilistic independencies from this graph: for sets of variables X, Y, Z , if Z separates X from Y in the constraint graph then X and Y will be probabilistically independent conditional on Z , $X \perp\!\!\!\perp Y \mid Z$, for the entropy maximising probability function p (Williamson, 2005a, Theorem 5.1).

The second step—determining the directed acyclic graph to go in the objective Bayesian net—is equally straightforward. One can transform the constraint graph into a directed acyclic graph G that satisfies the Markov Condition via the following algorithm:²³

- triangulate the constraint graph,
- re-order V according to maximum cardinality search,
- let D_1, \dots, D_l be the cliques of the triangulated constraint graph ordered according to highest labelled node,
- set $E_j = D_j \cap (\bigcup_{i=1}^{j-1} D_i)$ for $j = 1, \dots, l$,
- set $F_j = D_j \setminus E_j$ for $j = 1, \dots, l$,
- take variables in V as the nodes of G ,

²³ See Williamson (2005a, §5.7) for an explanation of the graph-theoretic terminology.

- add an arrow from each vertex in E_j to each vertex in F_j ($j = 1, \dots, l$),
- ensure that there is an arrow between each pair of vertices in D_j ($j = 1, \dots, l$).

The final step—determining the probability tables to go in the objective Bayesian net—requires some number crunching. One needs to find the parameters $p(v_i | par_i)$ that maximise the entropy equation, which can be written as $H = \sum_{i=1}^n H_i$ where $H_i = -\sum_{v_1 \dots v_n} \left(\prod_{V_j \in Anc_i} p(v_j | par_j) \right) \log p(v_i | par_i)$, (Anc_i being the set of ancestors of V_i in G). This optimisation task can be carried out in a number of ways. For instance, one can use numerical techniques or Lagrange multiplier methods to find the parameters.

This gives the general method for constructing an obnet. In §6.11 we shall tailor this method to our particular problem domain, that of breast cancer. But first we shall see how the method can be extended to handle qualitative evidence.

6.10 Qualitative Evidence

In the breast cancer domain, as elsewhere, evidence can take qualitative form. As well as quantitative evidence gleaned from clinical and molecular databases, there is qualitative causal knowledge and also qualitative evidence gleaned from medical ontologies and argumentation systems. In order to apply the maximum entropy principle in this type of domain, qualitative evidence must first be converted into a set of quantitative constraints on degrees of belief. Here we shall describe how this is possible.

Consider the following qualitative relationships: A is a cause of B ; A is a sub-type of B ; A is an argument in favour of B . These causal, ontological and evidential relations are all examples of what might be called *influence relations*. Intuitively A influences B if bringing about A brings about B but bringing about B does not bring about A . More precisely, a relation is an *influence relation* if it satisfies the following property: learning of the existence of new variables that are not influences of the other variables should not change degrees of belief concerning those other variables.²⁴

Qualitative knowledge of influence relationships can be converted into quantitative constraints on degrees of belief as follows. Suppose $V \supseteq U$ is a set of variables containing variables in U together with other variables that are known not to be influences of variables in U . As long as any other knowledge concerning variables in $V \setminus U$ does not itself warrant a change in degrees of belief on U , then $p_{\beta|U}^V = p_{\beta_U}^U$, i.e., one's belief function on the whole domain V formed on the basis of all one's background knowledge β , when restricted to U , should match the belief function one would have adopted on domain U given just the part β_U of one's knowledge involving U . These equality constraints can be used to constrain degrees of belief so that the maximum entropy principle can be applied. The equality constraints can also be fed into the procedure for constructing objective Bayesian nets: build the constraint graph as before from the

²⁴ (Williamson, 2005a, §11.4).

non-qualitative constraints; transform that graph into a directed acyclic graph as before; but take the qualitative constraints, converted to quantitative equality constraints, into account when determining the probability tables of the obnet (Williamson, 2005a, Theorem 5.6).

6.11 Obnets and Cancer

In the context of our project we have several sources of general (i.e., not patient-specific) evidence: databases of clinical data; databases of molecular data; a medical ontology; arguments from an argumentation framework; evidence of causal relationships from experts and also from published clinical trials. The study discussed in Part III focusses on databases of clinical and molecular data, but in this section we shall show how all these varied evidence sources might be integrated.

These sources impose a variety of constraints on a rational belief function p . Let C be the set of variables measured in a database of clinical data. Then $p|_C = \text{freq}_C$, the rational probability function when restricted to the variables in the clinical dataset should match the frequency distribution induced by that dataset. Similarly, if M is the set of variables measured in a molecular dataset, then $p|_M = \text{freq}_M$. A medical ontology determines influence relationships amongst variables. For example, knowledge that assignment a is a type (or sub-classification) of assignment b imposes the constraint $p(b|a) = 1$, as well as the influence constraint (§6.10) $p_{|A}^{\{A,B\}} = p^{\{A\}}$. An argumentation framework also determines influence relationships. An argument from a to b indicates that a and b are probabilistically dependent. This yields the constraint $p(b|a) \geq p(b) + \tau$ where τ is some threshold (which measures the minimum strength of arguments within the argumentation framework), as well as the influence constraint $p_{|A}^{\{A,B\}} = p^{\{A\}}$. Finally, causal evidence yields influence constraints of the form $p_{|A}^{\{A,B\}} = p^{\{A\}}$, and, if gleaned from a clinical trial, quantitative constraints of the form $p(b|a) \geq p(b) + \tau$.

In order to construct an objective Bayesian net from these sources, we can follow the three-step procedure outlined in §6.9.

The first task is to construct an undirected constraint graph. Following the recipe, we link all the variables in C (the clinical variables), link all the variables in M (the molecular / genomic variables), and link pairs of variables that are connected by an argument or by a clause in the ontology or by a causal relation. But we can reduce the complexity of the resulting obnet, which is roughly proportional to the density of the graph, still further as follows. One can use standard algorithms to induce a Bayesian net that represents the frequency distribution of the clinical database. Similarly, one can induce a Bayesian net from the molecular database. Then one can incorporate the independencies of these nets in the constraint graph, to render the constraint graph more sparse. This can be done as follows. Rather than linking every pair of variables in C with an edge in the constraint graph, include a link only if one is the parent of the other in frequency net, or if the two have a child in common in that net. Similarly for the variables in M . This yields a constraint graph with fewer edges, and thus a smaller obnet as a result.

The next two steps—converting the constraint graph into a directed acyclic graph, and adding the probability tables—can be carried out as detailed in §6.9.

Note that there is a particularly simple special case. If the evidence consists only of two databases which have just one variable in common, then one can construct the directed acyclic graph of the obnet thus: for each database learn a frequency net, ensuring that the variable in common is a root variable (i.e. has no parents); then just join the frequency nets at the root variable.

Having discussed the theoretical aspects of objective Bayesian nets, we now turn to a detailed description of the breast cancer application.

Part III: The Application

6.12 Obnets and Prediction in the Cancer Domain

We have applied objective Bayesian nets to the domain of breast cancer using three sources of data: one clinical, and two genomic, as well as a published study. The use of two genomic data sets was necessary as the more substantial genomic Bayesian net did not have a node in common with the clinical network, and so we used a smaller network to link the larger genomic network and the clinical one. We start by reviewing the data used (§§6.13, 6.14), and then in §6.15 describe how we constructed and merged the three separate networks. We then present some initial data on the performance of the network, and conclude the Part with a discussion of the uses of such networks in §6.16.

6.13 Breast Cancer

Breast Cancer is one of the commonest cancers in the Western World. It is the commonest non-skin cancer in women in the UK and US, and accounts for approximately a third of cancers in women, with lifetime rates of 1 in 10. Some 36000 cases are diagnosed each year in the UK, of whom about a third will die from the disease.²⁵ Consequently there has been a considerable amount of research focused on breast cancer, and death rates have fallen over the last 10 years.²⁶

The mainstay of treatment for breast cancer remains surgery and radiotherapy,²⁷ with hormonal and chemotherapeutic agents often used to treat presumed micro-metastatic disease. One of the advantages of surgery is that, as well as removing any local disease, a sample can also be taken of the axillary lymph nodes. These are a common site of metastatic spread for the cancer, and their removal not only removes any spread that may have occurred, but also allows analysis of the nodes to describe the degree of spread. The two main aims of treatment are to provide local control of, and to prevent premature death from, disease.

²⁵ (McPherson et al., 2000).

²⁶ (Quinn and Allen, 1995).

²⁷ (Richards et al., 1994).

Examination of the primary tumour and lymph nodes lets us define certain characteristics of the disease that make local recurrence and death more likely. These characteristics are primarily the grade of the tumour, (which represents the degree of abnormality displayed by the cells, scored 1-3), the size of the tumour (as its maximum diameter, in mm) and the number of involved nodes.²⁸ There are also newer tests for the presence or absence of certain proteins on the cell surface that may predict tumour behaviour or response to certain drugs.²⁹

The central aim of therapy planning is to match treatment with the risk of further disease. Thus those at high risk should be treated aggressively while those at low risk should be treated less aggressively. This allows more efficient use of resources, and restricts the (often considerable) side effects of intensive treatment to those patients who would benefit most.

Current Prognostic Techniques

These prognostic characteristics are currently modelled using statistical techniques to provide an estimate of the probability of survival and local recurrence. Two commonly used systems are the Nottingham Prognostic Index (NPI),³⁰ which uses data from large UK studies, and results derived from the American Surveillance, Epidemiology and End Results (SEER) database,³¹ which are used by systems such as Adjuvant Online.³² Both techniques rely on multivariate analyses of large volumes of data (based on over 3 million people for SEER) to calculate prognostic formulae.

These tools, and others like them, are effective at providing estimates of risk of death and local recurrence. However, they have two major weaknesses. Whilst effective, they lack explanatory power in a human-readable form. Therefore, extra knowledge that has not been captured by the statistical analysis (such as the presence and impact of other co-existing conditions) cannot be easily incorporated. Secondly, knowledge that post-dates the formation of the formulae (such as the discovery of Her-2neu, a cell-surface protein that is a marker for more aggressive disease) is very difficult to incorporate. Therefore, while they excel at providing an accurate assessment of population-based risk, they have weaknesses in the individualisation of that risk.

Humans are often poor at manipulating explicit probabilities;³³ however, clinicians have the ability to process additional knowledge that statistically-based systems often either ignore or treat on a perfunctory level. We would like to support clinical decision making by providing explicit probabilistic estimates of risk based on an integration of the variety of our knowledge sources.

²⁸ (Richards et al., 1994).

²⁹ (Veer et al., 2005; Cristofanilli et al., 2005).

³⁰ (Galea et al., 1992).

³¹ (Ries et al., 2004).

³² (Ravdin et al., 2001).

³³ (Kahneman and Tversky, 1973; Borak and Veilleux, 1982).

6.14 Our Knowledge Sources

Clinical Data

We used clinical data from a subset of the American Surveillance, Epidemiology and End Results (SEER) study. The total study is very large (over 3 million patients) and presents summary results on cancer diagnosis and survival in the USA between 1975 and 2003,³⁴ and subsets of the data are available for public use. We used a subset that initially consisted of 4878 individuals with breast cancer, which, once cases with incomplete data were removed, was reduced to 4731. The dataset consists of details of patient age (in 5 year bands, from 15–19 to 85+), the tumour size and histological grade, Oestrogen and Progesterone receptor status, the number of positive lymph nodes (if any), surgical type (mastectomy vs breast conserving), whether radiotherapy was given, the patients' survival from diagnosis (in months) and whether they had survived up until 5 years post-diagnosis. Patients in this subset were only followed up for 5 years, and so there is no data available on longer survival times.

Initial inspection of the clinical data was carried out using standard spreadsheet software (OpenOffice.org 2, 2005). Initial work concentrated on regrouping some of the data as follows. Oestrogen receptors are produced as part of the same intra-cellular pathway as Progesterone receptors, and as a result there is a very close correlation between ER & PR status. Since they are regarded as being one entity for most clinical purposes, we combined them into a single 'Hormone Receptor' variable. The Lymph Node status was converted from a number of positive lymph nodes (from 0–15) into a binary variable (True/ False), patient age was converted from 5 year age bands into 15–50, 50–70, 70–90, and Tumour size was converted from size in millimetres to sizes 0–20, 20–50, and 50–150 (these corresponding to clinical *T* Stages 1, 2, and 3+4). Patients with incomplete data (for example missing number of involved lymph nodes) were deleted from the dataset. A sample of the dataset is depicted in Table 6.1.

Table 6.1. A sample of the clinical dataset

Age	T Size	Grade	HR_status	Positive LN	Surgery	Radiotherapy	Survival	Status
70-74	22	2	1	1	1	1	37	1
45-49	8	1	1	0	2	1	41	1

The variables of the clinical database—i.e., the column headings of Table 6.1 are as follows:

Age: Age in years;

T Size: size of the primary tumour, in millimetres;

Grade: Histological grade of the tumour, from 1–3; 3 being most abnormal;

³⁴ (Ries et al., 2004).

HR_Status: Positive if the sample was positive for *either* Oestrogen or Progesterone receptors;

Positive LN: 1 if the patient had any lymph nodes involved by tumour, 0 otherwise;

Surgery: 1 if the patient had surgery for the tumour;

Radiotherapy: 1 if the patient received radiotherapy for their tumour;

Survival: Recorded survival in months;

Status: Status at final follow-up, 1 = alive, 0 = died.

Genomic Data

We used two karyotype datasets from the progenetix database (www.progenetix.de).³⁵ Progenetix contains discretised data on band-specific chromosomal rearrangements of cancer and leukemia cases (loss -1, gain +1, no change 0). It consists of a compilation of published data from comparative genome hybridisation (CGH), array CGH, and matrix CGH experiments, as well as some studies using metaphase analysis. Progenetix is, with 12320 CGH experiments, by far the largest public CGH database. We had available:

- (i) a breast cancer CGH dataset of 502 cases which lacked consistent clinical annotations, which we used to learn the genomic Bayesian net from band data only,
- (ii) a second CGH data set of 119 cases with clinical annotation, including lymph node status (an additional 12 individual cases with clinical annotation were set aside as a validation set), and
- (iii) a recent study, Fridlyand et al. (2006), which contains quantitative information concerning the probabilistic dependence between the variables HR_status and 22q12—this provided a further bridge between clinical and genomic variables.

From the total number of chromosomal bands in the human genome, we selected 28 bands in this proof-of-principle application. The chosen bands were hypothesised to be closely associated with tumour characteristics, progression and outcome (as represented by variables in the clinical net) based on genes with known function present on the bands. Genes were evaluated according to the biological processes they participate in, using their Gene Ontology annotations, e.g., cell cycle regulation, DNA damage repair and cancer-related signal pathways. An additional selection criterion was the presence of at least 3 relevant genes on the band.

The larger dataset consisted of 116 separate data fields. For reasons of space, 12 sample fields are reproduced in Table 6.2. The code of the form Np/qn indicates:

- N : which chromosome (1–22, X or Y);
- p/q : the short (p)/ long (q) arm of the chromosome;
- n : the band on the chromosome arm (0–40, depending on chromosome).

³⁵ (Baudis and Cleary, 2001).

Table 6.2. A sample of the larger of the genomic datasets

1p31	1p32	1p34	2q32	3q26	4q35	5q14	7p11	8q23	20p13	Xp11	Xq13
0	0	0	1	-1	0	0	1	0	0	0	-1
0	0	1	1	0	0	0	-1	-1	0	0	0

Table 6.3. A sample of the smaller of the genomic datasets

Lymph Nodes	1q22	1q25	1q32	1q42	7q36	8p21	8p23	8q13	8q21	8q24
0	1	1	1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0

The level of each band could either be unchanged (0) decreased (-1) or increased (1).

The smaller dataset was similar to the larger one, except for the fact that it included data on whether the patient's lymph nodes were also involved. 11 of the 26 data fields are reproduced in Table 6.3.

6.15 Constructing the Network

Our knowledge takes the form of a clinical database, two molecular databases, and information about the relationship between two variables derived from a research paper (§6.14). The clinical database determines a probability distribution $freq_c$ over the clinical variables and imposes the constraint $p_{|C} = freq_c$, i.e., the agent's probability function, when restricted to the clinical variables, should match the distribution determined by the clinical dataset. Similarly the first molecular database imposes the constraint $p_{|M} = freq_m$. The additional molecular dataset and the paper contain the observations that define the probability distribution $freq_s$ of three variables $S = \{ \text{HR_status, Positive LN, 22q12} \}$, the first two of which occur in the clinical dataset and the other of which occurs in the molecular dataset; it imposes the constraint $p_{|S} = freq_s$.³⁶

Given constraints of this form, an obnet on the variables in C , M and S can be constructed in the following way. First use standard methods, such as Hugin software, to learn a Bayesian net from the clinical dataset that represents $freq_c$, subject to the condition that the linking variables (positive LN, HR status) are root variables (Fig. 6.2). Similarly learn a Bayesian net from the larger genomic

³⁶ Fridlyand et al. (2006) report frequency data on gain and loss of 22q12 in breast cancer dependent on oestrogen hormone receptor status. Interestingly, loss of 22q12 is far more frequent in ER positive tumours; in their study, 45% of ER positive cases showed loss of 22q12, and 5% exhibited gain. In contrast, in ER negative tumours, loss or gain occurs with equal frequency of 20%. This information was used to add an additional arrow between HR status and 22q12, and the conditional probability table for 22q12 was amended to reflect this dependence using the frequency data.

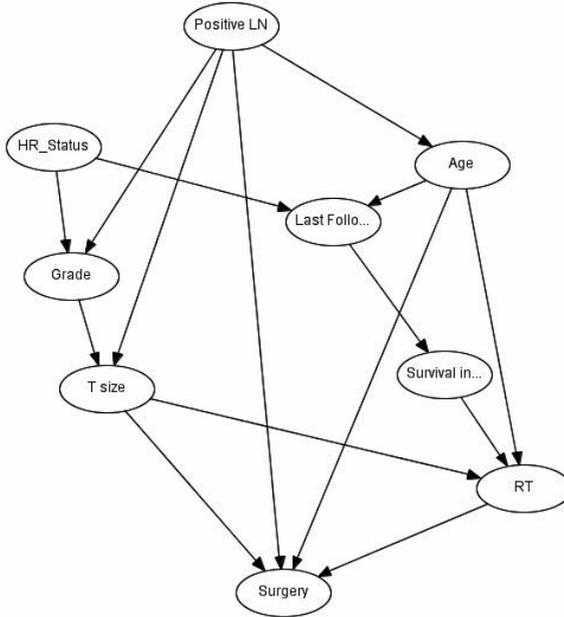


Fig. 6.2. The graph of the Bayesian net constructed from the clinical dataset

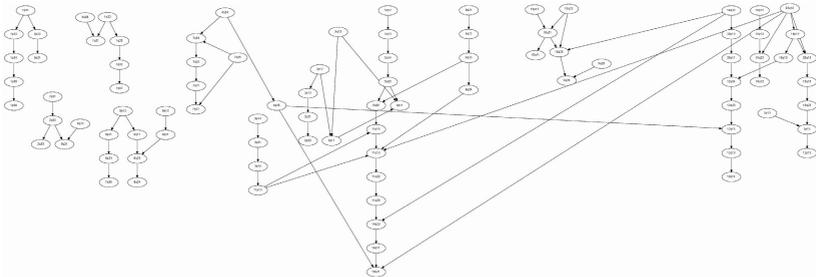


Fig. 6.3. The graph of the Bayesian net constructed from the large genomic dataset

dataset that represents $freq_m$, ensuring that the linked variable (22q12) is a root of the net (Fig. 6.3). Finally learn a bridging network, Fig. 6.4, from the smaller genomic dataset and the study, merge the three graphs to form one graph by merging identical variables, and integrate the conditional probability tables. Fig. 6.5 shows the graph of the resulting integrated obnet.

Here a conflict arose between the probability distribution determined by the clinical dataset and the probability distribution determined by the genomic dataset used to bridge the genomic and clinical variables: these gave different values to the probability of Positive LN. In §6.7, we pointed out that if neither dataset were to be preferred over the other, then the conflicting datasets

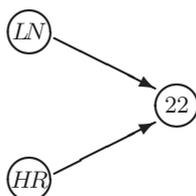


Fig. 6.4. The graph of the Bayesian net constructed from the smaller genomic dataset and the published study. The variables are Positive LN, HR status and 22q12.

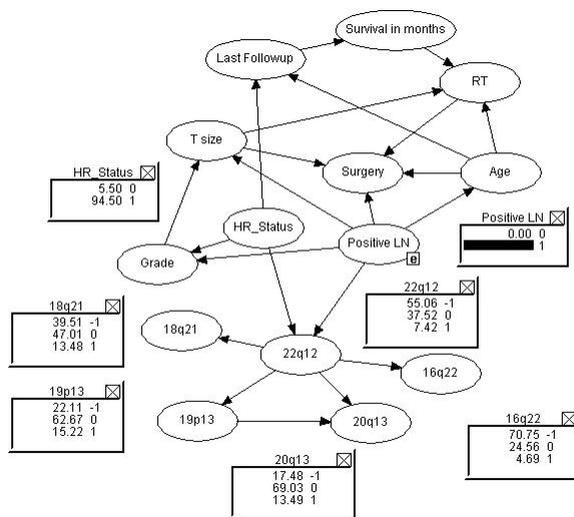


Fig. 6.5. The objective Bayesian net. Probability for positive lymph node status is set to 1 (black bar), and the calculated probability distributions for selected nodes are shown (HR status: 0 negative, 1 positive; chromosomal bands: -1 loss, +1 gain, 0 no rearrangement; RT radiotherapy).

constrain the probability of the assignment in question to lie within the closed interval bounded by the conflicting values; one should then take the least committal value in that interval. But in this case there are reasons to prefer one dataset over the other. First, the clinical dataset is based on a much larger sample than the bridging genomic dataset—for example, the clinical dataset has 2935 people with $LN = 0$, while the bridging genomic dataset has 56. Second, the molecular dataset has a clear sample bias: overall a frequency bias in favour of loss of 22q12 in breast cancer has been observed (20% loss vs. 7% gain in 800 cases in the progenetix database; accessed on the 14th of June 2006); furthermore, one may hypothesise that the presence of the KREMEN1 gene on 22q12 suggests that band loss, rather than gain, is more likely to be correlated with positive lymph node status, at least in certain karyotype contexts (§6.16). Thus the clinical data should trump the genomic data over the conflict on LN , i.e.,

Table 6.4. Probability table for Positive LN in the obnet

Positive LN	p
0	0.62
1	0.38

Table 6.5. Conditional probability table for 22q12 in the obnet

22q12	Positive LN	0	0	1	1
	HR_Status	0	1	0	1
-1		0.082	0.205	0.266	0.567
0		0.835	0.772	0.468	0.370
1		0.082	0.023	0.266	0.063

the probability table of *LN* in the obnet, Table 6.4, is simply inherited from the clinical net. The conditional probability table for 22q12 is depicted in Table 6.5.

Validation

Validation of our merged network is difficult; almost by definition, a suitable dataset involving the whole domain does not exist (if it did, we would not need to use this technique; we would simply use the dataset to learn a Bayesian net). Because of this, the best we were able to do was to use a small validation set with 11 test cases; validation showed reasonable agreement between the test cases and the obnet, but we must be careful not to over-interpret the results.

We approached the validation from two sides; the first was to set the 22q12 status and observe the effect on lymph node (LN) status; the second was to set the status of the lymph node variable, and observe the effect on 22q12 status. Unfortunately, the test set is both small and contains few cases of of 22q12 alteration.

Table 6.6. Setting 22q12 and observing LN status

22q12	LN status: Predicted from net	Actual	No.
1	0.62	0	1
0	0.5	0.55	9
-1	1	1	1

As can be seen from Table 6.6, only the linkage between no change of 22q12 and LN status predicted by the net is reflected in the validation set. It is difficult to interpret the results for the other values of 22q12.

Table 6.7. Setting LN status and observing 22q12

LN status	22q12:	Predicted from net	Actual	No.
+		0: 0.84, 1: 0.08, -1:0.08	0: 1.00	5
-		0: 0.95, 1: 0.05	0: 0.66, 1: 0.16, -1:0.16	6

Entering evidence in LN status, we have the results of Table 6.7. As we can see, both cases of LN status agree reasonably well with the observed cases, but again we must be careful not to over-interpret this relationship.

6.16 Interpretation of the Integrated Obnet

We have presented a general method for merging Bayesian networks which model knowledge in different areas. This method was applied to an example application linking knowledge about breast cancer genomics and clinical data. As a result, we have been able to examine the influence of karyotype pattern on clinical parameters (e.g., tumour size, grade, receptor status, likelihood of lymph node involvement) and vice versa (Fig. 6.5).

In the post-genome era, prognostic prediction and prediction of targets for new anti-cancer treatments from omic and clinical data are becoming ever more closely related—both need to relate molecular parameters to disease type and outcome. This correspondence is very clearly reflected in the uses to which the integrated obnet may be put. Obnet analysis may facilitate (i) discovery of genomic markers and signatures, and (ii) translation of clinical data to genomic research and discovery of novel therapeutic targets.

Discovery of Genomic Markers

Since hormone receptor status and lymph node involvement are well-known prognostic factors for survival and disease recurrence in patients with breast cancer, the ability to link karyotype patterns to this is clearly of great potential significance. Previous tumour genotyping in breast cancer has already shown the usefulness of genomic rearrangements as prognostic indicators.³⁷

For clinical decision making, this technique may also be useful when applied to integrate karyotype or other molecular data with parameters that cannot be observed in routine clinical practice, but are of clinical significance. An example might be the presence of distant metastasis on PET-CT, an imaging modality that may be present in the research setting but is not widely available in the clinic, but which may have prognostic significance for breast cancer recurrence. The use of such a net would then allow practitioners, where PET-CT is not available, to use genomic data to estimate the likelihood of a positive scan. There are of course, many different possible options for such networks, and it

³⁷ See, e.g., Al-Kuraya et al. (2004).

remains an open question as to which will, in clinical terms, prove to be the most useful.

Large clinical datasets are extremely expensive and difficult to collect. This is particularly true in diseases such as breast cancer, where the risk of recurrence extends up to at least 10 years, and hence requires long-term follow-up for accurate estimation. However, the generation of potential new predictive markers, such as genomic information or cell surface proteins, for exploration is currently a significant area of research. The correlation of such markers with better known clinical markers is (relatively) simple, in that it does not require long-term follow-up, and can be estimated following standard surgical treatment. However, for such information to be useful, it must be integrated with the existing databases on long-term outcomes, and it is this that we have demonstrated here.

Translation of Clinical Data to Genomic Research

The probabilistic dependence between 22q12 status and lymph node involvement was followed up by analysis of the genes with known function on this chromosomal band. This strongly suggested a causal interpretation of the dependency relationship based on knowledge of cellular pathways which regulate biological processes (mechanistic causation). KREMEN1 encodes a high-affinity dickkopf homolog 1 (DKK1) transmembrane receptor that functionally cooperates with DKK1 to block wntless (WNT)/beta-catenin signalling, a pathway which promotes cell motility.³⁸ Loss of 22q12 may therefore contribute to cancer cell migration through loss of the inhibiting KREMEN1 protein. The probability distribution for 22q12 is consistent with this hypothesis (Fig. 6.5).

In total, twelve genes implicated in cell migration and metastatic potential were identified on 22q12 and the other bands shown in Fig. 6.5. Like KREMEN1, the protein products of the other eleven genes can also be placed in the context of the metastatic pathways they participate in. Provided that appropriate kinetic interaction data are available, computational pathway modelling³⁹ can be employed to predict changes in pathway function resulting from the probabilistically dependent band gains and losses and concomitant changes in gene copy number. Molecularly targeted intervention strategies aimed at bringing about a therapeutic response in the cells so affected can be explored by running simulations using such pathway models. Simulation may be seen as being motivated by an agency-oriented notion of causation (see also §6.20).

Part IV: Further Development of the Method

6.17 Qualitative Knowledge and Hypotheses

There are various ways in which we intend to develop the method presented here.

³⁸ (Mao et al., 2002).

³⁹ (Alves et al., 2006).

First, as discussed in §6.11, there are a variety of knowledge sources that we hope to integrate. These include argumentation systems, medical ontologies, and causal relationships, as well as the clinical and molecular datasets which have been the focus of this chapter. In this Part, we shall discuss some of these other knowledge sources.

Second, as indicated in §6.16, we intend to exploit the objective Bayesian net that integrates these knowledge sources by using it not only for prognosis but also as a device for hypothesising new qualitative relationships amongst the variables under consideration. If the obnet reveals that two variables are probabilistically dependent, and that dependence is not explained by background knowledge, then we may hypothesise some new connection between the variables that accounts for their dependence. For example, we may hypothesise that the variables are causally (§6.20) or ontologically (§6.19) related. Furthermore, any such dependence can be used to generate qualitative arguments (§6.18): each variable will be an argument for or against the other, according to the direction of the dependence.

Third, we can increase the complexity of the formalism, in order to model temporal change or different levels of interaction, for instance. We shall discuss such extensions in §6.21.

These are avenues for future research. In this Part it will suffice to make some remarks on the likely directions that such research will take.

6.18 Argumentation

So far we have described how the network was developed, and analysed its performance as a Bayesian network. However, as suggested in §6.10 above, we are interested in more than just the probabilistic interpretation of the network—we are also interested in what the new network says about the world. Whereas in that section we suggested moving from qualitative to quantitative knowledge, here we shall discuss the opposite.

Bayesian Networks as Arguments

Bayesian networks are a useful tool for providing estimates of the probability for a variable of interest based on the evidence. Of course, Bayesian networks are not the only method of doing so, and there has been much work over the years on different formal methods to support decision making (rule-based systems, support vector machines, regression models, etc.). More generally, humans often use the notion of weighing up ‘the arguments’ for a belief or action to help them come to a conclusion. The argumentative method goes back at least 2500 years, and extends beyond the Graeco-Roman tradition.⁴⁰ Arguments have the advantage that they can present not only a conclusion, but also its justification. The idea of trying to base decision-making on arguments has a long history. The first clear example of an (informal) procedure for doing so was described by Benjamin Franklin.⁴¹

⁴⁰ (Gard, 1961).

⁴¹ (Franklin, 1887).

Of course, this informal notion of an argument can be neither implemented nor assessed in any rigorous fashion, but over the last 10-15 years there has been some work on developing competing formal (and computable) models of argument.⁴² More recent work has drawn together developments in non-monotonic logic and insights from cognitive science to produce a number of different argumentation frameworks.⁴³ We do not intend to present a review of the field here, but suffice to say that there are two general themes, *argument formation* and *argument resolution*. Each competing formalism defines these slightly differently, but in general an argument is a set of premises that allow one to deduce, via a set of rules, some set of conclusions. Resolution of competing arguments varies considerably between formalisms, is difficult to summarise in general terms, and matters less for our discussion here. However, what interests us is how one can interpret our new Bayesian network in terms of arguments. In other words, given a Bayesian network, what can we say about the arguments for and against a set of propositions, and given a new Bayesian network (formed from two or more existing ones) what new arguments can we make? Two of the authors of this paper have previously presented a simple technique for developing arguments from a Bayesian network,⁴⁴ basing the arguments on a relatively simple argumentation formalism,⁴⁵ and we use the method outlined there to develop our arguments from the Bayesian network. For reasons of space, we do not present the details of our method here; they can be found in Williams and Williamson (2006). Instead, let us consider what it might mean, in general terms, for a probabilistic statement to be interpreted as an argument. Firstly, therefore, let us consider what we mean by an argument.

Rules, Arguments and Probability

Intuitively, an argument is a line of reasoning that proceeds from some premise(s) via a set of deductions to some conclusion. As we all know, arguments are defeasible—that is, their conclusions may at some point be challenged and what was at some point held to be ‘true’ by argument may later be found to be untrue. We can formalise this ‘method of argument’ in various different ways (as mentioned above) but in general we have a quartet of premises, rules (for deduction), conclusions, and conflict between arguments. In order to map probabilistic statements into an argumentative framework, therefore, we need to consider how different aspects of a Bayesian system map into the quartet of argumentation, and what effects this has. We shall do this below, but first we need to establish a (fairly trivial) mapping between Bayesian notation and argumentative notation. We do this by considering all variables to be binary in nature, and each node in the network to represent a single binary-valued variable. The mapping between such a network and a truth-valued logic (say, propositional logic) should be clear: for any variable X , $p(X = 1)$ is interpreted as x and $p(X = 0)$ is interpreted as $\neg x$.

⁴² (Fox and Parsons, 1997).

⁴³ (Amgoud et al., 2004; Hunter and Besnard, 2001; Krause et al., 1995).

⁴⁴ (Williams and Williamson, 2006).

⁴⁵ (Prakken and Sartor, 1996).

Given the correspondence above, mapping the premises is fairly simple: they are the inputs to the network, i.e., the variables in the net that are instantiated. Similarly, the conclusions are also relatively simple—they are the values of other nodes in the network. Given any two nodes in a Bayesian network, the absence of any connection between them implies probabilistic independence, and therefore precludes one being an argument for the other. The presence of a connection suggests that there may be a relationship between them. It is this relationship that we interpret as forming the ‘rules’⁴⁶ for an argumentation system, and in its most basic form, if the truth of one variable A increases the probability of another variable B being true, then we might write $a \Rightarrow b$. An argument is the association of a set of premises and rules that lead to a conclusion⁴⁷—e.g., $\langle \{a, a \Rightarrow b\}, b \rangle$, where $\langle \rangle$ denotes the argument, the first element $\{a, a \Rightarrow b\}$ is the support and b is the conclusion. Arguments are in conflict if they argue for conclusions which are mutually exclusive—so if we had another argument $\langle \{c, c \Rightarrow \neg b\}, \neg b \rangle$, this would be in conflict with our first argument.

Our Network as Arguments

We are now in a position to return to the first question we posed above—‘what can we say about the arguments for and against a set of propositions?’ The first thing to observe is that our approach will only allow us to develop arguments about literals that correspond to nodes in the network. Secondly, we can only develop rules between literals that are linked in the network. Thus, while *we* might know of some connection, that connection will not appear unless it also appears as a conditional dependence (and hence a link) in our network. This is why the procedure outlined in §6.10 is so important: all background knowledge must be taken into account in the construction of the objective Bayesian net. Of course, in general we might add some additional rules (from other sources), but the rules (and hence the arguments) developed from the network will only be concerned with those literals that appear and are associated in the network. Thirdly, given the dichotomised nature of the variables, we have a tendency to develop arguments both for and against literals. We can see this from the following example. The CPT for the Tumour size node from our clinical network is shown in Table 6.8. From these values we can calculate that $p(T_Size = 0-20)$ is 0.657, whilst $p(T_Size = 0-20 | LN = 0)$ is 0.753, and $p(T_Size = 0-20 | LN = 1)$ is 0.5. Therefore, following the method outlined above, we can see that we should develop the following rules:

- $(LN = 0) \Rightarrow (T_Size = (0-20))$
- $(LN = 1) \Rightarrow \neg(T_Size = (0-20))$

On the one hand, this may seem to be problematic—the generation of pairs of opposing rules (and hence arguments) might lead us to some sort of deadlock.

⁴⁶ A ‘rule’ in this context is a defeasible piece of knowledge that allows one to infer the value of one variable from another.

⁴⁷ Most systems, including ours, also impose further restrictions to ensure arguments that are consistent and non-circular.

Table 6.8. Conditional probability table for Tumour Size

<i>T_Size</i> (mm)	Grade	1	1	2	2	3	3
	Positive LN	0	1	0	1	0	1
0—20		0.85	0.66	0.76	0.5	0.62	0.42
20—50		0.14	0.33	0.21	0.5	0.35	0.58
50—150		0.005	0	0.02	0	0.03	0
No.		689	271	1668	990	578	535

However, this is not a bad as it may seem. Firstly, it seems intuitively correct to develop rules for both options—after all, the whole point of the Bayesian net is that it contains information about both options. Secondly, the ‘deadlock’ between the rules can be resolved in a variety of ways (for example, we could encode the likelihood of the different options as ‘weights’ given to the rules). Thirdly, the point of the rules, and arguments, is to allow us to help integrate human decision-making with our Bayesian techniques, and to allow us to do this we need to display arguments for both options, even if they have different weights. Finally, in this case it would of course be impossible to have a measurement for both $LN = 0$ and $LN = 1$ at the same time (although we might have other *arguments* for both at the same time, as we shall see below).

New Arguments from New Networks

The second question we asked above was ‘given a new Bayesian network...what new arguments can we make?’ In a sense, the answer is (almost) ‘none’. After all, as we said above, all the rules are developed from existing literals and relationships in a Bayesian network. Since our new network is only a combination of the existing networks, there should not be anything new. However, this answer misses one of the aspects that is crucial to the difference between argumentation and Bayesian networks.



Fig. 6.6. The graph of a Bayesian network

One of the key features of Bayesian networks, as mentioned in §6.8 is that each variable is probabilistically independent of its non-descendants conditional on its parents, and as is noted above, this has some very desirable properties from a computational aspect. However, once we consider developing arguments from our network, we see that this relationship comes out differently when applied to arguments. For example, consider a (very simple) Bayesian net, whose graph is shown in Fig. 6.6. Depending on the probabilities in the net, we may develop the rules:

- $(A = 1) \Rightarrow (B = 1)$ which we write as $a \Rightarrow b$
- $(B = 1) \Rightarrow (C = 1)$ which we write as $b \Rightarrow c$

- $(A = 0) \Rightarrow (B = 0)$ which we write as $\neg a \Rightarrow \neg b$
- $(B = 0) \Rightarrow (C = 0)$ which we write as $\neg b \Rightarrow \neg c$

Now, according to the Bayesian network, B screens C off from A . However, under our argumentation formalism, we might have the following arguments:

- $A_1: \langle \{a, a \Rightarrow b, b \Rightarrow c\}, c \rangle$ as an argument for c
- $A_2: \langle \{\neg a, \neg a \Rightarrow \neg b, \neg b \Rightarrow \neg c\}, \neg c \rangle$ as an argument for $\neg c$

which seems to make explicit the dependence of c on a . Now, if we do not know the status of b , then in both formalisms, we understand that in fact the best guide to the state of c is the state of a , and so the two approaches are in agreement. If we know both a and b , and they are ‘concordant’ (e.g. a and b or $\neg a$ and $\neg b$) then we will find that indeed a is ‘redundant’, and c is entirely determined by b . However, our work is motivated by the fact that our knowledge is often partial and conflicting. For example, we might have one piece of information about a and another about b , and they may conflict—for example, we may believe both $\neg a$ and b . In such a situation, the Bayesian net approach would typically discard the information about a , as it would be over-ridden by the information about b . Under an argumentative approach, however, we are able to construct arguments for *both* c and $\neg c$, as shown below:

- $A_1: \langle \{b, b \Rightarrow c\}, c \rangle$
- $A_2: \langle \{\neg a, \neg a \Rightarrow \neg b, \neg b \Rightarrow \neg c\}, \neg c \rangle$

Thus the argumentation differs from the Bayesian net in that it does not follow the probabilistic independencies of the net. Obviously at some point we will need to resolve this disagreement, but we can at least start by considering both cases. The Bayesian net approach retains probabilistic validity, but only by enforcing a set of strict rules, one of which is committing to a particular value of certain variables (b in our example); the argumentative approach loses this precision, but has the advantage that it can handle conflicting premises and generate arguments based on them, which it then resolves, rather than losing this information (b and $\neg a$ in our example). Such differences are not unique to our particular brand of argumentation (for example, they are seen in Parsons’ qualitative probability framework,⁴⁸ which devotes a considerable amount of space to discussing the problem).

We are now finally in a position to answer our second question. When we add a new network, we can still only develop the same rules that we developed in each network. However, because we can use the rules to form arguments, we can form arguments that are ‘bigger’ than those formed in either network alone. For example, consider our merged network, Fig. 6.5. In this case, we can see that 22q12 is connected to lymph node status, and we would have been able to form an argument linking the two from the genomic net alone. However, given the links in the network, we can form an argument (but not a valid probabilistic relationship) which would link lymph node status and 19p13 status, even if we

⁴⁸ (Parsons, 2003, 2004).

know 22q12 status. Such an argument is not interpreted probabilistically but may still be useful for explaining the links to human users.

6.19 Ontologies

In §6.11 we mentioned some of the different types of knowledge that we might try and integrate into our Bayesian network; one important category is ontological knowledge. Ontologies (formal representations of vocabularies and concept relationships) and common data elements support automated reasoning including artificial intelligence methods such as Bayesian networks. Various standard vocabularies and object models have already been developed for genomics, molecular profiles, certain molecular targeted agents, mouse models of human cancer, clinical trials and oncology-relevant medical terms and concepts (SNOMED-RT/CT, ICD-O-3, MeSH, CDISC, NCI Health Thesaurus, caCORE, HUGO). There are also existing ontologies describing histopathology (standards and minimum datasets for reporting cancers, Royal College of Pathologists; caIMAGE, NCI). The European Bioinformatics Institute (EBI) is developing standards for the representation of biological function (Gene Ontology) and the Microarray Gene Expression Data (MGED) Society is developing MIAME, MAGE, and the MAGE ontology, a suite of standards for microarrays (transcriptomic, array CGH, proteomic, SNP). However, significant gaps still exist and eventually, all cancer-relevant data types (see the NCRI Planning Matrix, www.cancerinformatics.org.uk/planning_matrix.htm) will need to be formalised in ontologies. These efforts are ongoing and pursued by a large community of researchers (see above, and [ftp1.nci.nih.gov/pub/cacore/ ExternalStds/](http://ftp1.nci.nih.gov/pub/cacore/ExternalStds/) for further details on available standards). Clearly, it would be desirable to incorporate the fruits of these efforts in our scheme for knowledge integration; the potential for using ontologies as a knowledge source will increase with the maturation of these other initiatives.

While we would like to base our objective Bayesian net on ontological knowledge as well as our other knowledge sources, we also believe that we could establish some possible ontological relationships from our Bayesian network. The most obvious of these is in establishing a sub-/super-class relationship between two variables. For example, imagine a dataset which recorded both whether someone had had breast cancer and if they had had each individual subtype of breast cancer (and also included those without breast cancer). Such a network would contain several nodes, but in each case, if any subtype of cancer was positive, then the ‘has cancer’ variable would also be positive. Such patterns of conditional dependence may be complex—in our example, there would be several different nodes linking to the ‘has cancer’ node, but in general are indicative of the presence of a sub-/super-class relationship (where the dependent variable is the superclass). We may take this idea further by suggesting that if there are certain combinations of variables that (together) are highly predictive of another variable, we might regard those individuals as acting as a ‘definition’ of the outcome variable. For example, consider a dataset which records whether

individuals (of different species) are human or not, whether they are women or men, and if they have XX or XY chromosomes. Now, those individuals with XX chromosomes are of course women, and so all individuals who are human and have XX chromosomes will also be women. From this, we might deduce that in fact, the two are equivalent, and thus being human and having XX chromosomes is the same as being a woman. This approach is, to say the least, prone to error, but provides a way to start learning such definitions from data, something that currently has to be done by hand.

In an ideal world, we would expect these relationships to be absolute, but in reality, we should allow for there being some ‘noise’ in the data, and may well be willing to accept a near-absolute (say 95% or 98%) as being suggestive of such a link. However, this is *not* the same as saying that the ontological relationship is probabilistic, as some authors do. Instead, it is based upon a supposition that the ontological relationship is absolute, but that the data may imperfectly reflect this. Interestingly, however, we seem to rarely see this sort of relationship in our networks. The reason for this is that the resolution of ontological relationships is one of the things that we tend to do at either the data collection or pre-processing stage. For example, as a part of our pre-processing of the data we combined the Oestrogen and Progesterone receptor status into a new variable, Hormone Receptor status, where the class of Oestrogen and Progesterone receptors are subclasses of Hormone Receptors. However, this is not to say that such relationships will never be important. One of the aims of the semantic web, and science on the semantic web, is to enable large amounts of data to be shared; such sharing will necessitate automatic handling of data (as manual processing of larger and larger databases becomes harder and harder), and tools for the handling of data may be able to use such strong probabilistic relationships to highlight potential ontological issues to the user.

6.20 Causal Relationships

Concepts of Causation in Complex Systems

Since each patient’s cells evolve through an independent set of mutations and selective environments, the resulting population of cancer cells in each patient is likely to be unique in terms of the sum total of the mutational changes they have undergone. Inter-personal variability has given rise to the new field of ‘pharmacogenomics’ (in cancer and other diseases) which has as its ultimate aim diagnostic and prognostic prediction, and design of individualised treatments based on patient-specific molecular characteristics. Given the prevailing high degree of uncertainty in the face of biological complexity, pharmacogenomics offers great promise, but is also ‘high risk’. Risk has, for example, been highlighted by recent findings of the nonreproducibility of classification based on gene expression profiling (expression microarrays, transcriptomics).⁴⁹ In this situation, diagnosis and prognosis based on biomarkers or profiles of multiple molecular indicators

⁴⁹ See, for example, Michielsen et al. (2005).

may lead to mis-classification, and may identify patients as likely non-responders to a given treatment when in fact they would derive benefit or, conversely, may falsely predict efficacy in patients for whom none can be achieved. One may argue that this uncertainty, at least in part, is compounded by prevailing notions of biological causality which is still preoccupied with the search for single (or a small number of) physical causes, and a failure to take into account the characteristics of complex systems.

Different views on the nature of causality lead to different suggestions for discovering causal relationships.⁵⁰ Medicine has been, due to its very nature, particularly focused on an agency-oriented account of causality which seeks to analyse causal relations in terms of the ability of agents (doctors, health professionals and scientists) to achieve goals (cures, amelioration of symptoms) by manipulating their causes. According to this conception of causality, *C* causes *E* if and only if bringing about *C* would be an effective way of bringing about *E*. Or conversely, for example, in the context of therapeutic intervention, *C* is seen as a cause of *E* if by inhibiting *C* one can stop *E* from happening. In this intervention-oriented stance, the agent would also seek to ground this view of causality in a mechanistic account of physical processes, as, for example, in the mechanistic mode of action of a drug. In diagnostic and prognostic prediction from patient data, a causal framework is also implied; here, causation may be conceptualised as agency-based, mechanistic or in terms of a probabilistic relationship between variables. However, the extensive literature on the subject reveals a number of problems associated with all three approaches.⁵¹

An alternative view of causality, termed *epistemic causality* by Williamson (2005a), overcomes the strict compartmentalisation of current theories of causation, and focuses on causal beliefs and the role that *all* of these indicators (mechanistic, probabilistic, agency-based) have in forming them. It takes causality as an objective notion yet primarily a mental construct, and offers a formal account of how we ought to determine causal beliefs.⁵² This approach will be applied to glean causal hypotheses from an obnet, as outlined below.

We are faced with a profound challenge regarding causation in complex biological systems. In her discussion of developmental systems and evolution, Susan Oyama observes ‘what a cause causes is contingent and is thus itself caused’.⁵³ The influence of a gene, or a genetic mutation, depends on the context, such as availability of other molecular agents and the state of the biological system, including the rest of the genome. Oyama argues for a view of causality which gives weight to all operative influences, since no single influence is sufficient for a biological phenomenon or for any of its properties. Variation in any one influence, or many of them, may or may not bring about variation in the result, depending on the configuration of the whole. The *mutual dependence of (physical) causes* leads to a situation where an entire ensemble of factors contribute to any given

⁵⁰ (Williamson, 2007a).

⁵¹ (Williamson, 2005a; Williamson, 2007a, and references therein).

⁵² (Williamson, 2007a).

⁵³ (Oyama, 2000, pp. 17–18 and references therein).

phenomenon, and the effect of any one factor depends both on its own properties and on those of the others, often in complex combinations. This gives rise to the concept of organised *causal networks* and is a central insight of systems thinking. The biological relevance of any factor, and therefore the information it conveys, is jointly determined, typically in a statistically interactive fashion, by that factor and the entire system's state.⁵⁴

Whilst 'systems thinking' is likely to be fundamental for biomedicine and for cancer, in particular, due to its overwhelming complexity, we still lack a principled methodology for addressing these questions. Methodology development is a pressing need, and it is with this major objective in mind that our research is undertaken. The work presented here combines a multidisciplinary framework of biological systems theory and objective Bayesian network modelling; our next step will be to integrate epistemic causality into this framework.

Here, it may be helpful, or even necessary, to draw a distinction between fundamental science and applied biomedical research. In systems biology, the ultimate goal may be to gain a complete mechanistic explanation of the system complexity underlying causal networks. The achievement of this aim still lies in the, possibly far distant, future. In contrast, in biomedical research and clinical practice, we tend to be more immediately interested in discovering molecular predictors for diagnostic and prognostic purposes, and in developing effective strategies for intervention in malfunctioning body systems and disease processes.

Perhaps surprisingly, an applied focus of this kind may work in our favour vis-à-vis biological complexity, as progress is not as severely constrained by a requirement for an exhaustive mechanistic elucidation of the complete system. In this chapter we sketch a discovery strategy which is based on the epistemic view of causality (see below and §6.16). The strategy integrates *probabilistic* dependency networks (Bayesian networks) with expert knowledge of biological *mechanisms*, where available, to hypothesise causal networks inherent in the system. This approach enables one to predict probabilistic biomarker profiles and targets for intervention based on the identified dependencies between system components. Interventions may then be tested by computational modelling and experimental validation which may be seen as foregrounding '*agency-based*' causation. This is a pragmatic strategy which can yield insights into system function which are attainable now and are valuable from a biomedical point of view.

Gleaning Causal Relationships from an Obnet

The epistemic theory of causality maintains the following.⁵⁵ Just as, under the objective Bayesian account, an agent's rational degrees of belief should take the form of a probability function objectively determined by her evidence, so too her rational causal beliefs, represented by a directed acyclic graph (*dag*), are objectively determined by her evidence. An agent should adopt, as her causal belief graph, the most non-committal graph (i.e., the dag with fewest arrows) that satisfies constraints imposed by her evidence.

⁵⁴ (Oyama, 2000, p. 38).

⁵⁵ (Williamson, 2005a, Chapter 9).

Now, evidence imposes constraints in several ways. (i) The agent may already know of causal connections, in which case her causal graph should contain the corresponding arrows. (ii) She may know that A only occurs after B , in which case her causal graph should not contain an arrow from A to B . (iii) Or a causal connection from A to B may be incompatible with her scientific knowledge inasmuch as her scientific knowledge implies that there is no physical mechanism from A to B and hence no possible physical explanation of B that involves A ; then there should be no arrow from A to B in her causal belief graph. (iv) Or there may be a *strategic dependence* from A to B (i.e., A and B may be probabilistically dependent when intervening to fix A and controlling for B 's other causes) for which the agent has no explanation in her background knowledge; she should then have an arrow from A to B in her causal graph to explain the dependence, as long as other knowledge does not rule out such an arrow.

One can determine the agent's causal belief graph by running through all dags and taking a minimal graph that satisfies the constraints (i–iv) imposed by background knowledge; but such a method is clearly computationally intractable. Two other, more feasible methods are worth investigating. The agent's causal belief graph can be approximated by a minimal dag that satisfies the Markov condition and constraints of type (i–iii); standard Bayesian net software can be used to construct such a graph. Or one can generate an approximation to the causal belief graph by constructing a graph that satisfies constraints (i–iii) and incrementally adding further arrows (also satisfying these constraints) that correspond to strategic dependences in the obnet. These extra arrows are causal hypotheses generated by the objective Bayesian net.

6.21 Object-Oriented, Recursive and Dynamic Obnets

Another avenue for future research concerns extensions of the objective Bayesian net framework to cope with object orientation, recursion and temporal change.

Object-oriented and dynamic Bayesian networks possess certain advantages for the modelling of complex biological systems. For example, Dawid, Mortera and Vicard have applied OOBNs to the domain of genetics and complex forensic DNA profiling⁵⁶ and Bangsø and Olesen showed how OOBNs can be adapted to dynamically model processes over time, such as glucose metabolism in humans.⁵⁷

Object-oriented Bayesian networks (OOBNs) allow one to represent complex probabilistic models.⁵⁸ Objects can be modelled as composed of lower-level objects, and an OOBN can have nodes that are themselves instances of other networks, in addition to regular nodes. In an OOBN, the internal parts of an object can be encapsulated within the object. Probabilistically, this implies that the encapsulated attributes are d -separated from the rest of the network by the object's inputs and outputs. This separation property can be utilised to locally

⁵⁶ (Dawid et al., 2007).

⁵⁷ (Bangsø and Olesen, 2003).

⁵⁸ (Koller and Pfeffer, 1997; Laskey and Mahoney, 1997).

constrain probabilistic computation within objects, with only limited interaction between them.

By representing a hierarchy of inter-related objects, an OOBN makes organisational structure explicit. OOBN ‘is-a’ and ‘part-of’ hierarchical structuring mirrors the organisation of ontologies in the biomedical knowledge domain (see §6.19), and ontologies can therefore be used as background knowledge to structure an OOBN.

Recursive Bayesian networks offer a means of modelling a different kind of hierarchical structure—the case in which variables may themselves take Bayesian networks as values.⁵⁹ This extra structure is required, for instance, to cope with situations in which causal relationships themselves act as causes and effects. This is often the case with policy decisions: e.g., the fact that smoking causes cancer causes governments to restrict tobacco advertising.

The timing of observations (e.g., symptoms, measurements, tests, events) plays a major role in diagnosis, prognosis and prediction. Temporal modelling can be performed by a formalism called Temporal Bayesian Network of Events (TBNE).⁶⁰ In a TBNE each node represents an event or state change of a variable, and an arc corresponds to a causal-temporal relationship. A temporal node represents the time that a variable changes state, including an option of no-change. The temporal intervals can differ in number and size for each temporal node, so this allows multiple granularity. The formalism of dynamic Bayesian nets can also be applied.⁶¹

OOBN properties allow one to exploit the modular organisation of biological systems for the generation of complex models. To our knowledge, OOBNS have not been applied to systems-oriented cancer modelling. We aim to assess the usefulness of OOBN methods for multi-scale models of cancer systems, especially to represent variables associated with heterogeneity in tumours. Our research will also evaluate the uses of the TBNE formalism and dynamic Bayesian nets for temporal models of karyotype evolution (§§6.2, 6.3) and evolving therapeutic systems (patient/tumour-therapy-response).

In sum, then, there are a variety of situations which call for a richer formalism. Since an obnet is a Bayesian net, one can enrich an obnet using all the techniques available for enriching Bayesian nets: one can render an obnet object-oriented, recursive or dynamic. The details of these extensions are questions for further work.

6.22 Conclusion

In this chapter we have presented a scheme for systems modelling and prognosis in breast cancer. A multiplicity of knowledge sources can be integrated by forming the objective Bayesian net generated by this evidence. This obnet represents the probabilistic beliefs that should be adopted by an agent with that evidence;

⁵⁹ (Williamson and Gabbay, 2005).

⁶⁰ (Arroyo-Figueroa and Sucar, 2005).

⁶¹ (Neapolitan, 2003).

Table 6.9. The interplay between evidence and belief

Evidence	↔	Belief
Clinical data		Probabilistic (obnet)
Genomic data		Argumentative
Published studies		Ontological
Argumentation systems		Causal
Medical ontologies		
Causal knowledge		
Biological theory		

it can be used to assist prognosis of cancer patients. The obnet together with evidence can, in turn, be used to generate sets of argumentative, ontological and causal beliefs. These are just hypotheses and require testing; more data must be collected to confirm or disconfirm these hypotheses. These new data increase the base of evidence and consequently new beliefs (probabilistic, causal and so on) must be formulated. We thus have a dialectical back-and-forth between evidence and belief, as depicted in Table 6.9.

This iterative approach to knowledge discovery facilitates novel insights and hypotheses regarding the organisation and dynamic functioning of complex biological systems, and can lead to fruitful discovery from limited data. Objective Bayesian nets thus provide a principled and practical way of integrating domain knowledge, and of using it for inference and discovery.

Acknowledgements

This research was carried out as part of the caOBNET project (www.kent.ac.uk/secl/philosophy/jw/2006/caOBNET.htm). We are very grateful to Nadjet El-Mehidi and Vivek Patkar for their assistance with this work. For financial support we are grateful to Cancer Research UK, the Colyer-Fergusson Awards of the Kent Institute for Advanced Study in the Humanities and the Leverhulme Trust.

References

- Abramovitz, M., Leyland-Jones, B.: A systems approach to clinical oncology: Focus on breast cancer. *BMC Proteome Science* 4, 5 (2006)
- Al-Kuraya, K., Schraml, P., Torhorst, J., Tapia, C., Zaharieva, B., Novotny, H., Spichtin, H., Maurer, R., Mirlacher, M., Kochl, O., Zuber, M., Dieterich, H., Mross, F., Wilber, K., Simon, R., Sauter, G.: Prognostic relevance of gene amplifications and coamplifications in breast cancer. *Cancer Research* 64, 8534–8540 (2004)
- Alves, R., Antunes, F., Salvador, A.: Tools for kinetic modeling of biochemical networks. *Nature Biotechnology* 24, 667–672 (2006)
- Amgoud, L., Cayrol, C., Lagasquie-Schiex, M.-C.: On bipolarity in argumentation frameworks. In: *NMR*, pp. 1–9 (2004)
- Arroyo-Figueroa, G., Sucar, L.: Temporal Bayesian network of events for diagnosis and prediction in dynamic domains. *Applied Intelligence* 23, 77–86 (2005)

- Bangsø, O., Olesen, K.: Applying object oriented Bayesian networks to large (medical) decision support systems. In: Proceedings of the Eighth Scandinavian Conference on Artificial Intelligence. IOS Press, Amsterdam (2003)
- Baudis, M., Cleary, M.: Progenetix.net: an online repository for molecular cytogenetic aberration data. *Bioinformatics* 17, 1228–1229 (2001)
- Borak, J., Veilleux, S.: Errors of intuitive logic among physicians. *Soc. Sci. Med.* 16, 1939–1947 (1982)
- Bulashevskaya, S., Szakacs, O., Brors, B., Eils, R., Kovacs, G.: Pathways of urothelial cancer progression suggested by Bayesian network analysis of allelotyping data. *International Journal of Cancer* 110, 850–856 (2004)
- Cristofanilli, M., Hayes, D., Budd, G., Ellis, M., Stopeck, A., Reuben, J., Doyle, G., Madera, J., Allard, W., Miller, M., Fritsche, H., Hortobagyi, G., Terstappen, L.: Circulating tumor cells: A novel prognostic factor for newly diagnosed metastatic breast cancer. *J. Clin. Oncol.* 23, 1420–1430 (2005)
- Dawid, A., Mortera, J., Vicard, P.: Object-oriented Bayesian networks for complex forensic DNA profiling problems. *Forensic Science International* 169(256), 195–205 (2007)
- Depew, D., Weber, B.: Darwinism evolving: systems dynamics and the genealogy of natural selection. MIT Press, Cambridge (1996)
- Fox, J., Parsons, S.: On using arguments for reasoning about actions and values. In: Proc. AAAI Spring Symposium on Qualitative Preferences in Deliberation and Practical Reasoning, Stanford (1997)
- Franklin, B.: Collected Letters, Putnam, New York (1887)
- Fridlyand, J., Snijders, A., Ylstra, B., Li, H., Olshen, A., Segev, R., Dairkee, S., Tokuyasu, T., Ljung, B., Jain, A., McLennan, J., Ziegler, J., Chin, K., Devries, S., Feiler, H., Gray, J., Waldman, F., Pinkel, D., Albertson, D.: Breast tumor copy number aberration phenotypes and genomic instability. *BMC Cancer* 6, 96 (2006)
- Galea, M., Blamey, R., Elston, C., Ellis, I.: The Nottingham Prognostic Index in primary breast cancer. *Breast Cancer Research and Treatment* 3, 207–219 (1992)
- Gard, R.: Buddhism. George Braziller Inc., New York (1961)
- Holland, J.: Hidden order: how adaptation builds complexity. Helix Books, New York (1995)
- Holland, J.: Emergence: from chaos to order. Addison-Wesley, Redwood City (1998)
- Hunter, A., Besnard, P.: A logic-based theory of deductive arguments. *Artificial Intelligence* 128, 203–235 (2001)
- Jaynes, E.T.: Information theory and statistical mechanics. *The Physical Review* 106(4), 620–630 (1957)
- Kahneman, D., Tversky, A.: On the psychology of prediction. *Psychol. Rev.* 80, 237–251 (1973)
- Khalil, I., Hill, C.: Systems biology for cancer. *Curr. Opin. Oncol.* 17, 44–48 (2005)
- Kitano, H.: Biological robustness. *Nat. Rev. Genet.* 5, 826–837 (2004)
- Koller, D., Pfeffer, A.: Object-oriented Bayesian networks. In: Geiger, D., Shenoy, P. (eds.) Proceedings of the 13th Annual Conference on Uncertainty in Artificial Intelligence, pp. 302–313. Morgan Kaufmann Publishers, San Francisco (1997)
- Korb, K.B., Nicholson, A.E.: Bayesian artificial intelligence. Chapman and Hall / CRC Press, London (2003)
- Krause, P., Ambler, S., Elvang-Goransson, M., Fox, J.: A logic of argumentation for reasoning under uncertainty. *Computational Intelligence* 11, 113–131 (1995)

- Laskey, K., Mahoney, S.: Network fragments: Representing knowledge for constructing probabilistic models. In: Geiger, D., Shenoy, P. (eds.) *Proceedings of the 13th Annual Conference on Uncertainty in Artificial Intelligence*, pp. 334–341. Morgan Kaufmann Publishers, San Francisco (1997)
- Lupski, J., Stankiewicz, P.: *Genomic disorders: The genomic basis of disease*. Humana Press, Totowa (2006)
- Mao, B., Wu, W., Davidson, G., Marhold, J., Li, M., Mechler, B., Delius, H., Hoppe, D., Stannek, P., Walter, C., Glinka, A., Niehrs, C.: Kremen proteins are Dickkopf receptors that regulate Wnt/beta-catenin signalling. *Nature* 417, 664–667 (2002)
- McPherson, K., Steel, C., Dixon, J.: Breast cancer: Epidemiology, risk factors and genetics. *BMJ* 321, 624–628 (2000)
- Michielsen, S., Koscielny, S., Hill, C.: Prediction of cancer outcome with microarrays: a multiple random validation strategy. *The Lancet* 365(9458), 488–492 (2005)
- Mitchell, S.: *Biological complexity and integrative pluralism*. Cambridge University Press, Cambridge (2003)
- Nagl, S.: Objective Bayesian approaches to biological complexity in cancer. In: Williamson, J. (eds.) *Proceedings of the Second Workshop on Combining Probability and Logic*. (2005), www.kent.ac.uk/secl/philosophy/jw/2005/progic/
- Nagl, S.: A path to knowledge: from data to complex systems models of cancer. In: Nagl, S. (ed.) *Cancer Bioinformatics*, pp. 3–27. John Wiley & Sons, London (2006)
- Nagl, S., Williams, M., El-Mehidi, N., Patkar, V., Williamson, J.: Objective Bayesian nets for integrating cancer knowledge: a systems biology approach. In: Rouso, J., Kaski, S., Ukkonen, E. (eds.) *Proceedings of the Workshop on Probabilistic Modelling and Machine Learning in Structural and Systems Biology*, Tuusula, June 17–18 2006, vol. B-2006-4, pp. 44–49. Helsinki University Printing House, Finland (2006)
- Neapolitan, R.E.: *Probabilistic reasoning in expert systems: theory and algorithms*. Wiley, New York (1990)
- Neapolitan, R.E.: *Learning Bayesian networks*. Pearson / Prentice Hall, Upper Saddle River (2003)
- Nygren, P., Larsson, R.: Overview of the clinical efficacy of investigational anticancer drugs. *Journal of Internal Medicine* 253, 46–75 (2003)
- Oyama, S.: *The ontogeny of information: developmental systems and evolution*, 2nd edn. Duke University Press, Durham (2000)
- Parsons, S.: Order of magnitude reasoning and qualitative probability. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 11(3), 373–390 (2003)
- Parsons, S.: On precise and correct qualitative probabilistic reasoning. *International Journal of Approximate Reasoning* 35, 111–135 (2004)
- Prakken, H., Sartor, G.: Argument-based extended logic programming with defeasible priorities. In: Schobbens, P.-Y. (ed.) *Working Notes of 3rd Model Age Workshop: Formal Models of Agents*, Sesimbra, Portugal (1996)
- Quinn, M., Allen, E.: Changes in incidence of and mortality from breast cancer in England and Wales since introduction of screening. *BMJ* 311, 1391–1395 (1995)
- Rasnick, D., Duesberg, P.: How aneuploidy affects metabolic control and causes cancer. *Biochemical Journal* 340, 621–630 (1999)
- Ravdin, Siminoff, Davis.: A computer program to assist in making decisions about adjuvant therapy for women with early breast cancer. *J. Clin. Oncol.* 19, 980–991 (2001)

- Reis-Filho, J., Simpson, P., Gale, T., Lakhan, S.: The molecular genetics of breast cancer: the contribution of comparative genomic hybridization. *Pathol. Res. Pract.* 201, 713–725 (2005)
- Richards, M., Smith, I., Dixon, J.: Role of systemic treatment for primary operable breast cancer. *BMJ* 309, 1263–1366 (1994)
- Ries, L., Eisner, M., Kosary, C., Hankey, B., Miller, B., Clegg, L., Mariotto, A., Feuer, E., Edwards, B.: SEER Cancer Statistics Review 1975-2001. National Cancer Institute (2004)
- Russo, F., Williamson, J.: Interpreting probability in causal models for cancer. In: Russo, F., Williamson, J. (eds.) *Causality and probability in the sciences*. Texts in Philosophy, pp. 217–241. College Publications, London (2007)
- Toyoda, T., Wada, A.: ‘omic space’: coordinate-based integration and analysis of genomic phenomic interactions. *Bioinformatics* 20, 1759–1765 (2004)
- Veer, L., Paik, S., Hayes, D.: Gene expression profiling of breast cancer: a new tumor marker. *J. Clin. Oncol.* 23, 1631–1635 (2005)
- Vogelstein, B., Kinzler, K.: Cancer genes and the pathways they control. *Nature Medicine* 10, 789–799 (2004)
- Williams, M., Williamson, J.: Combining argumentation and Bayesian nets for breast cancer prognosis. *Journal of Logic, Language and Information* 15, 155–178 (2006)
- Williamson, J.: Maximising entropy efficiently. *Electronic Transactions in Artificial Intelligence Journal*, 6 (2002), www.etaij.org
- Williamson, J.: *Bayesian nets and causality: philosophical and computational foundations*. Oxford University Press, Oxford (2005a)
- Williamson, J.: Objective Bayesian nets. In: Artemov, S., Barringer, H., d’Avila Garcez, A.S., Lamb, L.C., Woods, J. (eds.) *We Will Show Them! Essays in Honour of Dov Gabbay*, vol. 2, pp. 713–730. College Publications, London (2005b)
- Williamson, J.: Causality. In: Gabbay, D., Guenther, F. (eds.) *Handbook of Philosophical Logic*, vol. 14, pp. 89–120. Springer, Heidelberg (2007a)
- Williamson, J.: Motivating objective Bayesianism: from empirical constraints to objective probabilities. In: Harper, W.L., Wheeler, G.R. (eds.) *Probability and Inference: Essays in Honour of Henry E. Kyburg Jr.*, pp. 151–179. College Publications, London (2007b)
- Williamson, J., Gabbay, D.: Recursive causality in Bayesian networks and self-fibring networks. In: Gillies, D. (ed.) *Laws and models in the sciences*, pp. 173–221. With comments, pp. 223–245. King’s College Publications, London (2005)
- Xia, Y., Yu, H., Jansen, R., Seringhaus, M., Baxter, S., Greenbaum, D., Zhao, H., Gerstein, M.: Analyzing cellular biochemistry in terms of molecular networks. *Annu. Rev. Biochem.* 73, 1051–1087 (2004)

Modeling the Temporal Trend of the Daily Severity of an Outbreak Using Bayesian Networks

Xia Jiang, Michael M. Wagner, and Gregory F. Cooper

Department of Biomedical Informatics, University of Pittsburgh, Pittsburgh, PA
xjiang@cbmi.pitt.edu, mmw1@pitt.edu, gfc@cbmi.pitt.edu

Summary. A disease outbreak is an epidemic limited to localized increase, e.g., in a village, town, or institution. An epidemic curve is a graphical depiction of the number of outbreak cases by date of onset of illness. If we could estimate the epidemic curve early in an outbreak, this estimate could guide the investigation of other outbreak characteristics. Furthermore, a good estimate of the epidemic curves tells us how soon the outbreak will reach a given level of severity if it goes uncontrolled. Previously, methods for doing real-time estimation and prediction of the severity of an outbreak were very limited. As far as predicting future cases, ordinarily epidemiologists simply made an educated guess as to how many people might become affected. We develop a Bayesian network model for real-time estimation of an epidemic curve, and we show results of experiments testing its accuracy.

7.1 Introduction

Le Strat and Carrat [1999] define an **epidemic** as “the occurrence of a number of cases of a disease, in a given period of time in a given population, that exceed the expected number.” Last [2000] defines an **outbreak** as “an epidemic limited to localized increase, e.g., in a village, town, or institution.” An **epidemic curve** is a graphical depiction of the number of outbreak cases by date of onset of illness [Wagner et al., 2006]. An epidemic curve is one of the most important characteristics of an outbreak. If we could estimate the epidemic curve early in an outbreak, this estimate could guide the investigation of other outbreak characteristics such as the disease-causing biological agent, source, and route of transmission. Furthermore, a good estimate of the epidemic curves tells us how soon the outbreak will reach a given level of severity if it goes uncontrolled. This information is crucial to public health officials when they are making decisions concerning preparatory measures. That is, once the epidemic curve estimate informs them as to when the epidemic may reach a given level of severity, they become aware of the immediacy with which they must obtain sufficient resources and supplies to handle disease treatment.

As an example, consider the epidemic curve in Figure 7.1. This curve was constructed (after the outbreak was over) from clinically defined and laboratory-confirmed cases of a foodborne *Cryptosporidium* outbreak that occurred on a

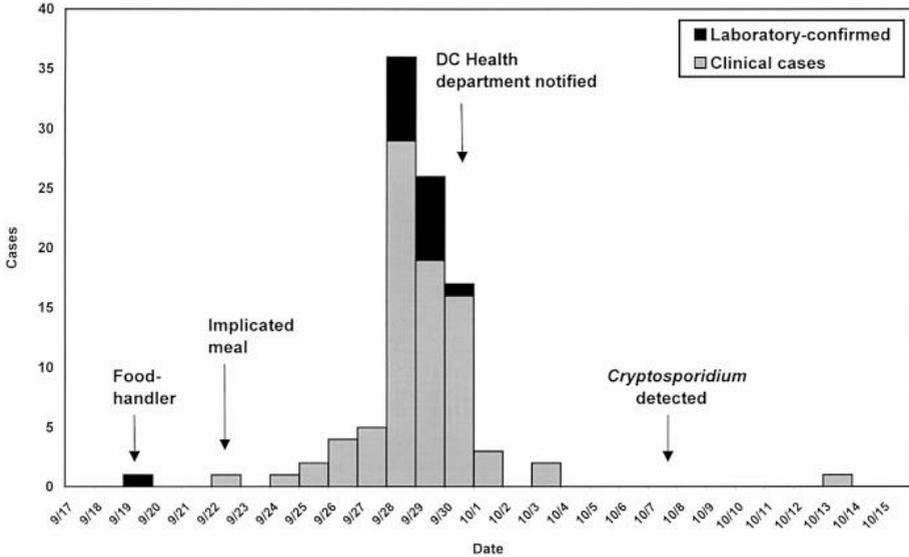
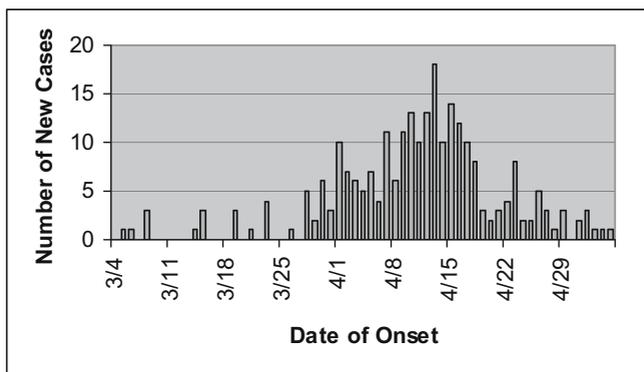


Fig. 7.1. An Epidemic Curve for the Washington D.C. *Cryptosporidium* outbreak

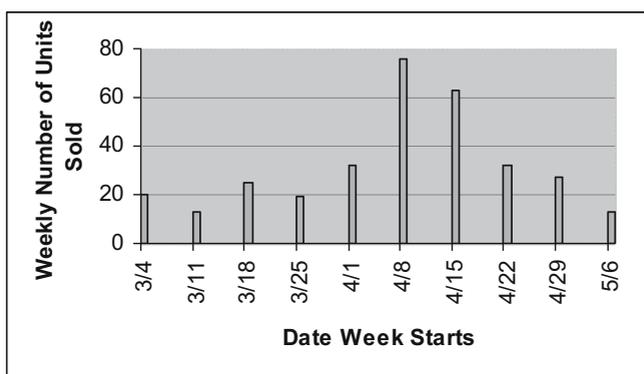
Washington, DC university campus. The curve indicates a possible food contamination through a tight clustering of cases in three days. If health care officials had been able to estimate this curve early during the outbreak, they could possibly have determined the biological agent and the route of transmission early enough so that appropriate control measures could have been taken to prevent additional cases.

The epidemic curve for an outbreak is often correlated with the daily counts of some observable event. For example, Figure 7.2 (a) shows an epidemic curve constructed from a sample of the population affected by a *Cryptosporidium* outbreak in North Battleford, Saskatchewan in spring, 2001. The outbreak was caused by a contamination of public drinking water. *Cryptosporidium* infection causes diarrhea. Figure 7.2 (b) shows the weekly over-the-counter (OTC) sales of antidiarrheal drugs at one pharmacy in North Battleford during the time period affected by the outbreak. The correlation between these two curves suggests that by monitoring OTC sales of such drugs we can possibly detect a *Cryptosporidium* outbreak at an early stage, and perhaps even estimate the epidemic curve.

Previously, methods for doing real-time estimation and prediction of the severity of an outbreak were very limited. For the most part, investigators simply did their best to intensify surveillance in an effort to identify all cases so that the observed number of cases was as close to the real number of cases as possible [Wagner et al., 2006]. However, as far as predicting future cases, ordinarily epidemiologists simply made an educated guess as to how many people might become affected. Recently, some strides have been made in characterizing outbreaks and predicting their severity. PANDA [Cooper et al., 2004] and BARD



(a)



(b)

Fig. 7.2. An epidemic curve for a *Cryptosporidium* outbreak in North Battleford, Saskatchewan is in (a), while weekly OTC sales of antidiarrheal drugs at one pharmacy in North Battleford is in (b)

[Hogan et al., 2007], which are two state of the art outbreak detection algorithms, can provide estimates of some outbreak characteristics such as source and/or route of transmission of the outbreak. However, neither of them predicts the future characteristics of the outbreak. Jiang and Wallstrom [2006] describe a Bayesian network model for outbreak detection and estimation. That model not only detects an outbreak but also estimates important characteristics of the outbreak. Namely, it estimates the outbreak's size and duration, and how far into the outbreak we are. They note that knowledge of the probable values of these variables can help guide us to a decision that maximizes expected utility. A shortcoming of their model is that it only estimates the size and duration of the outbreak; it does not estimate the epidemic curve. However, as discussed

above, an estimate of the epidemic curve itself would be more valuable to public health care officials.

Next we develop a Bayesian network model that estimates an epidemic curve from the daily counts of some observable event. After presenting the model, we show the results of evaluating its performance using four actual influenza outbreaks.

7.2 A Model for Epidemic Curve Estimation

The Bayesian network model presented here is applicable to estimating the epidemic curve for any type of outbreak from the daily¹ counts of some observable event, whose daily count is positively correlated with the daily count of outbreak cases. For example, it could be used to estimate the epidemic curve for an outbreak of influenza from the daily counts of patients presenting in the ED with respiratory symptoms; in addition, it could be used to estimate the epidemic curve for a *Cryptosporidium* outbreak from the daily counts of units of OTC antidiarrheal drugs sold. We will describe the model in general terms, without referring to any particular type of outbreak. First, we present the structure of the Bayesian network; then we discuss the parameters. See [Neapolitan, 2004] for an introduction to Bayesian networks and dynamic Bayesian networks, which are networks specifically designed to model times series.

7.2.1 Network Structure

The general Bayesian network structure for the model is shown in Figure 7.3. It is assumed that we already know (or at least suspect) an outbreak is ongoing, for example, by way of clinical case findings or epidemiology surveillance. So the system does not try to detect whether there is an outbreak. Rather based on daily counts (The variables labeled $C[i]$) of the observable event up until today, the system estimates the epidemic curve values (The variables labeled $E[i]$ where $i \leq 0$) on today and days preceding today, and predicts the epidemic curve values (The variables labeled $E[i]$ where $i > 0$) on days following today. The set of values of all the variables labeled $E[i]$ constitute the epidemic curve. Our estimate of them constitutes our **estimate of the epidemic curve**. The darkly-shaded nodes in the networks are constants, while the lightly-shaded nodes are the ones that are instantiated when we use the network to make inference. Next we describe the nature of each node. First, we describe the nodes that are constants.

1. N : This constant is the number of people in the jurisdiction being monitored.
2. A : This constant is the probability that an individual, who is sick with a disease that could produce an occurrence of the observable event, actually does produce an occurrence. For example, if the observable event is the sale of one thermometer, it is the probability that an individual with a temperature buys a thermometer. We call this probability the **action tendency**.

¹ Although the unit of time is usually one day, it need not be. For example, we could use weekly counts.

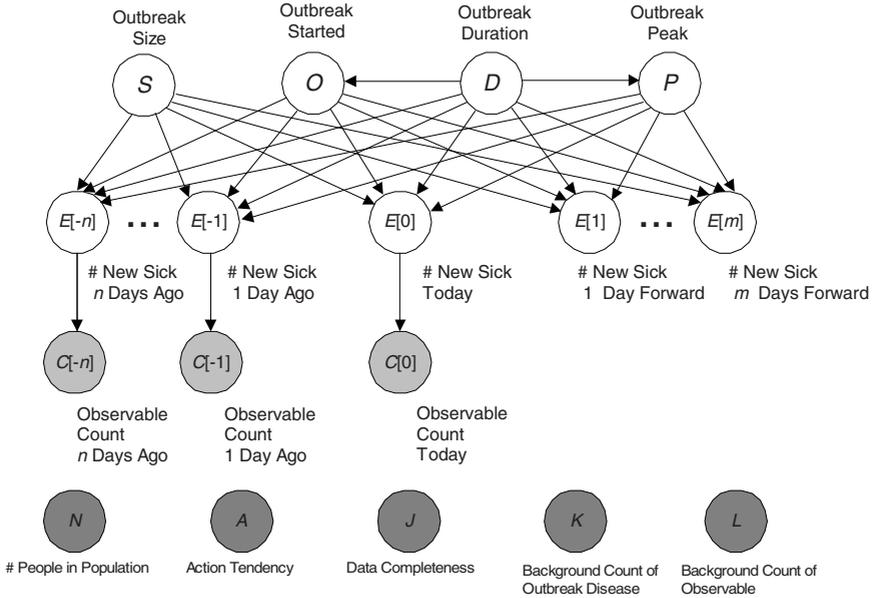


Fig. 7.3. Bayesian network for estimating an epidemic curve

3. J : This constant is the data completeness. It is the fraction of all occurrences of the observable event that, on the average, occur in the entities being monitored. For example, if the observable event is the sale of one thermometer, it is the fraction of all thermometers sales (in the monitored jurisdiction) that occur in the stores whose thermometer sales are included in our count.
4. K : This constant is the average daily count of occurrences of the outbreak disease when no outbreak is occurring. It is called the background count. For example, if the outbreak disease is influenza, it is the average daily count of sporadic influenza cases in non-flu season.
5. L : This constant is the average daily count of occurrences of the observable event that are not due to the outbreak disease. It is assumed to be the same regardless of whether an outbreak is taking place. For example, if the outbreak disease is influenza and the observable event is the sale of one thermometer, it is the average daily count of thermometer sales that are not due to influenza.

Next we discuss the nodes that are variables.

1. S : This is the size of the outbreak. The size is the percent of the population that eventually becomes ill due to the outbreak. Note that this is the percent if no measures are taken to control the outbreak. We hope to make the size smaller by taking appropriate measures when the potential epidemic curve is estimated. Its domain includes the integers between *minsize* and *maxsize*,

where *minsize* and *maxsize* are the minimum and maximum sizes of an outbreak of this type.

2. *D*: The variable *D* represents the duration of an outbreak that started some time in the previous *durmax* days. Its domain includes the integers between *durmin* and *durmax* inclusive, where *durmin* and *durmax* are the minimum and maximum durations of an outbreak of this type.
3. *O*: This variable represents the number of days ago an ongoing outbreak started. Its domain includes the integers between 1 and *durmax* inclusive. It depends on *D* because the outbreak could not have started longer ago than the duration of the outbreak.
4. *P*: This variable represents the day the outbreak reaches its peak. Its domain includes the integers between *peakmin* and *peakmax* inclusive, where *peakmin* and *peakmax* are the first and last days on which an outbreak of this type could reach its peak. It depends on *D* because the day on which the peak is reached must come before the last day of the outbreak.
5. $E[i]$: This variable is the number of individuals becoming sick with the monitored (outbreak) disease on day *i*. For example, if the disease being monitored is influenza, it is the number of individuals becoming sick with influenza on day *i*. $E[0]$ is the number today, $E[-i]$ is the number *i* days before today, and $E[i]$ is the number *i* days after today. By today we mean the current day on which we are trying to estimate the epidemic curve. The domain of these variables includes the integers between 0 and *maxsick*, where *maxsick* is the maximum number of people becoming sick on a given day in the jurisdiction being monitored. They depend on *S* and *D* because larger outbreaks result in more individuals becoming sick. They depend on *O* and *P* because, during an outbreak, the number of sick individuals increases, as the outbreak progresses, to the peak, and then decreases. In the next subsection, we illustrate how these dependencies can be modeled.
6. $C[i]$: This variable is the count of the observable event on day *i*. For example, if the observable event is the sale of one unit of some OTC drug, it would be the count of number of units sold. $OC[0]$ is the count today, and $OC[-i]$ is the count *i* days before today. By today we mean the current day on which we are trying to estimate the epidemic curve. We can look back as many days as deemed appropriate. Their domain includes the integers between 0 and *maxotc*, where *maxotc* is the maximum count of the observable event in the jurisdiction being monitored. $C[i]$ depends on $E[i]$ because when more individuals are sick the count is higher. In the next subsection, we illustrate how this dependency can be modeled.

7.2.2 Network Parameters

Although the parameters in different applications will be similar, it is difficult to describe them in the abstract. Therefore, we will describe the parameters and their values specific to the influenza outbreak detection system that is evaluated in the next section. Some of the parameter values were obtained from domain knowledge of influenza outbreaks. We will not repeatedly state this when it is

the case. First, we give the values of the constants and show how they were obtained.

1. The population size N was obtained from census data for each jurisdiction.
2. The data completeness J was obtained from data for each jurisdiction.
3. The background influenza count K was obtained from data for each jurisdiction.
4. For a particular jurisdiction, let

T_1 = Average daily count of thermometers sold in influenza season
 T_2 = Average daily count of thermometers sold in non-flu season
 F = Average daily number of influenza cases in influenza season
 K = Average daily number of influenza cases in non-flu season
 L = Average daily number of thermometer sold not due to influenza
 A = Probability of buying a thermometer if one has influenza
 J = Data completeness for this jurisdiction.

Then

$$T_1 = J(A \times F + L) \quad (7.1)$$

and

$$T_2 = J(A \times K + L). \quad (7.2)$$

We obtained the values of T_1, T_2, F, K , and J from data for a given jurisdiction. We then simultaneously solved Equations 7.1 and 7.2 to obtain the values of A and L .

Next we discuss the parameters. We assumed little prior knowledge concerning the probability distributions of S , D , O , and P . Our purpose was to see how much could be learned without imposing too much of our prior belief concerning influenza outbreaks. Specifically, we assumed the following probability distributions:

1. S is uniformly distributed over all integers between 2% and 11%.
2. D is uniformly distributed over all integers between 8 and 80.
3. For a given value of D , O is uniformly distributed over all integers between 1 and D .
4. For a given value of D , it is assumed the peak occurs in the first $D/2$ days, and is more likely to be near $D/2$ than 0. Specifically, P has the $beta(P; 3, 2)$ density function discretized and adjusted to the interval $[0, D/2]$. We chose the beta density function because its parameters allow us to readily model our prior belief as to where the peak might be. The least presumptive assumption would be to use the $beta(P; 1, 1)$ density function, which is the uniform density function. However, this assumption is completely unrealistic since the peak of the outbreak would not occur on the first day.

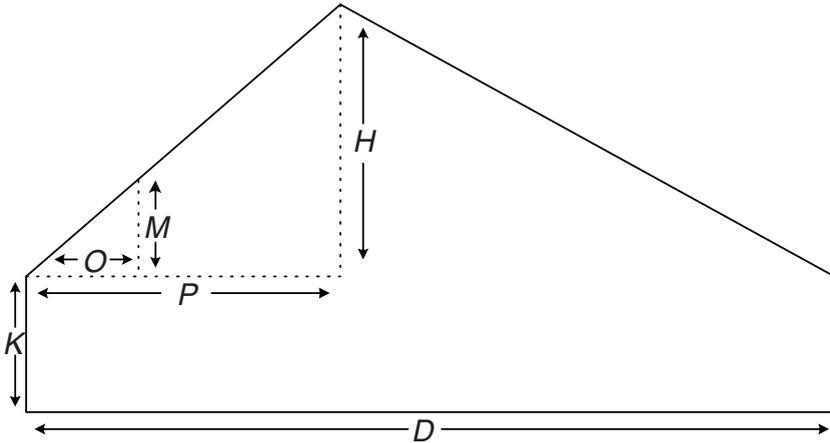


Fig. 7.4. The area of this region is about equal to the total number of influenza cases during the outbreak

5. When an outbreak is ongoing, all cases of the outbreak disease are included in the epidemic curve. This is illustrated in Figure 7.4. In that figure, M represents the number of new cases due to the outbreak today (the day we are making our estimate), and H represents the number of new cases due to the outbreak on the day the outbreak reaches its peak. The remaining variables in the figure are the ones we previously defined. Given values of K , P , and D , the total number of influenza cases during the outbreak is given approximately by the area of the region in Figure 7.4. We say approximately, because the total number cases is the sum of discrete values on each day. The area of the region in Figure 7.4 approximates the sum of the columns in the bar graph that represents these values (See e.g. Figure 7.1). We therefore have that

$$\frac{S \times N}{100} = KD + \frac{D \times H}{2}.$$

Furthermore, for $O \leq P$

$$\frac{M}{O} = \frac{H}{P}.$$

Solving for M , we have

$$M = \frac{O}{P} \left(\frac{S \times N}{50D} - 2K \right)$$

Finally, the height $E[0]$ of the figure at point O , which is the number of outbreak cases on day O of the outbreak, is given by

$$\begin{aligned} E[0] &= K + M \\ &= K + \frac{O}{P} \left(\frac{S \times N}{50D} - 2K \right). \end{aligned}$$

Similarly, if $O \geq P$

$$E[0] = K + \left(\frac{D - O}{D - P} \right) \left(\frac{S \times N}{50D} - 2K \right).$$

This would be the value of $E[0]$ if the number of daily cases was deterministically determined by the other variables. We inserted a random component by making $E[0]$ a random variable that has the Poisson distribution with mean E . Similarly, $E[i]$ for $i \neq 0$ is assumed to have the Poisson distribution with mean given by the previous equalities except O is replaced by $O + i$.

6. We have that the expected value μ_i of $C[i]$ is given by

$$\mu_i = J(A \times E[i] + L). \quad (7.3)$$

We made the assumption that the day on which an individual becomes sick with the outbreak disease is the same as the day on which the individual takes action A . Although this would not absolutely be the case, the assumption should be a good first order approximation as some individuals sick from previous days should take the action on day i , while some individuals sick on day i should take the action on future days. So the discrepancies should approximately offset. We made $C[i]$ a random variable that has the Poisson distribution with mean given by Equation 7.3. In the influenza outbreak detection system, $C[i]$ is the count of daily thermometer sales.

When the network is used to estimate an epidemic curve, the variables $C[i]$ for all i are instantiated to the OTC counts of thermometer sales for today and all days preceding today (for as many days as we choose to look back). Inference is then done to determine the conditional probability distributions of the variables $E[i]$ for all i . The expected values of these variables constitute our estimate of the epidemic curve. We used the Bayesian network package Netica (<http://www.norsys.com/>) to develop the network and perform the inference. Note that for a given i , the conditional probability distribution of $E[i]$ depends not only on the instantiated value of $C[i]$ but on the instantiated values of $C[j]$ for all $j \neq i$. This is because $C[i]$ does not d-separate $E[i]$ from $C[j]$. So the inference is much more complex than simply doing a calculation using Equality 7.3.

7.3 Experiments

We evaluated the model by determining how well it estimates the epidemic curves of actual influenza outbreaks.

7.3.1 Method

The data necessary to obtain the parameter values for influenza outbreak detection systems (as discussed in the previous section) were available for four

jurisdictions in the United States. Furthermore, all four jurisdictions had significant influenza outbreaks starting in fall, 2003 and lasting 66-68 days. For the sake of maintaining anonymity, we labeled the jurisdictions A, B, C, and D. We obtained the data necessary to the outbreak detection systems and information concerning the outbreaks from the Centers for Disease Control and Prevention (<http://www.cdc.gov>) and the National Retail Data Monitor system managed by RODS laboratory (<http://rods.health.pitt.edu>). The outbreak information did not include epidemic curves themselves. However, using the durations of the outbreaks, the influenza-like illness (ILI) curves, counts of deaths due to influenza during the outbreaks (which were also available), and national figures concerning influenza and influenza deaths, we were able to estimate weekly epidemic curves for these outbreaks. We evaluated our model using these estimates as gold standards. However, they too are really only estimates taken from all known data concerning the outbreaks, and are not true gold standards.

Table 7.1. Values of the constants for each jurisdiction

Jurisdiction	N	A	J	K	L
A	10,047,239	.143	.24	322	114
B	698,000	.161	.57	65	9.5
C	3,001,146	.175	.29	97	25
D	1,261,303	.1	.5	130	25

Table 7.1 shows the values of the constants for each of the four jurisdictions.

7.3.2 Results

First, we show the results of our study using the uninformative probability distributions of S , D , O , and P discussed above. Then we investigate the sensitivity of the results to these distributions.

Results Using Uninformative Probability Distributions

Figures 7.5, 7.6, 7.7, and 7.8 show the gold standard epidemic curves for the four jurisdictions along with the system's epidemic curve estimations on the 20th day, 25th day, and 30th day of the outbreaks. The weekly totals of the system's posterior expected values of the number of influenza cases were used as the estimates. The outbreaks lasted between 66 and 68 days.

If we let n be the number of weeks the outbreak took place, x_i be the sequence of weekly values in the gold standard epidemic curve, and y_i be the sequence of weekly values in the estimated epidemic curve, we set

$$\text{Error} = 100\% \times \frac{\sum_{i=1}^n |y_i - x_i|}{\sum_{i=1}^n x_i}.$$

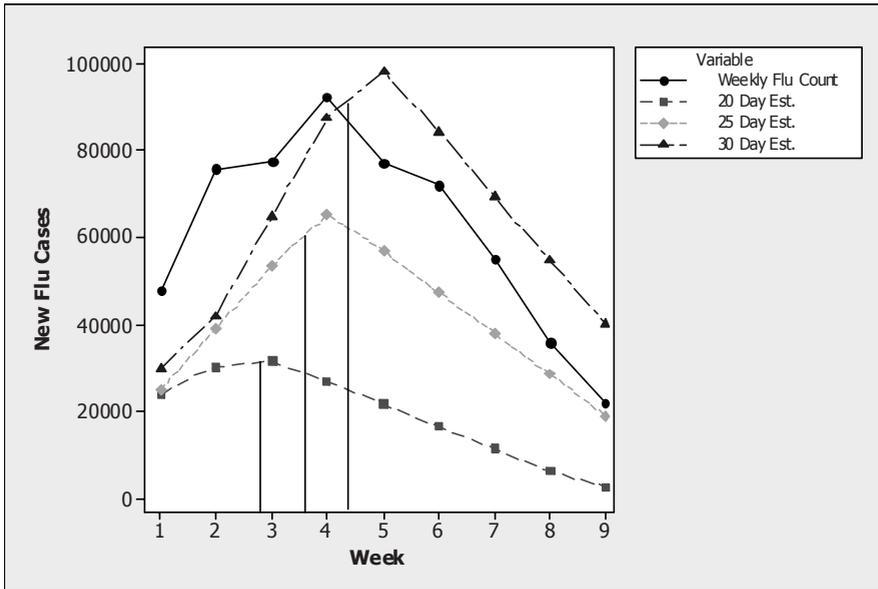


Fig. 7.5. Epidemic curve estimates for jurisdiction A

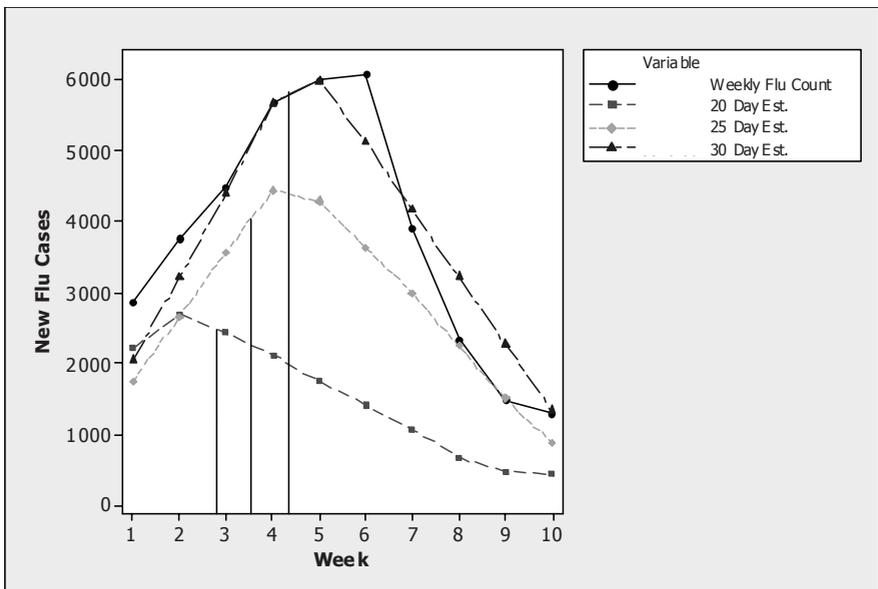


Fig. 7.6. Epidemic curve estimates for jurisdiction B

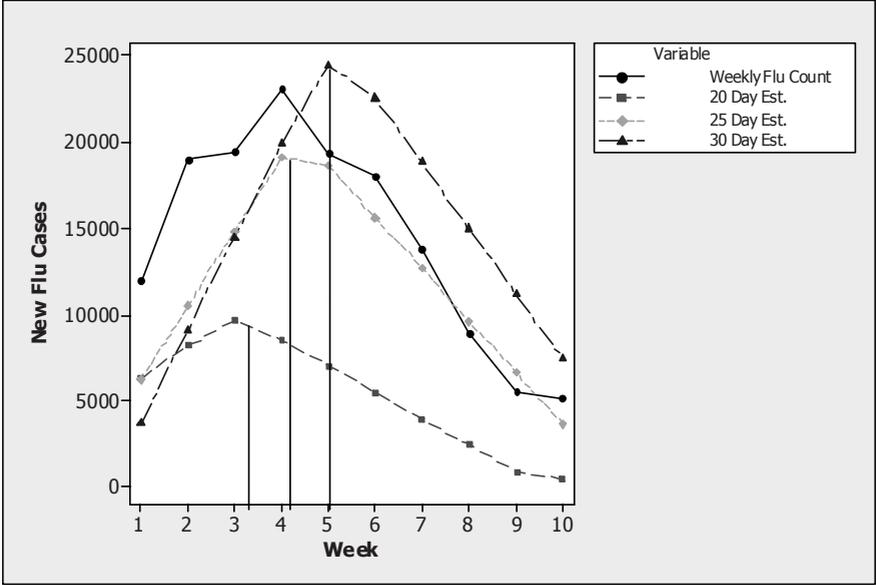


Fig. 7.7. Epidemic curve estimates for jurisdiction C

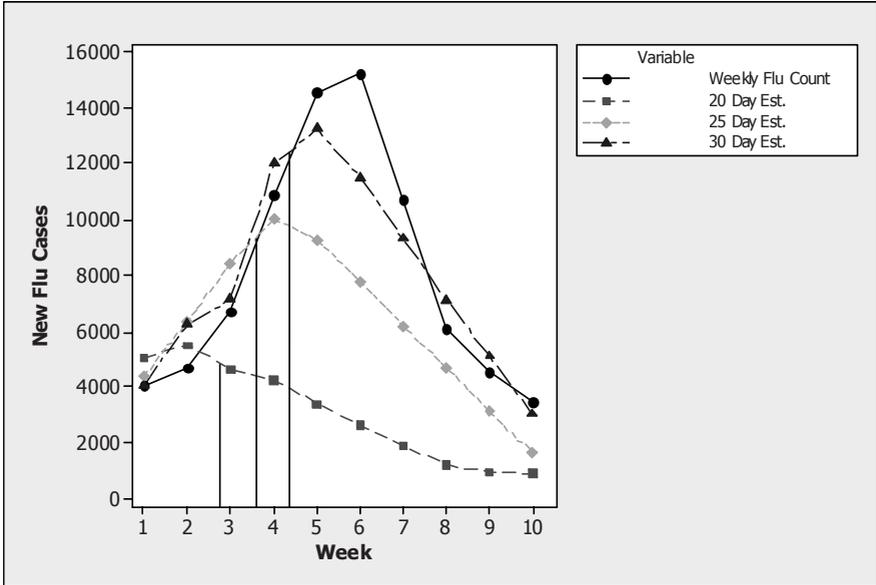


Fig. 7.8. Epidemic curve estimates for jurisdiction D

Table 7.2. Errors in estimated epidemic curves

Jurisdiction	20 Day Error	25 Day Error	30 Day Error
A	69.1%	32.9%	27.6%
B	59.6%	26.4%	11.8%
C	63.2%	20.9%	37.8%
D	66.9%	32.7%	15.1%
Average	64.7%	28.2%	23.1%

If the sequences were the same, this error would be 0. The values of the error on the 20th day, 25th day, and 30th day are shown in Table 7.2. As expected, the average error decreases as we proceed into the outbreak. Notice, however, that in the case of jurisdiction C, the error is higher on the 30th day than on the 25 day. Looking at Figure 7.7, we see that the epidemic curve estimate on the 30th day looks much like the gold standard, but is shifted to the right. Given the way we computed the error, this accounts for the large error. If we were only concerned with the error in the total number of influenza cases, the error would be 1.8%. In the same way, for jurisdiction A the 30-day estimate is shifted to the right, but not so much as to make its error greater than that of the 25-day estimate. These shifts may be occurring because individuals might often buy thermometers on days following the onset of illness.

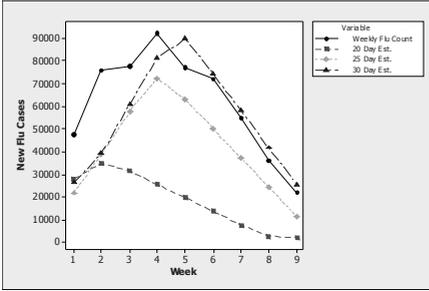
Results Using Informative Probability Distributions

Other than formulating a weak belief concerning when the outbreak reaches its peak, in the studies just discussed we assumed uninformative probability distributions of the size, duration, peak, and number of days since the outbreak started. Our purpose was to investigate the quality of the estimates obtained when we do not make additional assumptions about the nature of the outbreaks. Next we analyze the sensitivity of our results to these probability distributions.

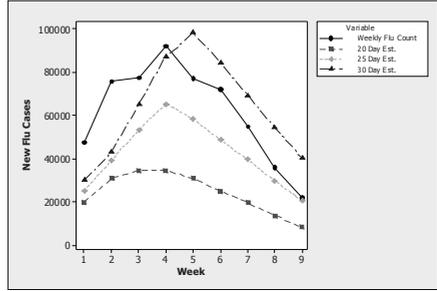
We repeated the study for jurisdiction A four times, each time using one of the following informative probability distributions:

1. The duration has the $beta(D; 10, 10)$ density function adjusted to the interval $[1, 80]$.
2. The size has the $beta(S; 10, 10)$ density function adjusted to the interval $[1, 11]$.
3. The peak has the $beta(P; 12, 8)$ density function adjusted to the interval $[1, D/2]$.
4. The number of days since the outbreak started has the $beta(O; 10, 10)$ density function adjusted to the interval $[1, D/2]$.

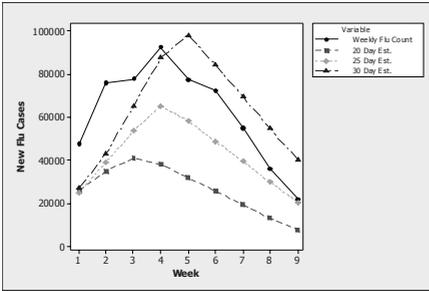
Our purpose was to model what we consider reasonable prior beliefs about the domain. For example, we believed that the duration was more likely to be in the mid-range than at either end point; however, we did not have a strong belief as



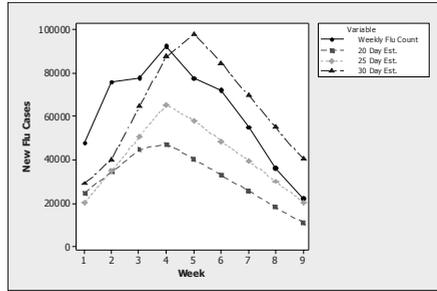
(a) Duration has an informative probability distribution.



(b) Size has an informative probability distribution.



(c) Peak has an informative probability distribution.

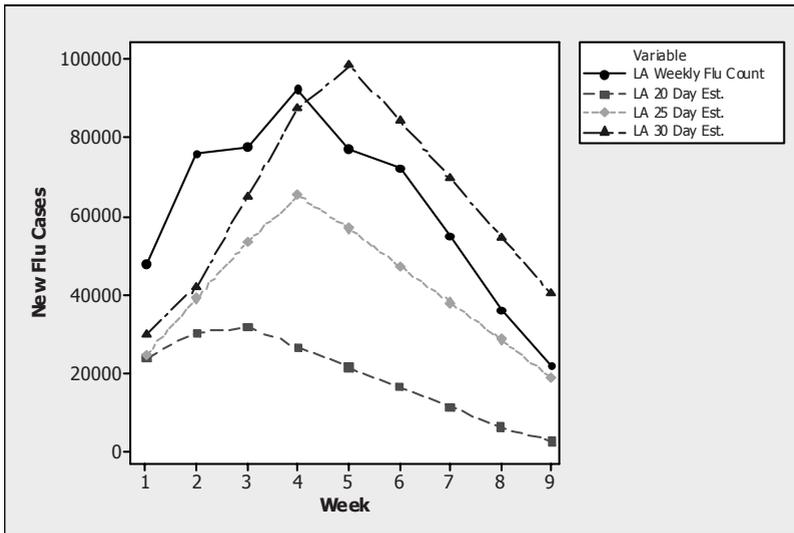


(d) # days has an informative probability distribution.

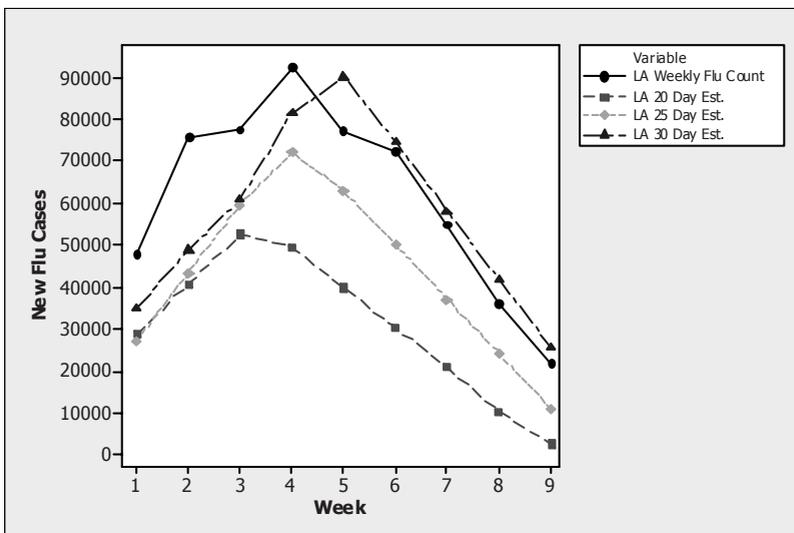
Fig. 7.9. Epidemic curve estimates for the influenza outbreak in jurisdiction A using informative probability distributions

to where it was actually located. So we used the $beta(D; 10, 10)$ density function. Similarly, we believed that the size was more likely to be in the mid-range. As to the peak, we simply made our belief stronger that it was located about $2/3$ of the way into the first half of the outbreak (Recall that previously we used the $beta(P; 3, 2)$ density function). As far as the number of days since the outbreak started, we assumed that we were investigating the outbreak during the first half of the outbreak. We also assumed that we were more likely to be in the mid-range of the first half of the outbreak. However, we obtained similar results when we assumed a uniform distribution over the first half. Note that, of the four informative probability distributions, the only ones that favored the outbreaks we were investigating were the ones concerning size and the peak. That is, the size and the peak of the four outbreaks were all at about the points that we considered most probable. On the other hand, the durations of the outbreaks were 66-68 days, which is not near the midpoint. Furthermore, the assumption concerning the number of days since the outbreak started has little to do with a particular outbreak.

Figure 7.9 shows the results. Interestingly, the informative probability distributions that improved the accuracy most were the ones that did not favor the outbreaks we were investigating. That is, using an informative probability



(a) No variable has an informative probability distribution.



(b) All variables have informative probability distributions.

Fig. 7.10. Epidemic curve estimates for the influenza outbreak in jurisdiction A using uninformative probability distributions are in (a), while estimates using informative probability distributions of all four variables are in (b)

Table 7.3. Errors obtained using informative probability distributions are on the left, while those obtained using uninformative probability distributions are on the right

Jurisdiction	20 Day Error		25 Day Error		30 Day Error	
A	50.3%	69.1%	30.1%	32.9%	17.2%	27.6%
B	50.4%	59.6%	27.3%	26.4%	11.9%	11.8%
C	41.6%	63.2%	18.6%	20.9%	22.3%	37.8%
D	57.3%	66.9%	33.1%	32.7%	15.8%	15.1%
Average	49.9%	64.7%	27.3%	28.2%	16.8%	23.1%

distribution for the duration improved the estimates at both the 25th and 30th days the most, and using an informative probability distribution for the number of days since the outbreak started improved the estimate at the 20th day the most.

We repeated the study for jurisdiction A with all four variables having the informative probability distributions listed above. Figure 7.10 (b) shows the results, while Figure 7.10 (a) shows the estimates when we do not use any of the informative distributions (These are the same estimates as those shown in Figure 7.5). Notice that the estimates in Figure 7.10 (b) combine the improvements that we see in Figures 7.9 (a) and (d).

Finally, we repeated the study for the other three jurisdictions with all four variables having informative probability distributions. Table 7.3 compares the errors obtained using informative and uninformative probability distributions. On the average, we obtain significantly better estimates using informative probability distributions on the 20th and 30th day of the outbreak, but not on the 25th day. It seems prior information would be more useful when we are earlier in the outbreak and do not have as much information concerning the specific outbreak. So it may simply be an anomaly that we obtain greater improvement on the 30th day than on the 25th day. Regardless, in general prior information does seem to improve our results.

7.4 Discussion

Our study indicates that a system, which uses uninformative probability distributions of size, duration, peak, and number of days since the outbreak started, may be capable of doing a good job of estimating an influenza epidemic curve as early as the 25th day of 66-68 day outbreaks, but it cannot closely estimate the curve on the 20th day. We repeated the study using informative probability distributions of these variables and obtained significantly better results, with even the 20-day estimate having an average error under 50%. These results indicate that modeling our probability distributions of these variables accurately can have a major impact on a system that estimates epidemic curves. Additional investigation is needed to determine how to best model these distributions. For the past several years, data on observable events related to influenza outbreaks

have been collected by the RODS Laboratory at the University of Pittsburgh. Once more data is collected, we can learn better as to how to represent our prior beliefs.

The system we evaluated assumed that everyone who becomes sick on a given day buys thermometers on that day (if they buy at all). In reality, there may be some lag in the number of days in which a sick individual buys a thermometer. A possible improvement would be to investigate what this lag is, and incorporate it into the system. Furthermore, the system could be improved by using a multivariate times series (several observable events) rather than only a univariate time series (a single observable event). Our model can readily handle multivariate times series. We merely need to include a set of count nodes for each observable event. By incorporating all these modifications, we may be able to obtain better estimates earlier.

Acknowledgements. This research was supported by the National Science Foundation Grant No. 0325581.

References

1. Cooper, G.F., Dash, D.H., Levander, J.D., Wong, W.K., Hogan, W.R., Wagner, M.M.: Bayesian Biosurveillance of Disease Outbreaks. In: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, Banff Park Lodge, Banff, Canada, July 7-11 (2004)
2. Hogan, W.R., Cooper, G.F., Wallstrom, G.L., Wagner, M.M., Dipinay, J.M.: The Bayesian aerosol release detector: An algorithm for detecting and characterizing outbreaks caused by an atmospheric release of *Bacillus anthracis*. *Statistics in Medicine* (October 22, 2007)
3. Jiang, X., Wagner, M.M., Wallstrom, G.L.: A Bayesian Network for Outbreak Detection and Prediction. In: Proceedings of AAAI 2006, Boston, MA (2006)
4. Last, J.M.: *A Dictionary of Epidemiology*. Oxford University Press, New York (2000)
5. Le Strat, Y., Carrat, F.: Monitoring Epidemiological Surveillance Data using Hidden Markov Models. *Statistics in Medicine* 18 (1999)
6. Neapolitan, R.E.: *Learning Bayesian Networks*. Prentice Hall, Upper Saddle River (2004)
7. Wagner, M.M., Gresham, L.S., Dato, V.: Case Detection, Outbreak Detection, and Outbreak Characterization. In: Wagner, M.M. (ed.) *Handbook of Biosurveillance*. Elsevier, NY (2006)

An Information-Geometric Approach to Learning Bayesian Network Topologies from Data

Eitel J.M. Lauría

School of Computer Science and Mathematics
 Marist College
 3399 North Road
 Poughkeepsie, NY 12601, USA
 Eitel.Lauria@marist.edu

Abstract. This work provides a general overview of structure learning of Bayesian networks (BNs), and goes on to explore the feasibility of applying an information-geometric approach to the task of learning the topology of a BN from data. An information-geometric scoring function based on the Minimum Description Length Principle is described. The info-geometric score takes into account the effects of complexity due to both the number of parameters in the BN, and the geometry of the statistical manifold on which the parametric family of probability distributions of the BN is mapped. The paper provides an introduction to information geometry, and lays out a theoretical framework supported by empirical evidence that shows that this info-geometric scoring function is at least as efficient as applying BIC (Bayesian information criterion); and that, for certain BN topologies, it can drastically increase the accuracy in the selection of the best possible BN.

8.1 Introduction

A Bayesian network (BN) with topology \mathcal{G} (represented by a directed acyclic graph, or DAG) and vector parameter $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_n\}$, encodes the joint probability distribution of a set of random variables $\mathbf{X} = \{X_1, X_2, \dots, X_n\}$. Recalling the product probability rule, the joint probability distribution of X_1, X_2, \dots, X_n can be written as:

$$p(\mathbf{x} | \boldsymbol{\theta}, \mathcal{G}) = \prod_{i=1}^n p(x_i | x_{pa(i)}, \boldsymbol{\theta}_i, \mathcal{G}) = \prod_{i=1}^n \theta_{ik}(j) \quad (8.1)$$

The expression $p(\mathbf{x} | \boldsymbol{\theta}, \mathcal{G})$ denotes the probability of a conjunction of particular assignments x_1, x_2, \dots, x_n to the set of variables $X_i \in \mathbf{X}$, and $x_{pa(i)}$ identifies a given configuration of the list of direct parents of X_i , linked to X_i through the arcs in DAG \mathcal{G} . For a discrete BN, the conditional probability tables at each node $X_i \in \mathbf{X}$ are represented by the parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_n\} = \left\{ \left[\theta_{ik}(j) \right]_{k=1}^{r_i} \right\}_{j=1}^{q_i}$, where $i=1..n$

identifies each variable $X_i \in \mathbf{X}$; $k = 1..r_i$ identifies each of the r_i states of variable $X_i \in \mathbf{X}$; $j = 1..q_i$ identifies the set of q_i valid configurations of values of the parent variables of $X_i \in \mathbf{X}$ ($x_{pa(i)}$).

8.2 Learning the Topology of Bayesian Networks from Data

The task of learning a BN's topology from data can be seen as a model selection problem. Standard scientific practice, summarized by the inductive bias known as Occam's razor, prescribes that it is usually preferable to choose the least complex network that equally fits the data. Given a training data set D of N cases $D = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}$ where each $\mathbf{x}^{(k)}$ is a row vector representing a conjunction of observed states $x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}$ of the vector variable $\mathbf{X} = \{X_1, X_2, \dots, X_n\}$, the goal is to find a BN $\langle \mathcal{G}; \boldsymbol{\theta} \rangle$ with topology \mathcal{G} and vector parameter $\boldsymbol{\theta}$ that best describe the joint probability distribution over the training data D . The literature portrays two general approaches applied to the task of learning the topology of a Bayesian network from data: constraint-based methods and score-based methods. In this work we focus on score-based methods, as they are among the most common and powerful methods currently in use.

8.2.1 Score-Based Methods

These methods pose learning structure as an optimization problem, where the goal is to find the BN with DAG \mathcal{G} in the space of network topologies that best matches the training data set D . In general, a search algorithm uses a scoring metric to evaluate each BN with respect to D , looking for the BN topology that optimizes this score. The optimal BN with DAG \mathcal{G}^* is then used as the model for the topology of the domain¹. An alternative outcome is a sample of the models found, which approximates a Bayesian posterior probability.

Two elements need to be specified when considering this approach: the search procedure and the scoring metric.

Clearly, exhaustive search procedures are not feasible for most problems, as it is not possible to enumerate all the possible BNs for even a small number of nodes. The number of DAGs is super-exponential in the number of nodes: for just 10 variables there are $4.2 \cdot 10^{18}$ possible networks. Hence, exhaustive search has to be replaced by heuristic search techniques.

A local search algorithm that changes one single arc at each step can efficiently assess the gains made by adding, removing, or reversing the arc. A *greedy hill-climbing algorithm*, for example, performs at each step the local change that maximizes the gain, until it reaches a local maximum. The algorithm takes a starting network (empty, full, chosen at random, or elicited from previous knowledge) and iteratively applies single-arc addition, removal, or reversal, keeping the resulting network that

¹ Note that the topology of a BN can only be learnt up to its Markov equivalent class.

maximizes the score as the most current candidate. According to [1], although there is no guarantee that the procedure will find a global maximum, it does perform well in practice.

Learning the topology of a BN using greedy hill-climbing, simulated annealing and other variants is analyzed in [2]. Cooper and Herskovits' K2 algorithm [3] chooses an order over the nodes of the BN. Although the ordering reduces considerably the total number of topologies, the number is still too big to allow for an exhaustive search. K2 uses a greedy heuristic algorithm that, for each node X_i in the ordering, successively adds as a parent of X_i a variable from the list $(X_1, X_2, \dots, X_{i-1})$ that contributes the maximum increase in the score (i.e. the highest increase in probability of the resulting topology). This procedure is repeated independently for each node X_i until no node increases the score, or the number of parents of X_i exceeds a given threshold.

An alternative solution that performs in a quite efficient manner is to make use of *simulated annealing*. Simulated annealing is a Markov Chain Monte Carlo approach to optimization of multivariate functions. The term derives from the physical process of heating and slow cooling of a solid material (i.e. metal) in order to increase its toughness and reduce brittleness. During annealing, a melted solid in a heat bath, initially hot and therefore disordered, is slowly cooled so that the system remains in thermodynamic equilibrium at any given time, gradually moving towards an equilibrium state of minimal energy. In terms of statistical physics, the Boltzman distribution describes the probability distribution of a state \mathcal{G} with energy $E(\mathcal{G})$ for a system in a heat bath at temperature T :

$$\pi(\mathcal{G}) \propto \exp\left(-\frac{E(\mathcal{G})}{T}\right) \quad (8.2)$$

The original Metropolis algorithm [4] simulates the evolution of a thermodynamic system from one configuration to another. Given an initial state \mathcal{G}_0 of the system at energy $E(\mathcal{G}_0)$ and temperature T , the initial configuration is perturbed ($\mathcal{G}_0 \rightarrow \mathcal{G}^*$) keeping T constant, and the change in energy $\Delta E = E(\mathcal{G}^*) - E(\mathcal{G}_0)$ is calculated. If $\Delta E < 0$, the new configuration \mathcal{G}^* with lower energy is accepted. Otherwise, \mathcal{G}^* is accepted with probability given by the Boltzman distribution $\exp(-\Delta E/T)$.

The cooling stage of the annealing process is simulated by the Metropolis algorithm taking a sequence of slowly decreasing temperatures converging to 0. The Metropolis algorithm is run with each value of a sequence of decreasing temperatures T_0, T_1, \dots, T_N , resulting in a sequence of annealing states $\mathcal{G}_0, \mathcal{G}_1, \dots, \mathcal{G}_N$ with decreasing energies $E(\mathcal{G}_0), E(\mathcal{G}_1), \dots, E(\mathcal{G}_N)$. In the limit, when the temperature approaches 0, the system evolves towards an equilibrium state with minimum energy. This can be seen as a combinatorial optimization process where a sequence of feasible solutions gradually approach an optimal solution (global minimum) using the energy equation for the thermodynamic system as the objective function. It has been shown [5] that simulated annealing converges asymptotically to the optimal solution. The

temperature T and number of steps of each Metropolis run at each state \mathcal{G}_i control the optimization process.

The algorithm, as applied to BNs, is described in Figure 8.1. It starts with an initial topology \mathcal{G}_0 and is controlled by five parameters: T_0 , the initial temperature; α , the number of iterations within the Metropolis subroutine; β , the number of effective changes of structure within Metropolis; γ , the temperature decay factor ($0 < \gamma < 1$); and δ , the number of temperature decrements in the simulated annealing procedure (Figure 8.1). The energy function $E(\mathcal{G})$ is given by the scoring function, and the perturbation on the BN is any of three randomly eligible operations: arc addition, arc deletion or arc reversal.

<pre> Procedure Sim_Anneal ($G_0, T_0, \alpha, \beta, \gamma, \delta$) $i \leftarrow 0$ $T \leftarrow T_0$ $G \leftarrow G_0$ while ($i < \delta$) [G, k] \leftarrow Metropolis(G, T, α) if ($k = 0$) then exit else $i \leftarrow i + 1$ $T \leftarrow \gamma \times T$ endif endwhile return [G] </pre>	<pre> Function Metropolis (G, T, α) $j \leftarrow 0$ $k \leftarrow 0$ while ($j < \alpha$) and ($k < \beta$) $j \leftarrow j + 1$ $G_{new} \leftarrow$ Perturb(G) $\Delta E \leftarrow E(G_{new}) - E(G)$ if ($\Delta E < 0$) or ($e^{-\Delta E/T} > \text{Unif}(0,1)$) then $G \leftarrow G_{new}$ $k \leftarrow j + 1$ endif endwhile return [G, k] </pre>
---	---

Fig. 8.1. Simulated Annealing Algorithm

Approximate model averaging can be useful when the amount of sample data is not large. When the number of possible topologies is large, it is usually not possible to average over all topologies. For such cases a search can be made that identifies highly probable topologies and then an average can be computed over them. Madigan and York [6] proposed an approach based on Markov Chain Monte Carlo (MCMC) methods. Friedman and Koller [7] have suggested using MCMC, but over orderings rather than over network topologies, adducing that the space of orderings is much smaller and more regular than the space of network topologies.

8.2.2 Incomplete Data

If the BN's topology is not known and not all of the variables are observable in the training data set (hidden variables or missing data), the problem is particularly challenging. Friedman [8] introduced the Structural EM algorithm, combining the Expectation Maximization (EM) algorithm [9], used to estimate network parameters with incomplete data, with model searching using penalized scoring functions. The algorithm is quite expensive, given that at each step the EM algorithm has to be executed in order to compute the maximum likelihood estimator, which is needed to compute the score of each network structure.

8.2.3 Network Scoring Metrics

Search-and-score methods applied to the task of learning structure from data require the definition of a scoring metric to assess at each step of the search process how well the current model approximates the data. The Bayesian approach gives a clear-cut answer to the problem of model selection. A scoring function can be associated with $p(\mathcal{G} \mid D)$, the posterior probability of the model given the data, where \mathcal{G} is a topology in the space of network topologies $\mathbb{S}^{\mathcal{G}}$.

$$\text{Score}(\mathcal{G}, D) \propto p(\mathcal{G} \mid D), \quad \mathcal{G} \in \mathbb{S}^{\mathcal{G}} \quad (8.3)$$

It is usually convenient to apply logarithms in order to simplify computations by replacing products by sums. The score can be therefore expressed in terms of the log posterior probability of the topology \mathcal{G} given the data set D . In other words,

$$\text{Best } \mathcal{G}^* \equiv \arg \max_{\mathcal{G} \in \mathbb{S}^{\mathcal{G}}} \log[p(\mathcal{G} \mid D)] \quad (8.4)$$

It should be stressed though that a fundamental feature of a score function is its decomposability. This means that the score function can be rewritten as a sum of contributions associated with each node X_i in the BN so that each contributing term depends only on the value x_i and the configurations of its parents $x_{pa(i)}$, given the data D and a DAG \mathcal{G} in the space of topologies $\mathbb{S}^{\mathcal{G}}$.

$$\text{Score}(\mathcal{G}, D) = \sum_{i=1}^n \text{LocalScoreTerm}(x_i, x_{pa(i)}; \mathcal{G}, D) \quad (8.5)$$

Cooper and Herskovits [5] make use of a Bayesian scoring metric and describe a method for computing the probability of BN topologies given a data set. The method considers a set of n discrete variables $\mathbf{X} = (X_1, X_2, \dots, X_n)$ of a BN with DAG \mathcal{G} , where each node $X_i \in \mathbf{X}$ in the network has r_i possible states, and has a set of parents $X_{pa(i)}$ for which there are q_i valid configurations of values $x_{pa(i)}$. The joint probability $p(\mathcal{G}, D)$ is taken over $\boldsymbol{\theta}$, the vector parameter given by the conditional

probabilities $p(x_i | x_{pa(i)}, \mathcal{G})$ in the BN $\langle \theta, \mathcal{G} \rangle$. Recall that, according to Bayes' theorem, $p(\mathcal{G} | D) \propto p(D | \mathcal{G}) \cdot p(\mathcal{G})$, where

$$p(D | \mathcal{G}) = \int_{\Theta} p(D | \theta, \mathcal{G}) \cdot p(\theta | \mathcal{G}) d\theta \tag{8.6}$$

is the marginal likelihood of the BN \mathcal{G} . Assuming that \mathcal{G} is uniformly distributed in the space of topologies $\mathbb{S}^{\mathcal{G}}$, so that $p(\mathcal{G}) = 1/|\mathbb{S}^{\mathcal{G}}|$, then in order to select the best network (the one with highest $p(\mathcal{G} | D)$), it is sufficient to find the maximum marginal likelihood $p(D | \mathcal{G})$, given by expression (8.6).

Consider the following assumptions:

- (i) The variables in the BN are discrete, constituting a multinomial BN
- (ii) Records in the data set occur independently, meaning that

$$p(D | \theta, \mathcal{G}) = \prod_{i=1}^N p(x_i | x_{pa(i)}, \mathcal{G});$$

- (iii) There are no missing values in the data set.
- (iv) The distribution of the parameters follows a Dirichlet distribution (conjugate prior for a multinomial). We use the notation:

$$\theta_i(j) \sim Di(\alpha_{ij_1}, \alpha_{ij_2}, \dots, \alpha_{ij_{r_i}}) \tag{8.7}$$

to identify each vector of parameters $\theta_i(j)$ following a Dirichlet distribution with hyperparameters $\alpha_{ij_1}, \alpha_{ij_2}, \dots, \alpha_{ij_{r_i}}$.

Given this set of assumptions, Cooper and Herskovits derive the marginal likelihood $p(D | \mathcal{G})$ in closed form as:

$$p(D | \mathcal{G}) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_i(j))} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ik}(j))}{\Gamma(\alpha_{ijk})} \tag{8.8}$$

where $\Gamma(\cdot)$ is the Gamma function², $N_{ik}(j)$ is number of cases in D in which $X_i \in \mathbf{X}$ has the value $x_i \equiv k$ for a given configuration of its parents $x_{pa(i)} \equiv j$, and $N_i(j) = \sum_{k=1}^{r_i} N_{ik}(j)$.

This approach of using $p(D | \mathcal{G})$ as a scoring function is useful for searching among a reduced number of network topologies, but when the space of topologies $\mathbb{S}^{\mathcal{G}}$

² The Gamma Function is defined as $\Gamma(\alpha) = \int_0^{\infty} x^{\alpha-1} e^{-x} dx$, where the expressions $\Gamma(1) = 1$; $\Gamma(\alpha + 1) = \alpha \cdot \Gamma(\alpha)$ apply for all α positive integers.

is large, searching $\mathbb{S}^{\mathcal{G}}$ exhaustively to find the DAG which maximizes the Bayesian scoring criterion becomes computationally impractical. For such reason, Cooper and Herskovits' K2 heuristic algorithm restricts the search by forcing an ordering of the nodes (see section 8.2.1).

8.2.3.1 Laplace's Approximation and Bayesian Information Criterion

Recalling the model selection criterion of choosing the network that renders the highest posterior probability $p(\mathcal{G} | D) \propto p(D | \mathcal{G}) \cdot p(\mathcal{G})$, the Gaussian (also called Laplace's) approximation is an efficient estimate of the marginal likelihood $p(\mathcal{G} | D)$. Laplace's approximation takes into account that: a) $p(\mathcal{G} | D)$ is averaged over the BN's vector parameter θ such that $p(D | \mathcal{G}) = \int_{\Theta} p(D | \theta, \mathcal{G}) \cdot p(\theta | \mathcal{G}) d\theta$; b) the product $p(D | \theta, \mathcal{G}) \cdot p(\theta | \mathcal{G})$ approximates a multivariate Gaussian distribution. This can be shown by expanding $\varphi(\theta) = \log[p(D | \theta, \mathcal{G}) \cdot p(\theta | \mathcal{G})]$ in Taylor series around the maximum posterior (MAP) estimator $\tilde{\theta}$ (note that the first order term vanishes when expanding around $\tilde{\theta}$):

$$\varphi(\theta) \cong \varphi(\tilde{\theta}) - \frac{1}{2}(\theta - \tilde{\theta})^t \nabla \nabla \varphi(\theta) \Big|_{\tilde{\theta}} (\theta - \tilde{\theta}) \tag{8.9}$$

The expression $(\theta - \hat{\theta})^t$ is the transpose of $(\theta - \hat{\theta})$, and $\nabla \nabla \varphi(\theta) \Big|_{\tilde{\theta}}$ is the Hessian matrix of second derivatives, whose (ij) th element is $\frac{\partial^2 \varphi(\theta)}{\partial \theta_i \partial \theta_j}$, evaluated at $\theta = \tilde{\theta}$. Taking the exponential of $\varphi(\theta)$:

$$e^{\varphi(\theta)} \cong p(D | \tilde{\theta}, \mathcal{G}) \cdot p(\tilde{\theta} | \mathcal{G}) \exp \left\{ -\frac{1}{2}(\theta - \tilde{\theta})^t \mathbf{A}(\theta - \tilde{\theta}) \right\} \tag{8.10}$$

Note that $\exp \left\{ -\frac{1}{2}(\theta - \tilde{\theta})^t \mathbf{A}(\theta - \tilde{\theta}) \right\}$ is proportional to a multivariate Gaussian density with $\mu = \tilde{\theta}$ and covariance matrix $\Sigma = \mathbf{A}$, where \mathbf{A} is equal to minus the inverse of the Hessian matrix evaluated at $\theta = \tilde{\theta}$. Therefore, replacing and integrating:

$$\begin{aligned} p(D | \mathcal{G}) &\cong p(D | \tilde{\theta}, \mathcal{G}) \cdot p(\tilde{\theta} | \mathcal{G}) \cdot \int_{\Theta} \exp \left\{ -\frac{1}{2}(\theta - \tilde{\theta})^t \mathbf{A}(\theta - \tilde{\theta}) \right\} d\theta \\ &\cong p(D | \tilde{\theta}, \mathcal{G}) \cdot p(\tilde{\theta} | \mathcal{G}) \cdot \frac{(2\pi)^{\frac{|\Theta|}{2}}}{\sqrt{(\det \mathbf{A})}} \end{aligned} \tag{8.11}$$

where $|\Theta|$ is the number of dimensions of vector parameter θ .

Taking logarithms we obtain the so called *Laplace's approximation* of the marginal likelihood:

$$\log p(D|\mathcal{G}) \cong \log p(D|\tilde{\boldsymbol{\theta}}, \mathcal{G}) + p(\tilde{\boldsymbol{\theta}}|\mathcal{G}) + \frac{|\boldsymbol{\theta}|}{2} \log(2\pi) - \frac{1}{2} \log(\det \mathbf{A}) \quad (8.12)$$

As shown by [10], Laplace's approximation is quite accurate for many problems with moderate sizes of data, exhibiting a relative error of order $O(1/N)$. A different version of expression (8.12) replaces \mathbf{A} by $\mathbf{I}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{\theta})$, the Fisher's information matrix evaluated at the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$. $\mathbf{I}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{\theta})$ is easier to calculate. As reported by [11], this approximation yields a larger relative error of order $O(1/\sqrt{N})$.

The *Bayesian Information Criterion* (BIC), first described by Schwartz [12], uses the fact that, as the size of the sample increases, the prior information is negligible relative to the likelihood of the data. Therefore, as N increases, it is possible to discard the use of parameter priors in expression (8.12). BIC's formulation results in:

$$\log p(D|\mathcal{G}) \cong \log p(D|\hat{\boldsymbol{\theta}}, \mathcal{G}) - \frac{|\boldsymbol{\theta}|}{2} \log(N) \quad (8.13)$$

As described by [13], BIC's formula has several interesting features: a) it eliminates the dependence of a prior choice; b) it is easier to calculate; c) it is intuitively attractive, as its expression characterizes the likelihood of the data penalized by the BN's complexity (given by the dimensionality of vector parameter $\boldsymbol{\theta}$).

8.2.3.2 Stochastic Complexity and the Minimum Description Length Principle

Occam's razor, the inductive bias of science applied to model selection, can be summarized as "choose the shortest hypothesis that fits the data" [14, pp 65]. The compromise between simplicity and complexity of a model is the trade-off between goodness of fit and generalization: if the model is too simple (description length required to explain data is too short) it may fail to encode the data accurately; if instead, the model is too complex (description length too long) it may be excessively tailored around the features of the sample data, failing to generalize for new observations, and therefore limiting its predictive power on future data. This criterion is easily related to the *stochastic complexity* of a model introduced by Rissanen [15] and the information theoretical principle generally referred to as Minimum Description Length Principle (MDL). Applying Bayes theorem to expression (8.4), taking logarithms and changing signs, we get:

$$\text{Best } \mathcal{G}^* = \arg \min_{\mathcal{G} \in \mathcal{S}^{\mathcal{G}}} [-\log p(D|\mathcal{G}) - \log p(\mathcal{G})] \quad (8.14)$$

Expression (8.14) can be analyzed from the perspective of information theory. If we think of a model in terms of a code capable of encoding a message (i.e. the data), a better model can be defined as the one that does a better job at eliminating the redundancies of the data, attaining a shorter description of the message (note that we use

“message” in a generic sense to mean the data to be compressed). Shannon’s noiseless coding theorem [16] showed that for any code encoding data $D = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}$ for which the Kraft inequality holds, the optimal code³ assigns a length of $-\log_2 p(\mathbf{x}^{(i)})$ to encode message $\mathbf{x}^{(i)}$ ^{4 5}. This clearly establishes a one-to-one relationship between the probability of a message and the description length of its optimal code: the expression implies that simpler hypotheses (i.e. short description lengths) correspond to large probability values and vice versa.

It follows that expression (8.14) can be interpreted in the following way: Select the BN with topology \mathcal{G}^* which minimizes the sum of the length of the description of the BN and the length of the description of the data when encoded by the BN [17]. Rewriting (8.14):

$$\text{Best } \mathcal{G}^* = \arg \min_{\mathcal{G} \in \mathbb{S}^{\mathcal{G}}} [L_{\mathcal{G}} + L_{D|\mathcal{G}}] \quad (8.15)$$

$L_{\mathcal{G}}$ is the description length of the BN with DAG \mathcal{G} under optimal encoding, and $L_{D|\mathcal{G}}$ is the description length of the data $D = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}$ given the BN with topology \mathcal{G} . This shortest description length is what Rissanen [15] called stochastic complexity, rooted in the work on algorithmic complexity by Solomonoff [18], Kolmogorov [19], and Chaitin [20], that Rissanen applied to those descriptions representing probability models, therefore equating the description of the data by a statistical model to the encoding of a message. Thus the purpose of model selection is to find models that can encode the data into short descriptions, and therefore compress the data.

8.2.3.3 Two-Part Coding and Minimum Description Length

The two-part coding scheme addresses the problem of encoding data with a chosen probability model so that the data strings can be transmitted with the shortest possible description length. To illustrate this let us consider the simple case⁶ of a set of observations $D = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}$ and a model $\mathcal{G} \in \mathbb{S}^{\mathcal{G}}$ consisting of a parametric family of distributions $f(\cdot | \boldsymbol{\theta})$, $\boldsymbol{\theta} \in \mathbb{R}^k$ and maximum likelihood (ML) estimator $\hat{\boldsymbol{\theta}}$ for the

³ The one with a mean code length equal to the lower bound given by the entropy.

⁴ It should be noted that Shannon’s theorem refers to log base 2, measuring the description length in *bits*. If natural logarithms are considered, the description length is measured in *nats*.

⁵ In general, we can think of $-\log_2 p(\mathbf{x})$ as the description length for ideal codes related to a probability density function.

⁶ It should be stressed that one of the strengths of the Minimum Descriptive Length Principle is that it can be generalized to far less restrictive settings [21].

data D . Recall that best codes are closely related to maximum likelihood estimators through:

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta} \in \Theta} p(D | \boldsymbol{\theta}) \equiv \arg \min_{\boldsymbol{\theta} \in \Theta} -\log p(D | \boldsymbol{\theta}) = \arg \min_{\boldsymbol{\theta} \in \Theta} L_{D | \boldsymbol{\theta}} \quad (8.16)$$

Part one of the scheme takes care of encoding the ML estimator $\hat{\boldsymbol{\theta}}$ whose prefix code of length $L_{\hat{\boldsymbol{\theta}}}$ is built after the probability model $f(\cdot | \hat{\boldsymbol{\theta}})$.

Part two takes care of encoding the data D using the distribution $f(\cdot | \hat{\boldsymbol{\theta}})$ such that its description length amounts to:

$$L_{D | \hat{\boldsymbol{\theta}}} = -\log p(D | \hat{\boldsymbol{\theta}}) \quad (8.17)$$

Rissanen [15, 22] showed that choosing a discretization⁷ of the parameter space with precision $\varepsilon = 1/\sqrt{N}$ (optimal for regular parametric families) leads to an encoding of the ML estimator $\hat{\boldsymbol{\theta}}$ with a minimum description length $L_{\hat{\boldsymbol{\theta}}} \cong \frac{|\boldsymbol{\theta}|}{2} \log N$, where $|\boldsymbol{\theta}|$ is the dimensionality of $\boldsymbol{\theta}$, and N is the data sample size. Putting both parts together we get a total minimum description length of:

$$\text{MDL} = -\log p(D | \hat{\boldsymbol{\theta}}) + \frac{|\boldsymbol{\theta}|}{2} \log N \quad (8.18)$$

We can see that expression (8.18) takes the form of a penalized likelihood, where the term $\frac{|\boldsymbol{\theta}|}{2} \log N$ is the price paid to encode the complexity of the model, besides encoding the data. It should also be noted that this version of MDL is equal to minus the Bayesian Information Criterion (BIC), as can be verified by comparing (8.18) with (8.13).

8.3 Geometric Complexity of a Bayesian Network

To properly interpret the meaning of the inherent complexity of a BN, we need to resort to geometry. This section describes some basic concepts of information geometry, a relatively new discipline that studies statistical inference from the point of view of modern differential geometry. The relevant differential geometry will be summarized here, together with its link to inferential statistics.

⁷ Real numbers cannot be encoded with codes of finite length. Therefore we need to resort to discretizing the parameter space.

8.3.1 Differential Geometry and Statistics

We start with some definitions taken from [23, 24, 25, 26, 27]:

A metric is a real-valued function $d(x,y)$ describing the distance between two points for a given set, that satisfy the following conditions:

- (i) $d(x, y) \geq 0$, with equality iff $x = y$
- (ii) Triangular inequality: $d(x, z) + d(z, y) \geq d(x, y)$
- (iii) Symmetry: $d(x, y) = d(y, x)$

An n -dimensional manifold \mathfrak{M} is defined as a topological, usually Hausdorff⁸ space with the property that for each point \mathcal{P} in \mathfrak{M} there exists an open neighborhood $U_{\mathcal{P}}$ in \mathfrak{M} and a mapping u which puts $U_{\mathcal{P}}$ into a one-to-one correspondence with a neighborhood in \mathbb{R}^n ⁹. This mapping of each point \mathcal{P} to an n -tuple of real numbers (u^1, u^2, \dots, u^n) is called a local coordinate system $\mathcal{P}(u^1, u^2, \dots, u^n)$ on \mathfrak{M} :

$$u: \mathcal{P} \rightarrow \bar{\mathbf{r}}(u^1, u^2, \dots, u^n) \quad (8.19)$$

For any point \mathcal{P} in \mathfrak{M} , the pair $(U_{\mathcal{P}}, u)$ is called a coordinate chart, and the collection of such pairs for which the set of neighborhoods $U_{\mathcal{P}}$ cover \mathfrak{M} is known as an atlas of \mathcal{P} (see Figure 8.2).

If $(U_{\mathcal{P}}, u)$ and $(V_{\mathcal{P}}, \bar{u})$ are two coordinate charts of \mathfrak{M} so that $U_{\mathcal{P}} \cap V_{\mathcal{P}} \neq \emptyset$, there is a one-to-one correspondence between the local coordinate systems $\mathcal{P}(u^1, u^2, \dots, u^n)$ and $\mathcal{P}(\bar{u}^1, \bar{u}^2, \dots, \bar{u}^n)$, such that $u^i = u^i(\bar{u}^1, \bar{u}^2, \dots, \bar{u}^n)$, $i = 1..n$ with inverse transformation given by $\bar{u}^i = \bar{u}^i(u^1, u^2, \dots, u^n)$, $i = 1..n$

For a differentiable manifold we assume that the transformations $u^i = u^i(\bar{u}^1, \bar{u}^2, \dots, \bar{u}^n)$ and their inverses $\bar{u}^i = \bar{u}^i(u^1, u^2, \dots, u^n)$ have continuous derivatives $\frac{\partial \bar{u}^i}{\partial u^j}$ and $\frac{\partial u^i}{\partial \bar{u}^j}$, and Jacobians satisfying $\det\left(\frac{\partial \bar{u}^i}{\partial u^j}\right) \neq 0$ and

$$\det\left(\frac{\partial u^i}{\partial \bar{u}^j}\right) \neq 0, \quad i, j = 1..n.$$

A differentiable curve (or trajectory) \mathcal{C} in an n -dimensional manifold \mathfrak{M} is a continuous mapping $\mathcal{C}(t)$ of an interval $a < t < b$ in \mathbb{R} into \mathfrak{M} . In terms of the local coordinates of a coordinate chart $(U_{\mathcal{P}}, u)$, the curve can be expressed as $\bar{\mathbf{r}}(u^1(t), u^2(t), \dots, u^n(t))$.

⁸ A topological space Ω is Hausdorff if for all $x, y \in \Omega$, with $x \neq y$, there exist open neighborhoods U_x, U_y in Ω such that $U_x \cap U_y = \emptyset$.

⁹ It is useful to picture a manifold as an extension of a surface embedded in high dimensional spaces.

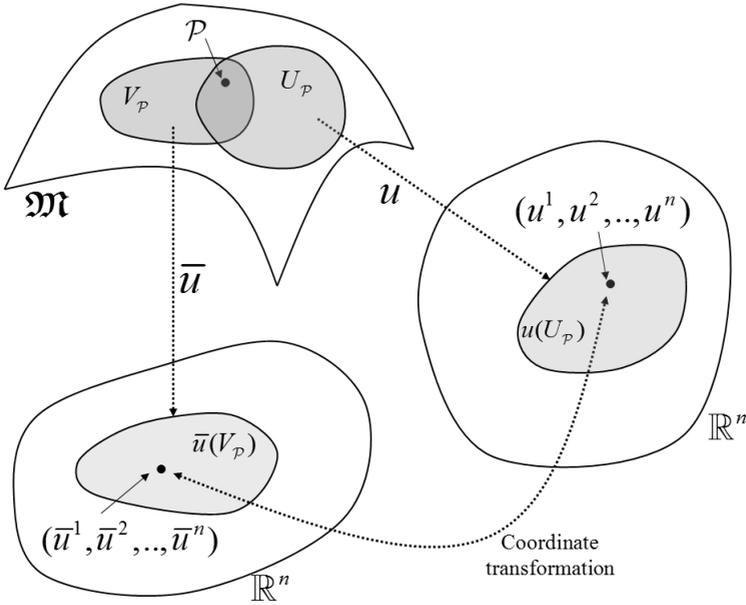


Fig. 8.2. Smooth Manifold

The tangent or velocity vector of a curve $C \equiv \vec{\mathbf{r}}(u^1(t), u^2(t), \dots, u^n(t))$ at time t is¹⁰:

$$\vec{\mathbf{v}} = \frac{d\vec{\mathbf{r}}}{dt} = \frac{\partial \vec{\mathbf{r}}}{\partial u^i} \cdot \frac{du^i}{dt} = \vec{\mathbf{r}}_i \cdot \frac{du^i}{dt}, \quad i = 1..n \tag{8.20}$$

where $\vec{\mathbf{r}}_i = \frac{\partial \vec{\mathbf{r}}}{\partial u^i}$ is a tangent vector of a curve $C_i(u_1, \dots, u_i(t), \dots, u_n)$ where only the value of the coordinate $u_i(t)$ changes, while the rest remain constant. The n tangent vectors $\vec{\mathbf{r}}_i$ form a basis for a vector space $T_p(\mathfrak{M})$ known as the tangent space of a differentiable manifold \mathfrak{M} at a point \mathcal{P} . The tangent space is then the vector space whose elements are velocities of curves in \mathfrak{M} that pass through \mathcal{P} (see Figure 8.3)

Under a reparameterization $\bar{u}^i = \bar{u}^i(u^1, u^2, \dots, u^n)$, the tangent vectors $\vec{\mathbf{r}}_i$ follow the transformation rule:

$$\vec{\bar{\mathbf{r}}}_j = \frac{\partial \vec{\bar{\mathbf{r}}}}{\partial \bar{u}_j} = \frac{\partial \vec{\mathbf{r}}}{\partial u_i} \frac{\partial u_i}{\partial \bar{u}_j} = \vec{\mathbf{r}}_i \frac{\partial u_i}{\partial \bar{u}_j} \tag{8.21}$$

¹⁰ For notational convenience, in expression (8.20) and the sequel in this section, we use Einstein's summation convention by which, if an index occurs twice in a term, one as a subscript and one as a superscript summation over that index is thereby assumed.

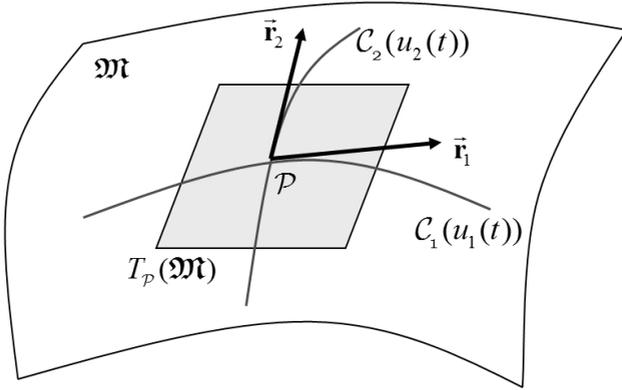


Fig. 8.3. Tangent space (adapted from [27])

The element of arc ds of a curve \mathcal{C} in an n -dimensional manifold \mathfrak{M} can be expressed as $ds = \left| \frac{d\vec{r}}{dt} \right| \cdot dt = |\vec{v}| \cdot dt$ such that the length of an arc of \mathcal{C} is equal to $s = \int_a^b |\vec{v}(t)| dt$, where $|\vec{v}(t)|$, the norm of $\vec{v}(t)$, is the square root of the inner product $\langle \vec{v}(t), \vec{v}(t) \rangle$.

Note that by application of (8.20), the element of arc ds satisfies:

$$\begin{aligned} ds^2 &= \langle d\vec{r}, d\vec{r} \rangle = (\vec{r}_i \cdot du^i)^2 = (\vec{r}_i \cdot \vec{r}_j) du^i du^j \\ &= g_{ij} du^i du^j \quad i, j = 1..n \end{aligned} \quad (8.22)$$

This bilinear¹¹ form $\Phi = g_{ij} du^i du^j$ given by the inner product $\langle \vec{v}(t), \vec{v}(t) \rangle$ is known as the first fundamental form, invariant under reparameterization (shown below), which induces a metric on the manifold, enabling us to measure lengths, angles and volumes on the manifold.

The resulting symmetric, positive definite matrix¹² of n^2 quantities $g_{ij}(u^1, u^2, \dots, u^n)$ forms a geometrical object called the metric tensor. Under a parameter transformation $\bar{u}^i = \bar{u}^i(u^1, u^2, \dots, u^n)$, we obtain:

¹¹ Given two vectors spaces V and W , for all vectors $\vec{v} \in V, \vec{w} \in W$, a real valued function $\psi(\vec{v}, \vec{w})$ is called a *bilinear form*, if for every fixed \vec{v} , it is a linear form on W , and for every fixed \vec{w} , it is a linear form on V . If $\psi(\vec{v}, \vec{w})$ is a bilinear form on V and $W=V$, then because of its linearity properties it can be written in the form $\psi(\vec{v}, \vec{w}) = \psi(v^i \vec{e}_i, w^j \vec{e}_j) = \psi(\vec{e}_i, \vec{e}_j) v^i w^j = g_{ij} v^i w^j$, where $(\vec{e}_1, \dots, \vec{e}_n)$ is a basis for V .

¹² Given 2 vectors \vec{a} and \vec{b} , (i) $\langle \vec{a}, \vec{b} \rangle = \langle \vec{b}, \vec{a} \rangle$; (ii) if $\vec{a} \neq 0, \langle \vec{a}, \vec{a} \rangle > 0$.

$$\bar{g}_{ij} = g(\bar{\mathbf{r}}_i, \bar{\mathbf{r}}_j) = g\left(\frac{\partial u^k}{\partial \bar{u}^i} \bar{\mathbf{r}}_k, \frac{\partial u^l}{\partial \bar{u}^j} \bar{\mathbf{r}}_l\right) = \frac{\partial u^k}{\partial \bar{u}^i} \frac{\partial u^l}{\partial \bar{u}^j} g(\mathbf{r}_k, \mathbf{r}_l) = \frac{\partial u^k}{\partial \bar{u}^i} \frac{\partial u^l}{\partial \bar{u}^j} g_{kl} \quad (8.23)$$

From (8.21) and (8.23) it follows that the first fundamental form is invariant:

$$\begin{aligned} ds^2 &= \bar{g}_{ij} d\bar{u}^i d\bar{u}^j = g_{kl} \frac{\partial u^k}{\partial \bar{u}^i} \frac{\partial u^l}{\partial \bar{u}^j} \frac{\partial \bar{u}^i}{\partial u^m} \frac{\partial \bar{u}^j}{\partial u^n} du^m du^n \\ &= g_{kl} \delta_i^k \delta_j^l du^i du^j = g_{ij} du^i du^j \end{aligned} \quad (8.24)$$

In expression (8.24), $\delta_i^k = 1$ if $i=j$, and $\delta_i^j = 0$ if $i \neq j$.

A manifold \mathfrak{M} on which there is defined a symmetric positive definite bilinear form $\psi(\bar{\mathbf{v}}, \bar{\mathbf{w}}) = g_{ij} v^i w^j$ is called a Riemannian manifold, and $\psi(\bar{\mathbf{v}}, \bar{\mathbf{w}})$ is called a Riemannian metric. We shall assume that ψ is infinitely differentiable. The simplest example is the inner product $\psi(\bar{\mathbf{v}}_\alpha, \bar{\mathbf{v}}_\beta) = \langle \bar{\mathbf{v}}_\alpha, \bar{\mathbf{v}}_\beta \rangle$ of two tangent vectors $\bar{\mathbf{v}}_\alpha, \bar{\mathbf{v}}_\beta \in T_p(\mathfrak{M})$.

The length of an arc of a curve \mathcal{C} in \mathfrak{M} bounded by $a < t < b$ is calculated by integrating expression (8.22) as:

$$s = \int_a^b \sqrt{\langle \bar{\mathbf{v}}(t), \bar{\mathbf{v}}(t) \rangle} dt = \int_a^b \sqrt{g_{ij} \frac{du^i}{dt} \frac{du^j}{dt}} dt \quad i, j = 1..n \quad (8.25)$$

The distance between two points \mathcal{P}_1 and \mathcal{P}_2 can be expressed as:

$$dist(\mathcal{P}_1, \mathcal{P}_2) = \inf \left\{ s(\mathcal{C}) : \mathcal{C} : [a, b] \rightarrow \mathfrak{M}, \text{ with } \mathcal{C}(a) = \mathcal{P}_1 \text{ and } \mathcal{C}(b) = \mathcal{P}_2 \right\} \quad (8.26)$$

Consider now the measurement of angles in the manifold. For that purpose, consider two curves \mathcal{C}_α and \mathcal{C}_β intersecting at point \mathcal{P} . Their tangent vectors $\bar{\mathbf{v}}_\alpha$ and $\bar{\mathbf{v}}_\beta$ at point \mathcal{P} can be expressed in terms of the tangent space basis at \mathcal{P} so that $\bar{\mathbf{v}}_\alpha = v_\alpha^i \cdot \bar{\mathbf{r}}_i$ and $\bar{\mathbf{v}}_\beta = v_\beta^j \cdot \bar{\mathbf{r}}_j$, $i = 1..n$.

If we denote by γ the angle between vectors $\bar{\mathbf{v}}_\alpha$ and $\bar{\mathbf{v}}_\beta$, then:

$$\cos \gamma = \frac{\bar{\mathbf{v}}_\alpha \cdot \bar{\mathbf{v}}_\beta}{|\bar{\mathbf{v}}_\alpha| \cdot |\bar{\mathbf{v}}_\beta|} = \frac{g_{ij} v_\alpha^i v_\beta^j}{\sqrt{g_{kl} v_\alpha^k v_\beta^l} \sqrt{g_{pq} v_\alpha^p v_\beta^q}} \quad (8.27)$$

The volume element of an n -dimensional manifold is defined by:

$$d\mathcal{V} = \sqrt{\det \mathbf{G}} du^1 du^2 .. du^n \quad (8.28)$$

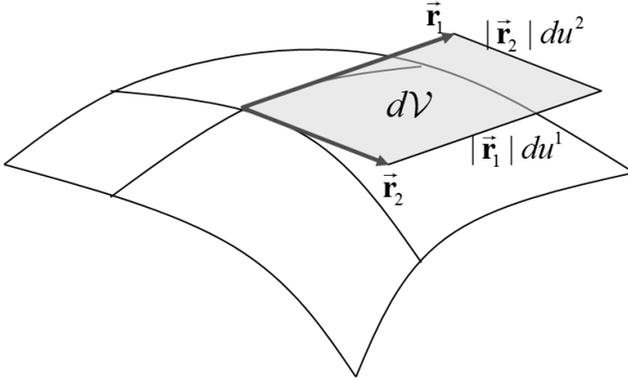


Fig. 8.4. Volume element of the manifold

where $\mathbf{G} = (g_{ij})$ is the metric tensor. This can be geometrically interpreted by viewing the manifold \mathfrak{M} , $\mathcal{P} \rightarrow \bar{\mathbf{r}}(u^1, u^2, \dots, u^n)$ as an n -dimensional surface in a space of dimension $N > n$ ¹³. Taking for example the simple case of a two-dimensional Riemannian manifold embedded in \mathbb{R}^N , expression (8.28) can be seen as the area of an ‘infinitesimal parallelogram’ whose sides are vectors $\bar{\mathbf{r}}_1 \cdot du^1$ and $\bar{\mathbf{r}}_2 \cdot du^2$ (see Figure 8.4).

This quantity can be calculated as the cross product:

$$\begin{aligned} d\mathcal{V} &= \det \left[(\bar{\mathbf{r}}_1 \cdot du^1) \times (\bar{\mathbf{r}}_2 \cdot du^2) \right] = \det \left[\bar{\mathbf{r}}_1 \times \bar{\mathbf{r}}_2 \right] du^1 du^2 \\ &= \sqrt{(\bar{\mathbf{r}}_1 \cdot \bar{\mathbf{r}}_1)(\bar{\mathbf{r}}_2 \cdot \bar{\mathbf{r}}_2) - (\bar{\mathbf{r}}_1 \cdot \bar{\mathbf{r}}_2)^2} du^1 du^2 \\ &= \sqrt{\det \mathbf{G}} du^1 du^2 \end{aligned} \quad (8.29)$$

From this, the result follows¹⁴.

Let us now try to make the link between differential geometry and statistics. For that purpose we will consider a family of probability distributions $\mathfrak{M} = \{p(\mathbf{x} | \boldsymbol{\theta})\}$ of

¹³ This poses the question of whether every Riemannian metric can be realized in terms of an n -dimensional manifold embedded isometrically in an Euclidean space \mathbb{R}^N . The initial proofs by Cartan [28] and Janet [29], showed that every sufficiently differentiable n -dimensional Riemannian manifold can be locally embedded in \mathbb{R}^N , with $N = (1/2) \cdot n \cdot (n + 1)$, having the nice derived feature that a Riemannian manifold of $n=2$ can be locally embedded as a surface in \mathbb{R}^3 . Nash’s *Imbedding Theorem* [30] proved that every compact smooth Riemannian manifold can be globally embedded in \mathbb{R}^N .

¹⁴ This simplified analysis, convenient to derive an intuitive explanation of basic differential geometry concepts, only makes sense for a space of dimension 3 or less.

a statistical model, indexed by a parameter $\boldsymbol{\theta} \in \mathbb{R}^n$, and the following regularity conditions [26, 27]:

- Given the log likelihood $\ell(\boldsymbol{\theta}) = \log p(x|\boldsymbol{\theta})$, for every fixed $\boldsymbol{\theta}$, the n partial derivatives $\partial_i \ell(\boldsymbol{\theta})$, $i = 1..n$ are linearly independent (we use $\partial_i \equiv \frac{\partial}{\partial \theta^i}$).
- The moments of $\partial_i \ell(\boldsymbol{\theta})$, $i = 1..n$ exist up to the necessary orders.
- The expression $\partial_i \int f(x, \boldsymbol{\theta}) dx = \int \partial_i f(x, \boldsymbol{\theta}) dx$ is valid for any function $f(x, \boldsymbol{\theta})$ considered in this analysis.
- The Fisher Information matrix is given by $[\mathbf{I}(\boldsymbol{\theta})]_{ij} = E_{\boldsymbol{\theta}} [\partial_i \partial_j \ell(\boldsymbol{\theta})]$

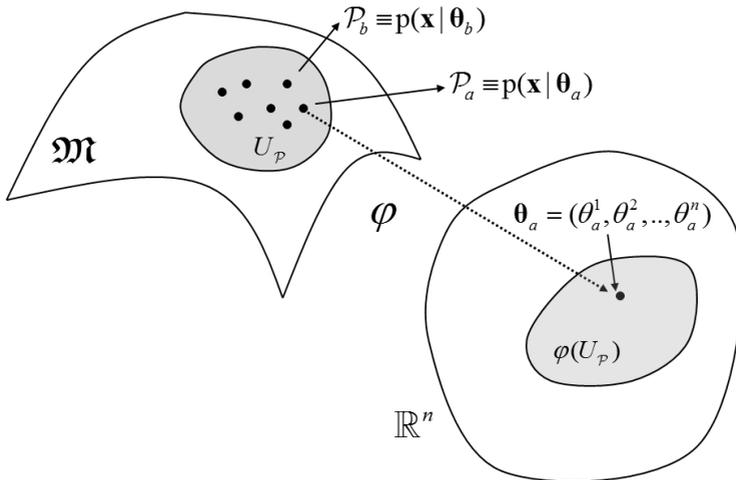


Fig. 8.5. Statistical Manifold

Given these conditions, if we consider a mapping $\varphi: \mathfrak{M} \rightarrow \mathbb{R}^n$ such that $\varphi(p(\mathbf{x}|\boldsymbol{\theta})) = \boldsymbol{\theta}$, the vector $\boldsymbol{\theta}$ is used as a coordinate system for the family $\{p(\mathbf{x}|\boldsymbol{\theta})\}$, forming a manifold \mathfrak{M} embedded in the space of probability distributions. The parameter $\boldsymbol{\theta}$ labels each point \mathcal{P} of \mathfrak{M} (see Figure 8.5).

Having established the connection between a parametric family of distributions and a manifold, we can try to identify the objects in the language of differential geometry that map those objects in the statistical domain:

The tangent space $T_{\mathcal{P}}(\mathfrak{M})$ at the point \mathcal{P} with coordinates $\boldsymbol{\theta} = \bar{\mathbf{r}}(\theta^1, \theta^2, \dots, \theta^n) \in \mathbb{R}^n$ corresponds to the vector space

$T_0^{(1)} = \{V(x) | V(x) = V^i \partial_i \ell(\theta)\}$ spanned by the functions in $x \partial_i \ell(\theta)$, $i = 1..n$, such that:

$$\bar{\mathbf{r}}_i \in T_p(\mathfrak{M}) \Leftrightarrow \partial_i \ell(\theta) \in T_0^{(1)} \tag{8.30}$$

The space $T_0^{(1)}$ is called by Amari [26] the 1-representation of the tangent space $T_p(\mathfrak{M})$.

The choice of $\partial_i \ell(\theta)$ as the basis for the vector space $T_0^{(1)}$, renders a 1-representation of a vector $V(x) \in T_0^{(1)}$ with vanishing expected value:

$$\begin{aligned} E[\partial_i \ell(\theta)] &= \int_x \partial_i [\log p(x|\theta)] p(x|\theta) dx \\ &= \partial_i \int_x p(x|\theta) dx = 0 \end{aligned} \tag{8.31}$$

Given two points \mathcal{P}_q and \mathcal{P}_p defining probability densities $q = p(x|\theta)$ and $p = p(x|\theta_p)$, with coordinates given by $\theta = \bar{\mathbf{r}}(\theta^1, \theta^2, \dots, \theta^n)$ and $\theta_p = \bar{\mathbf{r}} + d\bar{\mathbf{r}}$, where $d\bar{\mathbf{r}}$ is a tangent vector at \mathcal{P}_q , the Kullback number (relative entropy) $\mathcal{I}(p:q)$ between p and q is a nonnegative function with a minimum equal to zero at $\theta_p = \theta$. If we apply a Taylor expansion to second order on $\mathcal{I}(p:q)$ at the minimum $\theta_p = \theta$, the first two terms of the expansion vanish, yielding:

$$\mathcal{I}(p:q) \cong \frac{1}{2} d\bar{\mathbf{r}}^t \nabla \nabla \mathcal{I}(p:q) \Big|_{\theta} d\bar{\mathbf{r}} \tag{8.32}$$

where $\nabla \nabla \mathcal{I}(p:q) \Big|_p$ is the matrix of second derivatives $\left(\frac{\partial^2 \mathcal{I}(p:q)}{\partial \theta_i \partial \theta_j} \right)$ evaluated at θ . By applying straight forward computation on (32) we obtain the following quadratic form:

$$\begin{aligned} \mathcal{I}(p:q) &\cong \frac{1}{2} \left\{ \int_x \left(\frac{1}{p(x|\theta)} \partial_i p(x|\theta) - \frac{1}{p(x|\theta)} \partial_j p(x|\theta) \right) p(x|\theta) dx \right\} d\bar{\mathbf{r}}_i d\bar{\mathbf{r}}_j \\ &\cong \frac{1}{2} E[\partial_i \ell(\theta) \cdot \partial_j \ell(\theta)] d\bar{\mathbf{r}}_i d\bar{\mathbf{r}}_j \end{aligned} \tag{8.33}$$

If we define the inner product of the basis vectors $\partial_i \ell(\theta)$ and $\partial_j \ell(\theta)$ as:

$$\mathbf{g}_{ij}(\theta) = \langle d\bar{\mathbf{r}}_i, d\bar{\mathbf{r}}_j \rangle = E[\partial_i \ell(\theta) \cdot \partial_j \ell(\theta)] \tag{8.34}$$

This uniquely determines a Riemannian metric g_{ij} , invariant under reparameterization, where the matrix (g_{ij}) is the Fisher information matrix $\mathbf{I}(\boldsymbol{\theta})$. This can be shown as follows:

$$\begin{aligned}
 E[\partial_i \partial_j \ell(\boldsymbol{\theta})] &= \int_x \partial_i \partial_j [\log p(x|\boldsymbol{\theta})] p(x|\boldsymbol{\theta}) dx \\
 &= \int_x \partial_i \left[\frac{1}{p(x|\boldsymbol{\theta})} \partial_j p(x|\boldsymbol{\theta}) \right] p(x|\boldsymbol{\theta}) dx \\
 &= \int_x \left[\partial_i \partial_j p(x|\boldsymbol{\theta}) - \frac{1}{p(x|\boldsymbol{\theta})} \partial_i p(x|\boldsymbol{\theta}) \partial_j p(x|\boldsymbol{\theta}) \right] dx \tag{8.35} \\
 &= 0 - \int_x \left(\frac{1}{p(x|\boldsymbol{\theta})} \partial_i p(x|\boldsymbol{\theta}) \right) \left(\frac{1}{p(x|\boldsymbol{\theta})} \partial_j p(x|\boldsymbol{\theta}) \right) p(x|\boldsymbol{\theta}) dx \\
 &= -E[\partial_i \ell(\boldsymbol{\theta}) \partial_j \ell(\boldsymbol{\theta})]
 \end{aligned}$$

which coincides with expression (8.34), so that:

$$\mathbf{I}(\boldsymbol{\theta} | x) \equiv (g_{ij}) = E[\partial_i \ell(\boldsymbol{\theta}) \cdot \partial_j \ell(\boldsymbol{\theta})] = -E[\partial_i \partial_j \ell(\boldsymbol{\theta})]. \tag{8.36}$$

This means that by assuming that the Kullback number is the natural measure of separation between two probabilities in a parameter manifold, the induced local distance is equal to:

$$\mathcal{I}(p : q) \cong \frac{1}{2} g_{ij}(\boldsymbol{\theta}) d\bar{\mathbf{r}}_i d\bar{\mathbf{r}}_j \tag{8.37}$$

A more general derivation of these expressions can be found in [31].

The element of area in the manifold of probability distributions is given by:

$$d\mathcal{V} = \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta} \tag{8.38}$$

Normalizing this measure by dividing the volume of the model family $\int \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta}$ gives the so called Jeffreys' prior¹⁵.

¹⁵ Jeffreys [32] proposed to use the square root of the determinant of the Fisher information matrix as a non-informative prior. Roughly speaking, a prior distribution is non-informative if the prior is “flat” relative to the likelihood function; this means that the prior has small impact on the posterior distribution. Total ignorance seems to be best represented by using a uniform prior. But this condition is not sufficient: a uniform prior should also be uniform (i.e. remain invariant) under reparametrization.

Let $V_\alpha(x), V_\beta(x)$ be the 1-representations of $\bar{\mathbf{v}}_\alpha, \bar{\mathbf{v}}_\beta \in T_p(\mathfrak{M})$. Then their inner product is the covariance of the random variables $V_\alpha(x), V_\beta(x)$:

$$\langle \bar{\mathbf{v}}_\alpha, \bar{\mathbf{v}}_\beta \rangle = E[V_\alpha(x) \cdot V_\beta(x)] = \text{Covariance}[V_\alpha(x), V_\beta(x)] \tag{8.39}$$

given the fact that $E[V_\alpha(x)] = E[V_\beta(x)] = 0$. The length $|\bar{\mathbf{v}}_\alpha|$ of a tangent vector $\bar{\mathbf{v}}_\alpha$ is given by $|\bar{\mathbf{v}}_\alpha|^2 = \langle \bar{\mathbf{v}}_\alpha, \bar{\mathbf{v}}_\alpha \rangle = E[(V_\alpha(x))^2]$, which is the variance of the random variable V_α .

8.3.2 The Info-geometric Version of the Minimum Description Principle

The formulation of the Minimum Description Length Principle has gone through several iterations. The two-part code MDL version described before has the same form as the Bayesian Information Criterion (BIC)¹⁶ which shows that the model attaining:

$$\min_{\theta, \hat{\theta}} \left\{ -\log p(D | \hat{\theta}) + \frac{|\theta|}{2} \log N \right\} \tag{8.40}$$

provides the most efficient encoding of the data D .

Rissanen [33] has proposed the following version of the MDL score:

$$MDL = -\log p(D | \hat{\theta}) + \frac{|\theta|}{2} \log \frac{N}{2\pi} + \log \int \sqrt{\det \mathbf{I}(\theta)} d\theta + o(1) \tag{8.41}$$

The first term is minus the logarithm of the maximum likelihood, the second term measures the complexity given by the number of parameters in the model and the last term measures the complexity given by geometrical properties of the model (recall that $\int \sqrt{\det \mathbf{I}(\theta)} d\theta$ is the volume of the model manifold). As can be seen, the expression of the MDL is given by a log likelihood penalized by factors associated to the complexity of the model, and in that sense reinforces the bias towards selecting simpler models. The merit of this version of MDL is to quantify complexity by considering both the number of parameters and the geometry of the hypothesis space.

The complexity penalty measure can be explained heuristically [34] by considering the Kullback distance between the true distribution $p_{\theta_0} \equiv p(D | \theta_0)$ and the parametric family of probability distributions $p_{D,\theta} = \{p(D | \theta, \mathcal{G}), \theta \in \Theta\}$, with a prior $p(\theta | \mathcal{G})$ on Θ , which describes a data set D of N independent and identically distributed samples. Recalling that the marginal likelihood of the data can be

¹⁶ This has spurred some confusion, leading some people to think that MDL and BIC are the same measure.

expressed as $p(D|\mathcal{G}) = \int_{\Theta} p(D|\boldsymbol{\theta}, \mathcal{G}) \cdot p(\boldsymbol{\theta}|\mathcal{G}) d\boldsymbol{\theta}$, and assuming that the Central Limit Theorem holds for the maximum likelihood estimators of the family $p_{D,\boldsymbol{\theta}}$, it turns out that the Kullback number $\mathcal{I}(p_{\boldsymbol{\theta}_0} : p_{D,\boldsymbol{\theta}})$ can be expressed as:

$$\begin{aligned} \mathcal{I}(p_{\boldsymbol{\theta}_0} : p_{D,\boldsymbol{\theta}}) &= E_{\boldsymbol{\theta}_0} \left[\log \frac{p(D|\boldsymbol{\theta}_0)}{p(D|\boldsymbol{\theta}, \mathcal{G})} \right] \\ &\cong -\frac{|\boldsymbol{\theta}|}{2} \log \frac{N}{2\pi} - \frac{1}{2} \log \det \mathbf{I}(\boldsymbol{\theta}) + \log \frac{1}{p(\boldsymbol{\theta}|\mathcal{G})} \end{aligned} \tag{8.42}$$

If we let the prior $p(\boldsymbol{\theta}|\mathcal{G})$ be the normalized Jeffreys’ prior given by:

$$p(\boldsymbol{\theta}|\mathcal{G}) = \frac{\sqrt{\det \mathbf{I}(\boldsymbol{\theta})}}{\int \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta}} \tag{8.43}$$

then the Kullback distance $\mathcal{I}(p_{\boldsymbol{\theta}_0} : p_{D,\boldsymbol{\theta}})$ is equal to minus the model complexity penalty of the Minimum Description Length:

$$\mathcal{I}(p_{\boldsymbol{\theta}_0} : p_{D,\boldsymbol{\theta}}) \cong -\frac{|\boldsymbol{\theta}|}{2} \log \frac{N}{2\pi} - \log \int \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta} \tag{8.44}$$

Balasubramanian [34] introduces a measure of model complexity closely connected to expression (8.44), which he calls the “razor” of a model. Balasubramanian’s razor takes the form:

$$R_N(\mathcal{G}) = \frac{\int e^{-N\mathcal{I}(p_{\boldsymbol{\theta}_0}; p_{D,\boldsymbol{\theta}})} \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta}}{\int \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta}} \tag{8.45}$$

where, as before, $\mathcal{I}(p_{\boldsymbol{\theta}_0} : p_{D,\boldsymbol{\theta}^*})$ is the Kullback number between the true distribution $p_{D,\boldsymbol{\theta}}$ and the distribution $p_{D,\boldsymbol{\theta}^*}$ indexed by $\boldsymbol{\theta}^*$ that minimizes $\mathcal{I}(p_{\boldsymbol{\theta}_0} : p_{D,\boldsymbol{\theta}})$. His Taylor expansion of the razor renders the following expression, given by $\chi_N(\mathcal{G}) = -\log R_N(\mathcal{G})$:

$$\begin{aligned} \chi_N(\mathcal{G}) &= -\log p(D|\hat{\boldsymbol{\theta}}) - N \cdot h(p_{\boldsymbol{\theta}_0}) + \frac{|\boldsymbol{\theta}|}{2} \log \frac{N}{2\pi} + \\ &\frac{1}{2} \log \frac{1}{\mathcal{K}_y} + \log \int \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta} + \text{higher order terms} \end{aligned} \tag{8.46}$$

where $h(p_{\boldsymbol{\theta}_0})$ is the differential entropy calculated on the true distribution, and \mathcal{K}_y is the fraction of the volume of the model manifold occupied by distinguishable probability distributions lying close to the truth (two probability distributions are

indistinguishable if it cannot be established which of them generated a given data set when the sample data size grows indefinitely)¹⁷.

It is easily seen that if we discard $N \cdot h(p_{\theta_0})$, which depends on the true distribution, and we disregard $\left(\frac{1}{2} \log \frac{1}{\mathcal{K}_{\mathcal{V}}} \right)$, we obtain the same score as the one provided by Rissanen's info-geometric MDL, as shown in expression (8.41). As such, the MDL score can be seen in a certain sense as an approximation of the razor of a model.

According to the info-geometric version of MDL we would therefore prefer models that:

- Are accurate in describing the data, measured in terms of a larger likelihood of the data, or a smaller relative entropy between the true distribution and the distribution on the model manifold.
- Are simpler (fewer parameters), given the fact that smaller models are computationally easier to handle, are less prone to overfitting the data and to the consequent generalization error, and will generally have fewer local maxima in which an estimation procedure can get stuck.
- Hold a smaller volume in the space of distributions and are therefore more constrained.

8.3.3 The Info-geo Scoring Metric

We propose the use of a scoring metric based on the info-geometric version of the Minimum Description Length Principle, taking into account the volume of the BN's manifold. The *info-geo* score is a maximum likelihood function, penalized by the dimensionality of θ and the volume of the statistical manifold. With complete, discrete data, the likelihood term is reduced to a sum of local counts that consider the number of times a given node takes a certain value for a given configuration of its parent values; these values are easily computed. The dimensionality $|\theta|$ is obtained in a straightforward manner by adding up the product of rows and columns of the conditional probability tables at each node. Therefore our efforts should concentrate in finding a proper way of computing the volume $\mathcal{V} = \log \int \sqrt{\det \mathbf{I}(\theta)} d\theta$ at each step of the search procedure.

In order to exemplify the computation of the volume \mathcal{V} for a given DAG, let us consider the case of the discrete, binary BN depicted in Figure 8.6.

The BN can be expressed in terms of its joint probability as:

$$\begin{aligned} p(x_1, x_2) &= p(x_2 | x_1) \cdot p(x_1) \\ &= \theta_{2x_2}(x_1) \cdot \theta_{1x_1} \end{aligned} \tag{8.47}$$

¹⁷ For more details as to how to estimate $\mathcal{K}_{\mathcal{V}}$, see [34].

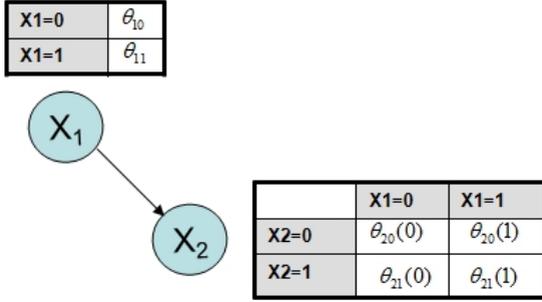


Fig. 8.6. Computing the volume \mathcal{V} of a binary BN of two nodes

or equivalently:

$$p(x_1, x_2) = [\theta_{20}(x_1)]^{1-x_2} \cdot [\theta_{21}(x_1)]^{x_2} \cdot [\theta_{10}]^{1-x_1} \cdot [\theta_{11}]^{x_1} \tag{8.48}$$

Acknowledging that $\theta_{11} = 1 - \theta_{10}$, and $\theta_{21}(x_1) = 1 - \theta_{20}(x_1)$, this means that:

$$\begin{cases} p(x_1 = 0, x_2 = 0 | \theta) = (1 - \theta_{11}) \cdot (1 - \theta_{21}(0)) \\ p(x_1 = 0, x_2 = 1 | \theta) = (1 - \theta_{11}) \cdot \theta_{21}(0) \\ p(x_1 = 1, x_2 = 0 | \theta) = \theta_{11} \cdot (1 - \theta_{21}(1)) \\ p(x_1 = 1, x_2 = 1 | \theta) = \theta_{11} \cdot \theta_{21}(1) \end{cases} \tag{8.49}$$

The Fisher information matrix $\mathbf{I}(\theta)$, is in this case a 3 x 3 matrix with elements

$I_{\theta_{ix_i}, \theta_{jx_j}}$:

$$I_{\theta_{ix_i}, \theta_{jx_j}} = -E_{\theta} \left[\frac{\partial^2 \ell(\theta)}{\partial \theta_{ix_i}(x_{pa(j)}) \partial \theta_{jx_j}(x_{pa(j)})} \right], \text{ for } \theta_{11}, \theta_{21}(0), \theta_{21}(1) \tag{8.50}$$

Taking logarithms on (8.49), we compute the log likelihood $\ell(\theta)$ as:

$$\ell(\theta) = \begin{cases} \log(1 - \theta_{11}) + \log(1 - \theta_{21}(0)) & \text{if } (x_1, x_2) = (0, 0) \\ \log(1 - \theta_{11}) + \log \theta_{21}(0) & \text{if } (x_1, x_2) = (0, 1) \\ \log \theta_{11} + \log(1 - \theta_{21}(1)) & \text{if } (x_1, x_2) = (1, 0) \\ \log \theta_{11} + \log \theta_{21}(1) & \text{if } (x_1, x_2) = (1, 1) \end{cases} \tag{8.51}$$

Using (8.51), we calculate the expected value of the second derivative to obtain the component $I_{\theta_{11}, \theta_{11}}$ of the Fisher information matrix $\mathbf{I}(\theta)$:

$$\frac{\partial \ell(\boldsymbol{\theta})}{\partial \theta_{11}} = \begin{cases} \frac{-1}{1-\theta_{11}} \\ \frac{-1}{1-\theta_{11}} \\ \frac{1}{\theta_{11}} \\ \frac{1}{\theta_{11}} \end{cases} \Rightarrow \frac{\partial \ell^2(\boldsymbol{\theta})}{\partial \theta_{11}^2} = \begin{cases} \frac{-1}{(1-\theta_{11})^2} \\ \frac{-1}{(1-\theta_{11})^2} \\ \frac{-1}{\theta_{11}^2} \\ \frac{-1}{\theta_{11}^2} \end{cases} \Rightarrow I_{\theta_{11}, \theta_{11}} = \begin{cases} \frac{-1}{(1-\theta_{11})^2} \cdot (1-\theta_{11}) \cdot (1-\theta_{21}(0)) + \\ \frac{-1}{(1-\theta_{11})^2} \cdot (1-\theta_{11}) \cdot \theta_{21}(0) + \\ \frac{-1}{\theta_{11}^2} \cdot \theta_{11} \cdot (1-\theta_{21}(1)) + \\ \frac{-1}{\theta_{11}^2} \cdot \theta_{11} \cdot \theta_{21}(1) \end{cases} \quad (8.52)$$

This renders:

$$\begin{cases} I_{\theta_{11}, \theta_{11}} = \frac{1}{\theta_{11}(1-\theta_{11})} \\ I_{\theta_{11}, \theta_{21(0)}} = 0 \\ I_{\theta_{11}, \theta_{21(1)}} = 0 \end{cases} \quad (8.53)$$

We calculate the other matrix elements in a similar fashion:

$$\begin{cases} I_{\theta_{21(0)}, \theta_{21(0)}} = \frac{(1-\theta_{11})}{\theta_{21}(0) \cdot (1-\theta_{21}(0))} \\ I_{\theta_{21(0)}, \theta_{11}} = 0 \\ I_{\theta_{21(0)}, \theta_{21(1)}} = 0 \end{cases} \quad (8.54)$$

$$\begin{cases} I_{\theta_{21(1)}, \theta_{21(1)}} = \frac{\theta_{11}}{\theta_{21}(1) \cdot (1-\theta_{21}(1))} \\ I_{\theta_{21(1)}, \theta_{11}} = 0 \\ I_{\theta_{21(1)}, \theta_{21(0)}} = 0 \end{cases} \quad (8.55)$$

Being that all the $I_{\theta_{\alpha_i}, \theta_{\beta_j}}$ elements outside the principal diagonal are equal to zero, the determinant of $\mathbf{I}(\boldsymbol{\theta})$ is equal to $I_{\theta_{11}, \theta_{11}} \times I_{\theta_{21(0)}, \theta_{21(0)}} \times I_{\theta_{21(1)}, \theta_{21(1)}}$:

$$\mathbf{I}(\boldsymbol{\theta}) = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{\theta_{11}(1-\theta_{11})} & 0 & 0 \\ 0 & \frac{(1-\theta_{11})}{\theta_{21}(0) \cdot (1-\theta_{21}(0))} & 0 \\ 0 & 0 & \frac{\theta_{11}}{\theta_{21}(1) \cdot (1-\theta_{21}(1))} \end{bmatrix} \quad (8.56)$$

This means that the volume of the manifold, calculated by integrating the square root of the determinant of $\mathbf{I}(\boldsymbol{\theta})$ over the vector parameter $\boldsymbol{\theta}$, is equal to:

$$\mathcal{V} = \int_{\Theta} \left[\frac{1}{\theta_{21}(0) \cdot (1 - \theta_{21}(0)) \cdot \theta_{21}(1) \cdot (1 - \theta_{21}(1))} \right]^{\frac{1}{2}} d\boldsymbol{\theta} \tag{8.57}$$

The multiple integral in expression (8.57) can be simplified as follows:

$$\begin{aligned} \mathcal{V} &= \left\{ \int_0^1 [\theta_{21}(0)]^{(1/2-1)} [1 - \theta_{21}(0)]^{(1/2-1)} d\theta_{21}(0) \right\} \times \\ &\quad \left\{ \int_0^1 [\theta_{21}(1)]^{(1/2-1)} [1 - \theta_{21}(1)]^{(1/2-1)} d\theta_{21}(1) \right\} \\ &= \text{Beta}(1/2) \cdot \text{Beta}(1/2) = \pi^2 \end{aligned} \tag{8.58}$$

Rodríguez [35] derives the following exact formula for the volume element of a Bayesian Net of n discrete nodes:

$$\begin{aligned} d\mathcal{V} &= \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta} \\ &= \prod_{i=1}^n \prod_{x_{pa(i)}} \frac{\left\{ \sum_{x_{ap(i)}} \prod_{j \in an(j)} \theta_{jx_j}(x_{pa(j)}) \right\}^{(r_i-1)/2}}{\prod_{k=0}^{r_i-1} [\theta_{ik}(x_{pa(i)})]^{1/2}} d\boldsymbol{\theta} \end{aligned} \tag{8.59}$$

The following convention is used:

- There are n variables in the network, $X_i, i = 1..n$
- $pa(i)$ denotes the set of parents of each node $X_i = x_i$
- $an(i)$ denotes the set of ancestors (including parents) of each node x_i
- $ap(i)$ denotes the set of ancestors not parents of each node x_i such that $ap(i) = an(i) \setminus pa(i)$.
- r_i denotes the number of possible states $x_i = 0, 1, \dots, r_i - 1$ of variable X_i
- $\theta_{ik}(x_{pa(i)})$ is an entry in the conditional probability table for node X_i taking value $x_i = k$ for each of the q_i configurations of its parents $x_{pa(i)}$
- $\sum \equiv \sum_{x_{pa(i)}} \sum_{x_s} \sum_{x_r} \dots \sum_{x_i}$ denotes the multiple sum over all possible values of $x_{pa(i)} = \{x_r, x_s, \dots, x_i\}$

Expression (8.59) can be obtained by calculating the Fisher Information matrix as

minus the expectation $E_{\boldsymbol{\theta}} \left[\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \theta_{ix_i}(x_{pa(j)}) \partial \theta_{jx_j}(x_{pa(j)})} \right]$, where:

$$\ell(\boldsymbol{\theta}) = \log \prod_{i=1}^n p(x_i | x_{pa(i)}) = \sum_{i=1}^n \log \theta_{ik}(x_{pa(i)}) \quad (8.60)$$

Taking first derivatives over the multinomial expression of the likelihood, we verify that:

$$\frac{\partial \ell(\boldsymbol{\theta})}{\partial \theta_{ik}(x_{pa(i)})} = \begin{cases} -1/\theta_{i0}(x_{pa(i)}) & \text{if } x_i = 0 \text{ and a given } x_{pa(i)} \\ 1/\theta_{ik}(x_{pa(i)}) & \text{if } x_i = k \neq 0, \text{ and a given } x_{pa(i)} \\ 0 & \text{otherwise} \end{cases} \quad (8.61)$$

As the first derivative is only dependent on $\theta_{ik}(x_{pa(i)})$, the second derivative vanishes in all cases except $\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \theta_{ik}(x_{pa(i)})^2}$. For the second derivatives we get the following expression:

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \theta_{ik}(x_{pa(i)})^2} = \begin{cases} -1/\theta_{i0}^2(x_{pa(i)}) & \text{if } x_i = 0 \text{ and a given } x_{pa(i)} \\ -1/\theta_{ik}^2(x_{pa(i)}) & \text{if } x_i = k \neq 0, \text{ and a given } x_{pa(i)} \\ 0 & \text{otherwise} \end{cases} \quad (8.62)$$

Taking expectations on (8.62):

$$\begin{aligned} E_{\boldsymbol{\theta}} \left[\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \theta_{ik}(x_{pa(i)})^2} \right] &= -\frac{1}{\theta_{i0}(x_{pa(i)})} \cdot p(x_i = 0, x_{pa(i)} | \boldsymbol{\theta}) - \\ &\quad \frac{1}{\theta_{ik}(x_{pa(i)})} \cdot p(x_i = k, x_{pa(i)} | \boldsymbol{\theta}) \\ &= -\left(\frac{1}{\theta_{i0}(x_{pa(i)})} + \frac{1}{\theta_{ik}(x_{pa(i)})} \right) \cdot p(x_{pa(i)} | \boldsymbol{\theta}) \end{aligned} \quad (8.63)$$

and computing the determinant of the Fisher information as the product of the main diagonal we get:

$$\begin{aligned} \det \mathbf{I}(\boldsymbol{\theta}) &= \prod_{i=1}^n \prod_{x_{pa(i)}} \left\{ \prod_{k=1}^{r_i-1} \left[\frac{1}{\theta_{i0}(x_{pa(i)})} + \frac{1}{\theta_{ik}(x_{pa(i)})} \right] \right\} \cdot p(x_{pa(i)} | \boldsymbol{\theta}) \\ &= \prod_{i=1}^n \prod_{x_{pa(i)}} \frac{p(x_{pa(i)} | \boldsymbol{\theta})}{\prod_{k=0}^{r_i-1} \theta_{ik}(x_{pa(i)})} \end{aligned} \quad (8.64)$$

Note that $p(x_{pa(i)} | \boldsymbol{\theta})$ can be expressed as:

$$\begin{aligned}
 p(x_{pa(i)} | \boldsymbol{\theta}) &= \sum_{x_{ap(i)}} p(x_{pa(i)}, x_{ap(i)} | \boldsymbol{\theta}) \\
 &= \sum_{x_{ap(i)}} \left[\prod_{j \in an(i)} \theta_{jx_j}(x_{pa(j)}) \right]
 \end{aligned}
 \tag{8.65}$$

Replacing (8.65) in (8.64) we obtain:

$$\det \mathbf{I}(\boldsymbol{\theta}) = \prod_{i=1}^n \prod_{x_{pa(i)}} \frac{\sum_{x_{ap(i)}} \left[\prod_{j \in an(i)} \theta_{jx_j}(x_{pa(j)}) \right]}{\prod_{k=0}^{r_i-1} \theta_{ik}(x_{pa(i)})}
 \tag{8.66}$$

from where (8.59) follows.

To compute the volume $\mathcal{V} = \log \int \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta}$, we need to calculate the integral of expression (8.59) which, in most cases, does not have a closed form solution. We therefore need to resort to computing an estimate by simulation, applying Markov chain Monte Carlo for example. This solution may be too expensive if the number of iterations in each MCMC run is significant. An alternative solution consists in calculating an approximate value of the volume based on computing high and low bounds of the integral. An algorithm by Rodríguez [36] takes as input parameter the adjacency matrix of a discrete BN with binary nodes, and returns a low and high bound of the volume together with the geometric mean of both values. The geometric mean is subsequently used to approximate the value of the volume. The algorithm takes into account the fact that the analytical expression of the volume of a discrete binary DAG can be assimilated to a factorization of Beta integrals (see the result of the example at the beginning of this section). The algorithm was benchmarked by comparing its outcome with exact solutions obtained by integrating expression (8.59), and with estimates drawn from MCMC simulations, and has proven to return adequate approximations of the true volume of the BN under analysis [37].

8.3.4 Assessing the Importance of the Geometric Term in the Info-geo Score Function

As we see from expression (8.59), the volume of the manifold is a geometric characteristic associated with the BN’s topology. Each BN produces a different magnitude of the volume based on the BN’s DAG. This means that if the geometric term is relevant enough compared to the other two terms of the MDL expression, it may very well be a critical factor in providing an accurate discrimination of competing BNs in the model selection process. Let us consider the general expression of the *info-geo* MDL score:

$$\text{info-geo MDL} = -\log p(D | \hat{\boldsymbol{\theta}}) + \frac{|\boldsymbol{\theta}|}{2} \log \frac{N}{2\pi} + \log \int \sqrt{\det \mathbf{I}(\boldsymbol{\theta})} d\boldsymbol{\theta}
 \tag{8.67}$$

For large enough sample sizes, the log likelihood term which increases linearly with N , dominates the score. The $O(\log N)$ term, $\frac{|\theta|}{2} \log \frac{N}{2\pi}$ tells us that the value of the score increases linearly in the dimension of the parameter space. But what happens with the geometrical term? The formulation of the geometric term suggests a preference for BNs with a smaller manifold volume. Very small volumes (close enough to zero) would imply a geometric term that adds substantially to the log likelihood instead of penalizing it. In the limit, a volume equal to zero would imply a negative infinite geometrical term that would evidently control the score [37].

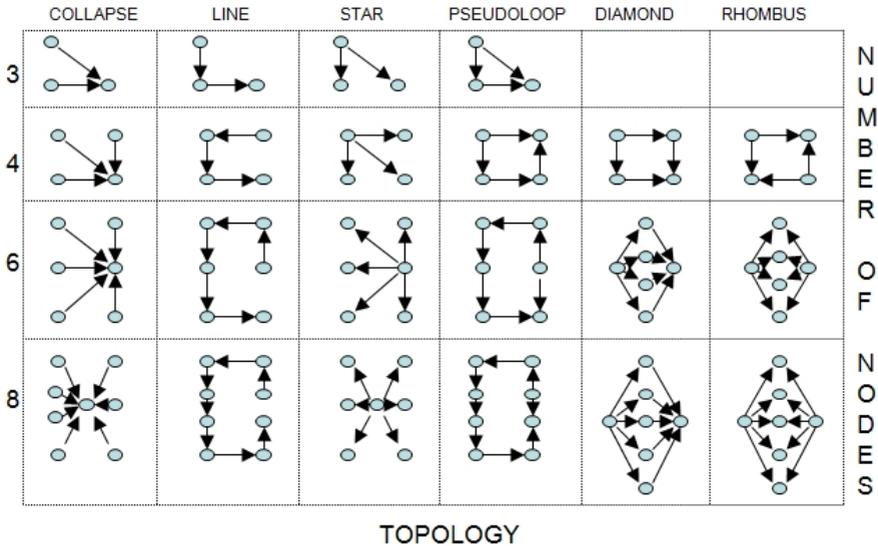


Fig. 8.7. Bayesian Networks with varying topologies (DAGs) and number of nodes

The implication of this analysis is quite straightforward: if the topology of the BN that we are trying to learn from sample data is such that the volume of the statistical manifold is small enough, for moderate sample sizes, the geometric term may provide a major contribution to the *info-geo* scoring function, therefore enhancing the discrimination accuracy of well tested scoring functions such as BIC.

To clarify this matter, let us consider the set of binary BN structures depicted in Figure 8.7. The group of DAGs has been organized in categories (*collapse*, *line*, *star*, *pseudoloop*, *diamond*, *rhombus*) according to their topologies. The set is by no means an exhaustive list of possible DAGs (the number of DAGs as a function of the number of nodes n is super-exponential in n), but it does represent a characteristic set of topologies: DAGs with all the edges converging to one node (*collapse*), DAGs with all the edges diverging from one node (*star*), DAGs with nodes linked sequentially (*line*), DAGs with nodes organized in a structure close to a cycle (*pseudoloop*), edges diverging from one node into a set of nodes and then converging to one last node (*diamond*), DAGs with two nodes feeding the rest of the nodes (*rhombus*).

Table 8.1. Approximate values of log(volume) for BNs with varying topologies and number of nodes

Topology	3 nodes		4 nodes		6 nodes		8 nodes	
	Volume	LogVol	Volume	LogVol	Volume	LogVol	Volume	LogVol
COLLAPSE	15.02	2.71	3.79	1.33	5.99E-10	-21.24	1.71E-75	-172.16
LINE	37.15	3.61	139.83	4.94	1981.21	7.59	28070.16	10.24
STAR	38.25	3.64	160.23	5.08	3121.60	8.05	65158.37	11.08
PSEUDOLOOP	16.23	2.79	95.99	4.56	1360.01	7.22	19268.85	9.87
DIAMOND	-	-	94.38	4.55	0.29	-1.24	8.51E-27	-60.03
RHOMBUS	-	-	51.44	3.94	1013.76	6.92	25574.13	10.15

Table 8.1 displays the approximate volume of each DAG for each of these categories and for a varying number of nodes (3, 4, 6 and 8). A chart tracing the values of the log(volume) as a function of the number of nodes is shown in Figure 8.8.

It is quite evident from Table 8.1 that those topologies (namely *collapse* and *diamond*) with a large maximum indegree (number of parents feeding a given node) have volumes which decrease in a super-exponential manner with the number of nodes.

As shown in Figure 8.8, if the number of nodes is big enough, for a moderate value of indegree the geometrical term could provide a substantial contribution of to the total score, which could in turn enhance the accuracy of the model searching process. The inclusion of the log(volume) as part of the score may be the difference, in such cases, between succeeding or failing in finding the underlying structure of the model.

In all the cases where the maximum number of fan-in nodes is low, the volume remains within a bounded interval, suggesting that the contribution to the score, given by the log(volume), is close to null. This means that in these cases it should be reasonable to imagine a smaller difference in the performance of the *info-geo* scoring function relative to the BIC score.

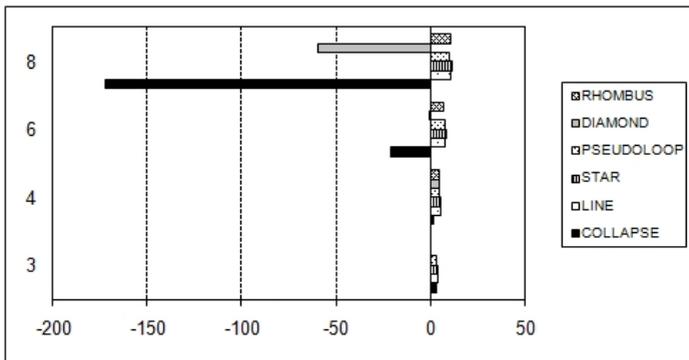


Fig. 8.8. Log(volume) as a function of the number of nodes (extracted from [37])

Lauria [37] checked the practical validity of the *info-geo* scoring function, benchmarking it against BIC on a series of experiments using binary Bayesian networks. The experiments tested the relative accuracy of the scoring metrics when varying the size and complexity of the BN. The general setup for the experiments was the following:

- Given a binary BN with a predefined topology and random network parameters (conditional probability tables), the BN was sampled to generate a fairly large collection of simulated data sets.
- The sampled data was used repeatedly to heuristically search the space of network topologies. The goal was to recover the DAG that best fitted the sampled data.

The tests showed that the *info-geo* score surpassed BIC for all data sets of moderate size (100 – 500) samples. As predicted, for larger data sets, the log-likelihood term dominated the score, dwarfing the contribution of the geometric term and therefore eliminating the difference between both scores. The tests confirmed also that in the case of the *collapse* and *diamond* structures, the *info-geo* score radically outperforms BIC as the number of fan-in nodes increase, while presenting a similar performance to BIC in the rest of the topologies. Complete details of these simulation experiments can be found in [37].

8.4 Conclusion

Learning a Bayesian network from data implies two stages: a qualitative stage that deals with the task of learning the dependency among the variables of the network (given by the topology of its directed acyclic graph); and a quantitative stage that determines the conditional probabilities of each variable, given its parents. In this work we describe an information-geometric scoring (*info-geo*) metric that enhances the Bayesian Information Criterion (BIC) score and is based on the Minimum Description Length Principle. While MDL has been applied before as a scoring metric in model searching tasks, the underlying geometric properties of this approach applied to learning the topology of BNs has not been formally established in the literature. In particular, the inclusion of an additional term that gauges the information-geometric properties of the underlying statistical manifold may be instrumental in selecting ‘better’ models from limited sets of data. We conclude that the *info-geo* score is at least as efficient as the BIC score and, under certain circumstances, can drastically improve the accuracy of the model searching process when learning BN topologies from data.

References

1. Friedman, N., Nachman, I., Peer, D.: Learning Bayesian Network Structures from Massive Datasets: The Sparse Candidate Algorithm. In: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI 1999), pp. 206–215 (1999)
2. Neapolitan, R.: Learning Bayesian Networks. Artificial Intelligence. Prentice-Hall, Englewood Cliffs (2003)
3. Cooper, G., Herskovits, E.: A Bayesian Method for the Induction of Probabilistic Networks from Data. Machine Learning 9(4), 309–347 (1992)

4. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E.: Equation of state calculations by fast computing machines. *Journal of Chemical Physics* 21, 1087–1092 (1953)
5. Kirkpatrick, S., Gelatt, D., Vecchi, M.: Optimization by simulated annealing. *Science* 220, 671–680 (1983)
6. Madigan, D., York, J.: Bayesian Graphical Methods for Discrete Data. *International Statistical Review* 63(2) (1995)
7. Friedman, N., Koller, D.: Being Bayesian About Network Structure: A Bayesian Approach to Structure Discovery in Bayesian Networks. In: *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI)* (2000)
8. Friedman, N.: Learning Bayesian networks in the presence of missing values and hidden variables. In: *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence (UAI)* (1997)
9. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm (with discussion). *J.R. Statist. Soc. B* 39, 1–38 (1977)
10. Kass, R.E., Tierney, L., Kadane, J.B.: The validity of posterior asymptotic expansions based on Laplace's method. In: Geisser, S., Hodges, J.S., Press, S.J., Zellner, A. (eds.) *Bayesian and Likelihood Methods in Statistics and Econometrics*. North Holland, New York (1990)
11. Kass, R., Raftery, A.E.: Bayes factors and model uncertainty. *Journal of the American Statistical Association* 90, 773–795 (1995)
12. Schwarz, G.: Estimating the dimension of a model. *Annals of Statistics* 6, 461–464 (1978)
13. Heckermann, D.: A tutorial on learning with Bayesian Networks. In: Jordan, M. (ed.) *Learning in graphical models*. MIT Press, Cambridge (1999)
14. Mitchell, T.: *Machine Learning*. McGraw-Hill, New York (1997)
15. Rissanen, J.: Modeling by the shortest data description. *Automatica J. IFAC* 14, 465–471 (1978)
16. Shannon, C.: A Mathematical Theory of Communication. *The Bell System Technical Journal* 27, 379–423, 623–656 (1948)
17. Vitányi, P., Ming, L.: Minimum Description Length Induction, Bayesianism, and Kolmogorov Complexity. *IEEE Transactions on Information Theory* 46(2) (2000)
18. Solomonoff, R.J.: A formal theory of inductive inference. *Inform. Contr.* pt. 1, 2, 7, 224–254 (1964)
19. Kolmogorov, A.N.: Three approaches to the quantitative definition of information. *Probl. Inform. Transm.* 1(1), 1–7 (1965)
20. Chaitin, G.J.: A theory of program size formally identical to information theory. *J. ACM* 22, 329–340 (1975)
21. Hansen, M., Yu, B.: Model Selection and the Principle of Minimum Description Length. *JASA* 96(454), 746–774 (2001)
22. Rissanen, J.: Stochastic Complexity and Modeling. *Annals of Statistics* 14(3), 1080–1100 (1986)
23. Lipschultz, M.: *Differential Geometry*. Schaum Series. McGraw-Hill, New York (1969)
24. Kreyszig, E.: *Differential Geometry*. Dover Publications (1991)
25. Rodríguez, C.: Entropic priors, Tech. Report, SUNY Albany, Department of Mathematics and Statistics (1991)
26. Amari, S.I.: *Differential Geometrical Methods in Statistics*. Springer, Heidelberg (1985)
27. Amari, S.I., Nagaoka, H.: *Methods of Information Geometry*. Oxford University Press, Oxford (2000)

28. Cartan, E.: Sur la possibilite de plonger un espace riemannian donne un espace Euclidean. *Ann. Soc. Pol. Math.* 6, 1–7 (1927)
29. Janet, M.: Sur la possibilite de plonger un espace riemannian donne das un espace Euclidean. *Ann. Soc. Math. Pol.* 5, 74–85 (1931)
30. Nash, J.: The imbedding problem for Riemannian manifolds. *Annals of Mathematics* 63, 20–63 (1956)
31. Rodríguez, C.: The Metrics Induced by the Kullback Number. In: Skilling, J. (ed.) *Maximum Entropy and Bayesian Methods*, Kluwer, Dordrecht (1989)
32. Jeffreys, H.: *The Theory of Probability*. Oxford University Press, Oxford (1961)
33. Rissanen, J.: Fisher Information and Stochastic Complexity. *IEEE Transaction on Information Theory* 42, 40–47 (1996)
34. Balasubramanian, V.: A Geometric Formulation of Occam’s Razor for Inference of Parametric Distributions. Princeton physics preprint PUPT-1588, Princeton (1996)
35. Rodríguez, C.: Entropic priors for discrete probabilistic networks and for mixtures of Gaussian models. In: *Proceedings of the 21st International Worskhop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, APL Johns Hopkins University, August 4–9 (2001)
36. Rodríguez, C.: The Volume of Bitnets. In: *Proceedings of the 24th International Worskhop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*. AIP Conference Proceedings, Garching, Germany, vol. 735(1), pp. 555–564 (2004)
37. Lauría, E.: Learning the Structure of a Bayesian Network: An Application of Information Geometry and the Minimum Description Length Principle. In: Knuth, K.H., Abbas, A.E., Morris, R.D., Castle, J.P. (eds.) *Proceedings of the 25th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, San José State University, USA, pp. 293–301 (2005)

Causal Graphical Models with Latent Variables: Learning and Inference

Philippe Leray¹, Stijn Meganck², Sam Maes³, and Bernard Manderick²

¹ LINA Computer Science Lab UMR6241, Knowledge and Decision Team,
Université de Nantes, France
`philippe.leray@univ-nantes.fr`

² Computational Modeling Lab, Vrije Universiteit Brussel, Belgium

³ LITIS Computer Science, Information Processing and Systems Lab EA4108,
INSA Rouen, France

9.1 Introduction

This chapter discusses causal graphical models for discrete variables that can handle latent variables without explicitly modeling them quantitatively. In the *uncertainty in artificial intelligence* area there exist several paradigms for such problem domains. Two of them are *semi-Markovian causal models* and *maximal ancestral graphs*. Applying these techniques to a problem domain consists of several steps, typically: structure learning from observational and experimental data, parameter learning, probabilistic inference, and, quantitative causal inference.

We will start this chapter by introducing causal graphical models without latent variables and then move on to models with latent variables.

We will discuss the problem that each of the existing approaches for causal modeling with latent variables only focuses on one or a few of all the steps involved in a generic knowledge discovery approach. The goal of this chapter is to investigate the integral process from observational and experimental data unto different types of efficient inference.

Semi-Markovian causal models (SMCMs) are an approach developed by (Pearl, 2000; Tian and Pearl, 2002a). They are specifically suited for performing quantitative causal inference in the presence of latent variables. However, at this time no efficient parametrisation of such models is provided and there are no techniques for performing efficient probabilistic inference. Furthermore there are no techniques to learn these models from data issued from observations, experiments or both.

Maximal ancestral graphs (MAGs) are an approach developed by (Richardson and Spirtes, 2002). They are specifically suited for structure learning in the presence of latent variables from observational data. However, the techniques only learn up to Markov equivalence and provide no clues on which additional experiments to perform in order to obtain the fully oriented causal graph. See Eberhardt et al. (2005); Meganck et al. (2006) for that type of

results for Bayesian networks without latent variables. Furthermore, as of yet no parametrisation for discrete variables is provided for MAGs and no techniques for probabilistic inference have been developed. There is some work on algorithms for causal inference, but it is restricted to causal inference quantities that are the same for an entire Markov equivalence class of MAGs (Spirtes et al., 2000; Zhang, 2006).

We have chosen to use SMCs as a final representation in our work, because they are the only formalism that allows to perform causal inference while fully taking into account the influence of latent variables. However, we will combine existing techniques to learn MAGs with newly developed methods to provide an integral approach that uses both observational data and experiments in order to learn fully oriented semi-Markovian causal models.

Furthermore, we have developed an alternative representation for the probability distribution represented by a SMC, together with a parametrisation for this representation, where the parameters can be learned from data with classical techniques. Finally, we discuss how probabilistic and quantitative causal inference can be performed in these models with the help of the alternative representation and its associated parametrisation¹.

The next section introduces the simplest causal models and their importance. Then we discuss causal models with latent variables. In section 9.4, we discuss structure learning for those models and in the next section we introduce techniques for learning a SMC with the help of experiments. Then we propose a new representation for SMCs that can easily be parametrised. We also show how both probabilistic and causal inference can be performed with the help of this new representation.

9.2 Importance of Causal Models

We start this section by introducing basic notations necessary for the understanding of the rest of this chapter. Then we will discuss classical probabilistic Bayesian networks followed by causal Bayesian networks. Finally we handle the difference between probabilistic and causal inference, or observation vs. manipulation.

9.2.1 Notations

In this work, uppercase letters are used to represent variables or sets of variables, i.e. $V = \{V_1, \dots, V_n\}$, while corresponding lowercase letters are used to represent their instantiations, i.e. v_1, v_2 and v is an instantiation of all V_i . $P(V_i)$ is used to denote the probability distribution over all possible values of variable V_i , while $P(V_i = v_i)$ is used to denote the probability of the instantiation of variable V_i to value v_i . Usually, $P(v_i)$ is used as an abbreviation of $P(V_i = v_i)$.

¹ By the term parametrisation we understand the definition of a complete set of parameters that describes the joint probability distribution which can be efficiently used in computer implementations of probabilistic inference, causal inference and learning algorithms.

The operators $Pa(V_i)$, $Anc(V_i)$, $Ne(V_i)$ denote the observable parents, ancestors and neighbors respectively of variable V_i in a graph and $Pa(v_i)$ represents the values of the parents of V_i . If $V_i \leftrightarrow V_j$ appears in a graph then we say that they are spouses, i.e. $V_i \in Sp(V_j)$ and vice versa.

When two variables V_i, V_j are independent we denote it by $(V_i \perp\!\!\!\perp V_j)$, when they are dependent by $(V_i \nsim V_j)$.

9.2.2 Probabilistic Bayesian Networks

Here we briefly discuss classical probabilistic Bayesian networks.

See Figure 9.1 for a famous example adopted from Pearl (1988) representing an alarm system. The alarm can be triggered either by a burglary, by an earthquake, or by both. The alarm going of might cause John and/or Mary to call the house owner at his office.

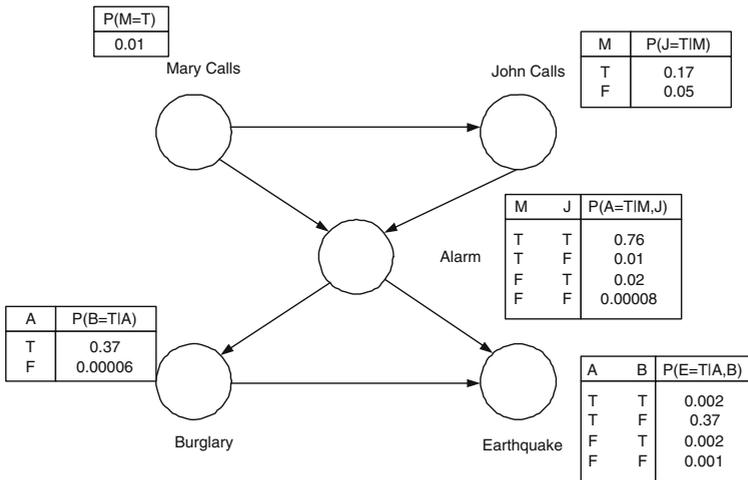


Fig. 9.1. Example of a Bayesian network representing an alarm system

In Pearl (1988); Russell and Norvig (1995) probabilistic Bayesian networks are defined as follows:

Definition 1. A *Bayesian network* is a triple $\langle V, G, P(v_i|Pa(v_i)) \rangle$, with:

- $V = \{V_1, \dots, V_n\}$, a set of observable discrete random variables
- a directed acyclic graph (DAG) G , where each node represents a variable from V
- parameters: conditional probability distributions (CPD) $P(v_i|Pa(v_i))$ of each variable V_i from V conditional on its parents in the graph G .

The CPDs of a BN represent a factorization of the joint probability distribution as a product of conditional probability distributions of each variable given its parents in the graph:

$$P(v) = \prod_{V_i \in V} P(v_i | Pa(v_i)) \quad (9.1)$$

Inference

A BN also allows to efficiently answer probabilistic queries such as

$$P(\text{burglary} = \text{true} | \text{Johncalls} = \text{true}, \text{Marycalls} = \text{false}),$$

in the alarm example of Figure 9.1. It is the probability that there was a burglary, given that we know John called and Mary did not.

Methods have been developed for efficient exact probabilistic inference when the networks are sparse (Pearl, 1988). For networks that are more complex this is not tractable, and approximate inference algorithms have been formulated (Jordan, 1998), such as variational methods (Jordan et al., 1999) and Monte Carlo methods (Mackay, 1999).

Structure Learning

There are two main approaches for learning the structure of a BN from data: *score-based* learning (Heckerman, 1995) and *constraint-based* learning (Spirtes et al., 2000; Pearl, 2000).

For score-based learning, the goal is to find the graph that best matches the data by introducing a scoring function that evaluates each network with respect to the data, and then to search for the best network according to this score.

Constraint-based methods are based on matching the conditional independence relations observed between variables in the data with those entailed by a graph.

However, in general a particular set of data can be represented by more than one BN. Therefore the above techniques have in common that they can only learn upto the *Markov equivalence class*. Such a class contains all the DAGs that correctly represent the data and for performing probabilistic inference any DAG of the class can be chosen.

9.2.3 Causal Bayesian Networks

Now we will introduce a category of Bayesian networks where the edges have a causal meaning.

We have previously seen that in general there is more than one probabilistic BN that can be used to represent the same JPD. More specifically, all the members of a given Markov equivalence class can be used to represent the same JPD.

Opposed to that, in the case of a causal Bayesian network (CBN) we assume that in reality there is a single underlying causal Bayesian network that *generates* the JPD. In Figure 9.2 we see a conceptual sketch: the box represents the real world where a causal Bayesian network generates the data in the form of a

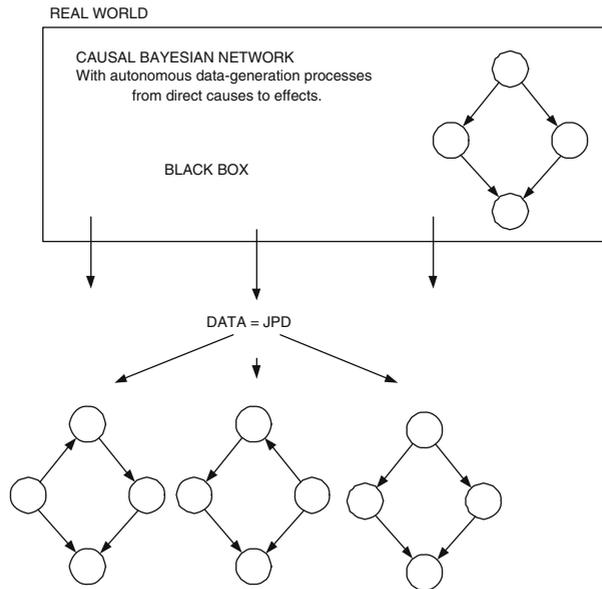


Fig. 9.2. Conceptual sketch of how a CBN generates a JPD, that in its turn can be represented by several probabilistic BNs of which one is a CBN

joint probability distribution. Below we see the BNs that represent all the independence relations present in the JPD. Only one of them is the causal Bayesian network, in this case the rightmost.

The definition of causal Bayesian networks is as follows:

Definition 2. A *causal Bayesian network* is a triple $\langle V, G, P(v_i|Pa(v_i)) \rangle$, with:

- $V = \{V_1, \dots, V_n\}$, a set of observable discrete random variables
- a directed acyclic graph (DAG) G , where each node represents a variable from V
- parameters: conditional probability distributions (CPD) $P(v_i|Pa(v_i))$ of each variable V_i from V conditional on its parents in the graph G .
- Furthermore, the directed edges in G represent an autonomous causal relation between the corresponding variables.

We see that it is exactly the same as Definition 1 for probabilistic Bayesian networks, with the extra addition of the last item.

This is different from a classical BN, where the arrows only represent a probabilistic dependency, and not necessarily a causal one.

Our operational definition of causality is as follows: a relation from variable C to variable E is *causal* in a certain context, when a manipulation in the form of a randomised controlled experiment on variable C , induces a change in the probability distribution of variable E , in that specific context (Neapolitan, 2003).

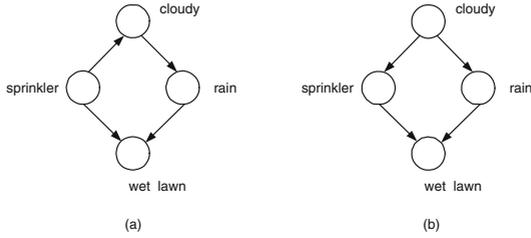


Fig. 9.3. (a) A BN where not all the edges have a causal meaning. (b) A CBN that can represent the same JPD as (a).

This means that in a CBN, each CPD $P(v_i|Pa(v_i))$ represents a stochastic assignment process by which the values of V_i are chosen in response to the values of $Pa(V_i)$ in the underlying domain. This is an approximation of how events are physically related with their effects in the domain that is being modeled. For such an assignment process to be autonomous means that it must stay invariant under variations in the processes governing other variables Pearl (2000).

In the BN of Figure 9.3(a), these assumptions clearly do not hold for all edges and nodes, since in the underlying physical domain, whether or not it is cloudy is not caused by the state of the variable *sprinkler*, i.e. whether or not the sprinkler is on.

Moreover, one could want to manipulate the system, for example by changing the way in which the state of the sprinkler is determined by its causes. More specifically, by changing how the sprinkler reacts to the cloudiness. In order to incorporate the effect of such a manipulation of the system into the model, some of the CPDs have to be changed. However, in a non-causal BN, it is not immediately clear which CPDs have to be changed and exactly how this must be done.

In contrast, in Figure 9.3(b), we see a causal BN that can represent the same JPD as the BN in (a). Here the extra assumptions do hold. For example, if in the system the state of the sprinkler is caused by the cloudiness, and thus the CPD $P(\textit{sprinkler}|\textit{cloudy})$ represents an assignment process that is an approximation of how the sprinkler is physically related to the cloudiness. Moreover, if the sensitivity of the sprinkler is changed, this will only imply a change in the CPD $P(\textit{sprinkler}|\textit{cloudy})$, but not in the processes governing other variables such as $P(\textit{rain}|\textit{cloudy})$.

Note that CBNs are a subclass of BNs and therefore they allow probabilistic inference. In the next section we will discuss what additional type of inference can be performed with them, but first we treat how CBNs can be learned.

Structure Learning

As CBNs are a subset of all BNs, the same techniques as for learning the structure of BNs can be used to learn upto the Markov equivalence class. As mentioned before, for BNs any member of the equivalence can be used.

For CBNs this is not the case, as we look for the orientation of the unique network that can both represent the JPD and the underlying causal influences between the variables. In general, in order to obtain the causal orientation of all the edges, experiments have to be performed, where some variables in the domain are experimentally manipulated and the potential effects on other variables are observed.

Eberhardt et al. (2005) discuss theoretical bounds on the amount of experiments that have to be performed to obtain the full oriented CBN. Meganck et al. (2006) have proposed a solution to learning CBNs from experiments and observations, where the total cost of the experiments is minimised by using elements from decision theory.

Other related approaches include Cooper and Yoo (1999) who derived a Bayesian method for learning from an arbitrary mixture of observational and experimental data.

Tong and Koller (2001) provide an algorithm that actively chooses the experiments to perform based on the model learned so far. In this setting they assume there are a number of query variables Q that can be experimented on and then measure the influence on all other variables $V \setminus Q$. In order to choose the optimal experiment they introduce a loss-function, based on the uncertainty of the direction of an edge, to help indicate which experiment gives the most information. Using the results of their experiments they update the distribution over the possible networks and network parameters. Murphy (2001) introduces a slightly different algorithm of the same approach.

9.2.4 Causal Inference

Here we will briefly introduce causal inference, we start by pointing out the difference with probabilistic inference, and then move on to discuss an important theorem related to causal inference.

Observation vs. Manipulation

An important issue in reasoning under uncertainty is to distinguish between different types of conditioning, each of which modify a given probability distribution in response to information obtained.

Definition 3. Conditioning by observation refers to the way in which a probability distribution of Y should be modified when a modeler passively observes the information $X = x$.

This is represented by conditional probabilities that are defined as follows:

$$P(Y = y|X = x) = P(y|x) = \frac{P(Y = y, X = x)}{P(X = x)}. \quad (9.2)$$

This type of conditioning is referred to as *probabilistic inference*. It is used when the modeler wants to predict the behavior of some variables that have not been

observed, based on the state of some other variables. E.g. will the patients' infection cause him to have a fever ?

This can be very useful in a lot of situations, but in some cases the modeler does not merely want to predict the future behavior of some variables, but has to decide which action to perform, i.e. which variable to manipulate in which way. For example, will administering a dose of 10mg of antibiotics cure the patients' infection ?

In that case probabilistic inference is not the right technique to use, because in general it will return the level of association between the variables instead of the causal influence. In the antibiotics example: if observing the administration of a dose of 10mg of antibiotics returns a high probability of curing the infection, this can be due to (a mix of) several reasons:

- the causal influence of antibiotics on curing the infection,
- the causal influence of curing the infection on antibiotics,
- the causal influence of another variable on both antibiotics and curing the infection, or,
- the causal influence of both antibiotics and curing the infection on another variable that we inadvertently condition on (i.e. selection bias).

Without extra information we cannot make the difference between these reasons. On the other hand if we want to know whether administering a dose of 10mg of antibiotics will cure the patients' infection, we will need to isolate the causal influence of antibiotics on curing the infection and this process is denoted by *causal inference*.

Definition 4. *Causal inference* is the process of calculating the effect of manipulating some variables X on the probability distribution of some other variables Y .

Definition 5. *Conditioning by intervention or manipulation*² refers to the way the distribution Y should be modified if we intervene externally and force the value of X to be equal to x .

To make the distinction clear, Pearl has introduced the **do-operator** (Pearl, 2000)³:

$$P(Y = y|do(X = x)) \tag{9.3}$$

The manipulations we are treating here are surgical in the sense that they only directly change the variable of interest (X in the case of $X = do(x)$).

To reiterate, it is important to realize that conditioning by observation is typically not the way the distribution of Y should be modified if we intervene externally and force the value of X to be equal to x , as can be seen next:

² Throughout this chapter the terms *intervention* and *manipulation* are used interchangeably.

³ In the literature other notations such as $P(Y = y||X = x)$, $P_{X=x}(Y = y)$, or $P(Y = y|X = \hat{x})$ are abundant.

$$P(Y = y|do(X = x)) \neq P(Y = y|X = x) \tag{9.4}$$

and the quantity on the left-hand side cannot be calculated from the joint probability distribution $P(v)$ alone, without additional assumptions imposed on the graph, i.e. that a directed edge represents an autonomous causal relation as in CBNs.

Consider the simple CBNs of Figure 9.4 in the left graph

$$P(y|do(x)) = P(y|x)$$

as X is the only immediate cause of Y , but

$$P(x|do(y)) = P(x) \neq P(x|y)$$

as there is no direct or indirect causal relation going from Y to X . The equalities above are reversed in the graph to the right, i.e. there it holds that $P(y|do(x)) = P(y) \neq P(y|x)$ and $P(x|do(y)) = P(x|y)$.



Fig. 9.4. Two simple causal Bayesian networks

Next we introduce a theorem that specifies how a manipulation modifies the JPD associated with a CBN.

Manipulation Theorem

Performing a manipulation in a domain that is modeled by a CBN, does modify that domain and the JPD that is used to model it. Before introducing a theorem that specifies how a CBN and the JPD that is associated with it must be changed to incorporate the change induced by a manipulation, we will offer an intuitive example.

Example 1. Imagine we want to disable the alarm in the system represented by the CBN of Figure 9.5(a) by performing the manipulation $do(alarm=off)$.

This CBN represents an alarm system against burglars, it can be triggered by a burglary, an earthquake or both. Furthermore, the alarm going off might cause the neighbors to call the owner at his work.

Such a manipulation changes the way in which the value of alarm is being produced in the real world. Originally, the value of alarm was being decided by its immediate causes in the model of Figure 9.5(a): *burglary* and *earthquake*.

After manipulating the alarm by disabling it, *burglary* and *earthquake* are no longer the causes of the alarm, but have been replaced by the manipulation.

In Figure 9.5(b) the graph of the post-manipulation CBN is shown. There we can see that the links between the original causes of *alarm* have been severed and that the value of *alarm* has been instantiated to *off*.

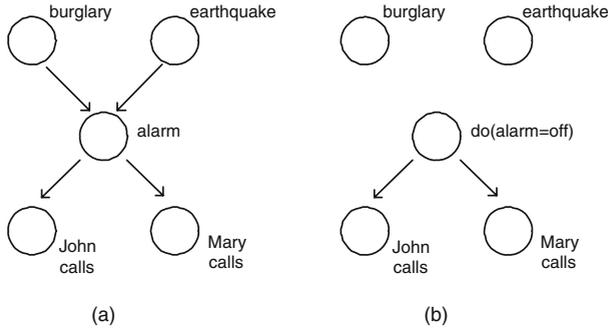


Fig. 9.5. (a) A CBN of an alarm system. (b) The CBN of the alarm system of (a) after disabling the alarm via an external manipulation: $do(alarm=off)$.

To obtain the post-manipulation distribution after fixing a set of variables $M \subseteq V$ to fixed values $M = m$, the factors with the variables in M conditional on their parents in the graph (i.e. their causes in the pre-intervention distribution), have to be removed from the JPD. Formally these are : $P(m_i|Pa(m_i))$ for all variables $M_i \in M$. This is because after the intervention, it is this intervention rather than the parent variables in the graph that cause the values of the variables in M . Furthermore the remaining occurrences of M in the JPD have to be instantiated to $M = m$.

A manipulation of this type only has a local influence in the sense that only the incoming links of a manipulated variable have to be removed from the model, no factors representing other links have to be modified, except for instantiating the occurrences of the manipulated variables M to m . This is a consequence of the assumption of CBNs that the factors of the JPD represent assignment processes that must stay invariant under variations in the processes governing other variables. Formally, we get from (Spirtes et al., 2000):

Theorem 1. *Given a CBN with variables $V = V_1, \dots, V_n$ and we perform the manipulation $M = m$ for a subset of variables $M \subseteq V$, the post-manipulation distribution becomes:*

$$P(v|do(m)) = \prod_{v_i \in V \setminus M} P(v_i|Pa(v_i)) \Bigg|_{M=m} \tag{9.5}$$

Where $|_{M=m}$ stands for instantiating all the occurrences of the variables M to values m in the equation that precedes it.

9.3 Causal Models with Latent Variables

In all the above we made the assumption of *causal sufficiency*, i.e. that for every variable of the domain that is a common cause, observational data can

be obtained in order to learn the structure of the graph and the CPDs. Often this assumption is not realistic, as it is not uncommon that a subset of all the variables in the domain is never observed. We refer to such a variable as a *latent* variable.

We start this section by briefly discussing different approaches to modeling latent variables. After that we introduce two specific models for modeling latent variables and the causal influences between the observed variables. These will be the two main formalisms used in the rest of this chapter so we will discuss their semantics and specifically their differences in a lot of detail.

9.3.1 Modeling Latent Variables

Consider the model in Figure 9.6(a), it is a problem with observable variables V_1, \dots, V_6 and latent variables L_1, L_2 and it is represented by a directed acyclic graph (DAG). As this DAG represents the actual problem henceforth we will refer to it as the **underlying DAG**.

One way to represent such a problem is by using this DAG representation and modeling the latent variables explicitly. Quantities for the observable variables can then be obtained from the data in the usual way. Quantities involving latent variables however will have to be estimated. This involves estimating the cardinality of the latent variables and this whole process can be difficult and lengthy. One of the techniques to learn models in such a way is the structural EM algorithm (Friedman, 1997).

Another method to take into account latent variables in a model is by representing them implicitly. With that approach, no values have to be estimated for the latent variables, instead their influence is absorbed in the distributions of the observable variables. In this methodology, we only keep track of the position of the latent variable in the graph if it would be modeled, without estimating values for it. Both the modeling techniques that we will use in this chapter belong to that approach, they will be described in the next two sections.

9.3.2 Semi-Markovian Causal Models

The central graphical modeling representation that we use are the semi-Markovian causal models. They were first used by Pearl (2000), and Tian and Pearl (2002a) have developed causal inference algorithms for them.

Definitions

Definition 6. A *semi-Markovian causal model (SMCM)* is an acyclic causal graph G with both directed and bi-directed edges. The nodes in the graph represent observable variables $V = \{V_1, \dots, V_n\}$ and the bi-directed edges implicitly represent latent variables $L = \{L_1, \dots, L_{n'}\}$.

See Figure 9.6(b) for an example SMCM representing the underlying DAG in (a).

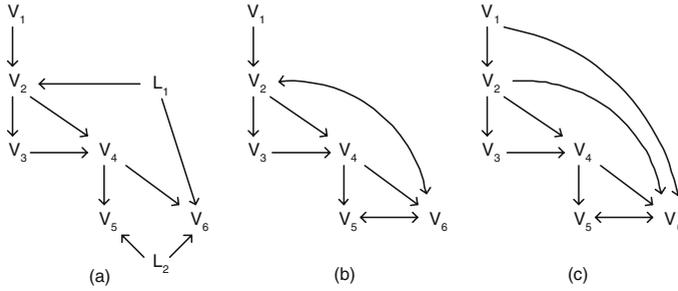


Fig. 9.6. (a) A problem domain represented by a causal DAG model with observable and latent variables. (b) A semi-Markovian causal model representation of (a). (c) A maximal ancestral graph representation of (a).

The fact that a bi-directed edge represents a latent variable, implies that the only latent variables that can be modeled by a SMCM can not have any parents (i.e. is a root node) and has exactly two children that are both observed. This seems very restrictive, however it has been shown that models with arbitrary latent variables can be converted into SMCMs, while preserving the same independence relations between the observable variables (Tian and Pearl, 2002b).

Semantics

In a SMCM, each directed edge represents an immediate autonomous causal relation between the corresponding variables, just as was the case for causal Bayesian networks.

In a SMCM, a bi-directed edge between two variables represents a latent variable that is a common cause of these two variables.

The semantics of both directed and bi-directed edges imply that SMCMs are not maximal, meaning that not all dependencies between variables are represented by an edge between the corresponding variables. This is because in a SMCM an edge either represents an immediate causal relation or a latent common cause, and therefore dependencies due to a so called *inducing path*, will not be represented by an edge.

Definition 7. An *inducing path* is a path in a graph such that each observable non-endpoint node is a collider, and an ancestor of at least one of the endpoints.

Inducing paths have the property that their endpoints can not be separated by conditioning on any subset of the observable variables. For instance, in Figure 9.6(a), the path $V_1 \rightarrow V_2 \leftarrow L_1 \rightarrow V_6$ is inducing.

Parametrisation

SMCMs cannot be parametrised in the same way as classical Bayesian networks (i.e. by the set of CPTs $P(V_i|Pa(V_i))$), since variables that are connected via a bi-directed edge have a latent variable as a parent.

For example in Figure 9.6(b), choosing $P(V_5|V_4)$ as a parameter to be associated with variable V_5 would only lead to erroneous results, as the dependence with variable V_6 via the latent variable L_2 in the underlying DAG is ignored. As mentioned before, using $P(V_5|V_4, L_2)$ as a parametrisation and estimating the cardinality and the values for latent variable L_2 would be a possible solution. However we choose not to do this as we want to leave the latent variables implicit for reasons of efficiency.

In (Tian and Pearl, 2002a), a factorisation of the joint probability distribution over the observable variables of an SMCM was introduced. Later in this chapter we will derive a representation for the probability distribution represented by a SMCM based on that result.

Learning

In the literature no algorithm for learning the structure of an SMCM exists, in this chapter we introduce techniques to perform that task, given some simplifying assumptions, and with the help of experiments.

Probabilistic Inference

Since as of yet no efficient parametrisation for SMCMs is provided in the literature, no algorithm for performing probabilistic inference exists. We will show how existing probabilistic inference algorithms for Bayesian networks can be used together with our parametrisation to perform that task.

Causal Inference

SMCMs are specifically suited for another type of inference, i.e. causal inference. An example causal inference query in the SMCM of Figure 9.6(a) is $P(V_6 = v_6 | do(V_2 = v_2))$.

As seen before, causal inference queries are calculated via the Manipulation Theorem, which specifies how to change a joint probability distribution (JPD) over observable variables in order to obtain the post-manipulation JPD. Informally, it says that when a variable X is manipulated to a fixed value x , the parents of variables X have to be removed by dividing the JPD by $P(X|Pa(X))$, and by instantiating the remaining occurrences of X to the value x .

When all the parents of a manipulated variable are observable, this can always be done. However, in a SMCM some of the parents of a manipulated variable can be latent and then the Manipulation Theorem cannot be directly used to calculate causal inference queries. Some of these causal quantities can be calculated in other ways but some cannot be calculated at all, because the SMCM does not contain enough information.

When a causal query can be unambiguously calculated from a SMCM, we say that it is *identifiable*. More formally:

Definition 8. *The causal effect of variable X on a variable Y is **identifiable** from a SMCM with graph G if $P_{M_1}(y|do(x)) = P_{M_2}(y|do(x))$ for every pair of*

SMCMs M_1 and M_2 with $P_{M_1}(v) = P_{M_2}(v) > 0$ and $G_{M_1} = G_{M_2}$, where P_{M_i} and G_{M_i} respectively denote the probability distribution and graph associated with the SMCM M_i .

In Pearl (2000), Pearl describes the *do-calculus*, a set of inference rules and an algorithm that can be used to perform causal inference. More specifically, the goal of do-calculus is to transform a mathematical expression including manipulated variables related to a SMCM into an equivalent expression involving only standard probabilities of observed quantities. Recent work has shown that do-calculus is complete (Huang and Valtorta, 2006; Shpitser and Pearl, 2006).

Tian and Pearl have introduced theoretical causal inference algorithms to perform causal inference in SMCMs (Pearl, 2000; Tian and Pearl, 2002a). However, these algorithms assume the availability of a subset of all the conditional distributions that can be obtained from the JPD over the observable variables. We will show that with our representation these conditional distributions can be obtained in an efficient way in order to apply this algorithm.

9.3.3 Maximal Ancestral Graphs

Maximal ancestral graphs are another approach to modeling with latent variables developed by Richardson and Spirtes (2002). The main research focus in that area lies on learning the structure of these models and on representing exactly all the independences between the observable variables of the underlying DAG.

Definitions

Ancestral graphs (AGs) are graphs that are complete under marginalisation and conditioning. We will only discuss AGs without conditioning as is commonly done in recent work (Zhang and Spirtes, 2005b; Tian, 2005; Ali et al., 2005).

Definition 9. An *ancestral graph* without conditioning is a graph with no directed cycle containing directed \rightarrow and bi-directed \leftrightarrow edges, such that there is no bi-directed edge between two variables that are connected by a directed path.

Definition 10. An ancestral graph is said to be a *maximal ancestral graph* if, for every pair of non-adjacent nodes V_i, V_j there exists a set Z such that V_i and V_j are d -separated given Z .

A non-maximal AG can be transformed into a unique MAG by adding some bi-directed edges (indicating confounding) to the model. See Figure 9.6(c) for an example MAG representing the same model as the underlying DAG in (a).

Semantics

In this setting a directed edge represents an ancestral relation in the underlying DAG with latent variables. I.e. an edge from variable A to B represents that in the underlying causal DAG with latent variables, there is a directed path between A and B .

Bi-directed edges represent a latent common cause between the variables. However, if there is a latent common cause between two variables A and B , and there is also a directed path between A and B in the underlying DAG, then in the MAG the ancestral relation takes precedence and a directed edge will be found between the variables. $V_2 \rightarrow V_6$ in Figure 9.6(c) is an example of such an edge.

Furthermore, as MAGs are maximal, there will also be edges between variables that have no immediate connection in the underlying DAG, but that are connected via an inducing path. The edge $V_1 \rightarrow V_6$ in Figure 9.6(c) is an example of such an edge.

These semantics of edges make some causal inferences in MAGs impossible. As we have discussed before the Manipulation Theorem states that in order to calculate the causal effect of a variable A on another variable B , the immediate parents (i.e. the old causes) of A have to be removed from the model. However, as opposed to SMCMs, in MAGs an edge does not necessarily represent an immediate causal relationship, but rather an ancestral relationship and hence in general the modeler does not know which are the real immediate causes of a manipulated variable.

An additional problem for finding the original causes of a variable in MAGs is that when there is an ancestral relation and a latent common cause between variables, that the ancestral relation takes precedence and that the confounding is absorbed in the ancestral relation.

Learning

There is a lot of recent research on learning the structure of MAGs from observational data. The Fast Causal Inference (FCI) algorithm (Spirtes et al., 1999), is a constraint based learning algorithm. Together with the rules discussed in Zhang and Spirtes (2005a), the result is a representation of the Markov equivalence class of MAGs. This representative is referred to as a *complete partial ancestral graph* (CPAG) and in Zhang and Spirtes (2005a) it is defined as follows:

Definition 11. Let $[G]$ be the Markov equivalence class for an arbitrary MAG G . The **complete partial ancestral graph** (CPAG) for $[G]$, P_G , is a graph with possibly the following edges $\rightarrow, \leftrightarrow, o-o, o\rightarrow$, such that

1. P_G has the same adjacencies as G (and hence any member of $[G]$) does;
2. A mark of arrowhead ($>$) is in P_G if and only if it is invariant in $[G]$; and
3. A mark of tail ($-$) is in P_G if and only if it is invariant in $[G]$.
4. A mark of (o) is in P_G if not all members in $[G]$ have the same mark.

In the next section we will discuss learning the structure in somewhat more detail.

Parametrisation and Inference

At this time no parametrisation for MAGs with discrete variables exists that represents all the properties of a joint probability distribution, (Richardson and Spirtes, 2002), neither are there algorithms for probabilistic inference.

As mentioned above, due to the semantics of the edges in MAGs, not all causal inferences can be performed. However, there is an algorithm due to Spirtes et al. (2000) and refined by Zhang (2006), for performing causal inference in some restricted cases. More specifically, they consider a causal effect to be identifiable if it can be calculated from all the MAGs in the Markov equivalence class that is represented by the CPAG and that quantity is equal for all those MAGs. This severely restricts the causal inferences that can be made, especially if more than conditional independence relations are taken into account during the learning process, as is the case when experiments can be performed. In the context of this causal inference algorithm, Spirtes et al. (2000) also discuss how to derive a DAG that is a minimal I -map of the probability distribution represented by a MAG.

In this chapter we introduce a similar procedure, but for a single SMCM instead of for an entire equivalence class of MAGs. In that way a larger class of causal inferences can be calculated, as the quantities do not have to be equal in all the models of the equivalence class.

9.4 Structure Learning with Latent Variables

Just as learning a graphical model in general, learning a model with latent variables consists of two parts: structure learning and parameter learning. Both can be done using data, expert knowledge and/or experiments. In this section we discuss structure learning and we differentiate between learning from observational and experimental data.

9.4.1 From Observational Data

In order to learn graphical models with latent variables from observational data a constraint based learning algorithm has been developed by Spirtes et al. (1999). It is called the Fast Causal Inference (FCI) algorithm and it uses conditional independence relations found between observable variables to learn a structure.

Recently this result has been extended with the complete tail augmentation rules introduced in Zhang and Spirtes (2005a). The results of this algorithm is a CPAG, representing the Markov equivalence class of MAGs consistent with the data.

Recent work in the area consists of characterising the equivalence class of CPAGs and finding single-edge operators to create equivalent MAGs (Ali and Richardson, 2002; Zhang and Spirtes, 2005a,b). One of the goals of these advances is to create methods that search in the space of Markov equivalent models (CPAGs) instead of the space of all models (MAGs), mimicking results in the case without latent variables (Chickering, 2002).

As mentioned before for MAGs, in a CPAG the directed edges have to be interpreted as representing ancestral relations instead of immediate causal relations. More precisely, this means that there is a directed edge from V_i to V_j if V_i is an ancestor of V_j in the underlying DAG and there is no subset of observable

variables D such that $(V_i \perp\!\!\!\perp V_j | D)$. This does not necessarily mean that V_i has an immediate causal influence on V_j , it may also be a result of an inducing path between V_i and V_j . For instance in Figure 9.6(c), the link between V_1 and V_6 is present due to the inducing path V_1, V_2, L_1, V_6 shown in Figure 9.6(a).

Inducing paths may also introduce \leftrightarrow , \rightarrow , $o\rightarrow$ or $o-o$ between two variables, although there is no immediate influence in the form of an immediate causal influence or latent common cause between the two variables. An example of such a link is $V_3 o-o V_4$ in Figure 9.7.

A consequence of these properties of MAGs and CPAGs is that they are not very suited for general causal inference, since the immediate causal parents of each observable variable are not available as is necessary according to the manipulation theorem. As we want to learn models that can perform causal inference, we will discuss how to transform a CPAG into a SMCM next.

9.4.2 From Experimental Data

As mentioned above, the result of current state-of-the-art techniques that learn models with implicit latent variables from observational data is a CPAG. This is a representative of the Markov equivalence class of MAGs. Any MAG in that class will be able to represent the same JPD over the observable variables, but not all those MAGs will have all edges with a correct causal orientation.

Furthermore as mentioned in the above, in MAGs the directed edges do not necessarily have an immediate causal meaning as in CBNs or SMCMs, instead they have an ancestral meaning. If it is your goal to perform causal inference, you will need to know the immediate parents to be able to reason about all causal queries. However, edges that are completely oriented but that do not have a causal meaning will not occur in the CPAG, there they will always be of the types $o\rightarrow$ or $o-o$, so orienting them in correct causal way suffices.

Finally, MAGs are maximal, thus every missing edge must represent a conditional independence. In the case that there is an inducing path between two variables and no edge in the underlying DAG, the result of the current learning algorithms will be to add an edge between the variables. Again, although these type of edges give the only correct representation of the conditional independence relations in the domain, they do not represent an immediate causal relation (if the inducing edge is directed) or a real latent common cause (if the inducing edge is bi-directed). Because of this they could interfere with causal inference algorithms, therefore we would like to identify and remove these type of edges.

To recapitulate, the goal of techniques aiming at transforming a CPAG must be twofold:

- finding the correct causal orientation of edges that are not completely specified by the CPAG ($o\rightarrow$ or $o-o$), and,
- removing edges due to inducing paths.

In the next section we discuss how these goals can be obtained by performing experiments.

9.5 From CPAG to SMCM

Our goal is to transform a given CPAG in order to obtain a SMCM that corresponds to the underlying DAG. Remember that in general there are four types of edges in a CPAG: \leftrightarrow , \rightarrow , $o\rightarrow$, $o-o$, in which o means either a tail mark $-$ or a directed mark $>$. As mentioned before, one of the tasks to obtain a valid SMCM is to disambiguate those edges with at least one o as an endpoint. A second task will be to identify and remove the edges that are created due to an inducing path.

In the next section we will introduce some simplifying assumptions we have to use in our work. Then we will discuss exactly which information is obtained from performing an experiment. After that, we will discuss the two possible incomplete edges: $o\rightarrow$ and $o-o$. Finally, we will discuss how we can find edges that are created due to inducing paths and how to remove them to obtain the correct SMCM.

9.5.1 Assumptions

As is customary in the graphical modeling research area, the SMCMs we take into account in this chapter are subject to some simplifying assumptions:

1. *Stability*, i.e. the independencies in the underlying CBN with observed and latent variables that generates the data are structural and not due to several influences exactly cancelling each other out (Pearl, 2000).
2. Only a *single immediate connection* per two variables in the underlying DAG. I.e. we do not take into account problems where two variables that are connected by an immediate causal edge are also confounded by a latent variable causing both variables. Constraint based learning techniques such as IC* (Pearl, 2000) and FCI (Spirtes et al., 2000) also do not explicitly recognise multiple edges between variables. However, Tian and Pearl (2002a) presents an algorithm for performing causal inference where such relations between variables are taken into account.
3. *No selection bias*. Mimicking recent work, we do not take into account latent variables that are conditioned upon, as can be the consequence of selection effects.
4. *Discrete variables*. All the variables in our models are discrete.
5. *Correctness*. The CPAG is correctly learned from data with the FCI algorithm and the extended tail augmentation rules, i.e. each result that is found is not due to a sampling error or insufficient sample size.

9.5.2 Performing Experiments

The experiments discussed here play the role of the manipulations discussed in Section 9.2.3 that define a causal relation. An experiment on a variable V_i , i.e. a randomised controlled experiment, removes the influence of other variables in the system on V_i . The experiment forces a distribution on V_i , and thereby changes the

joint distribution of all variables in the system that depend directly or indirectly on V_i but does not change the conditional distribution of other variables given values of V_i . After the randomisation, the associations of the remaining variables with V_i provide information about which variables V_i influences (Neapolitan, 2003). To perform the actual experiment we have to cut all influence of other variables on V_i . Graphically this corresponds to removing all incoming arrows into V_i from the underlying DAG.

We then measure the influence of the manipulation on variables of interest by obtaining samples from their post-experimental distributions.

More precisely, to analyse the results of an experiment on a variable V_{exp} , we compare for each variable of interest V_j the original observational sample data D_{obs} with the post-experimental sample data D_{exp} . The experiment consists of manipulating the variable V_{exp} to each of its values v_{exp} a sufficient amount of times in order to obtain sample data sets that are large enough to analyse in a statistically sound way. The result of an experiment will be a data set of samples for the variables of interest for each value i of variable $V_{exp} = i$, we will denote such a data set by $D_{exp,i}$.

In order to see whether an experiment on V_{exp} made an influence on another variable V_j , we compare each post-experimental data set $D_{exp,i}$ with the original observational data set D_{obs} (with a statistical test like χ^2). Only if at least one of the data sets is statistically significantly different, we can conclude that variable V_{exp} causally influences variable V_j .

However, this influence does not necessarily have to be immediate between the variables V_{exp} and V_j , but can be mediated by other variables, such as in the underlying DAG: $V_{exp} \rightarrow V_{med} \rightarrow V_j$.

In order to make the difference between a direct influence and a potentially mediated influence via V_{med} , we will no longer compare the complete data sets $D_{exp,i}$ and D_{obs} . Instead, we will divide both data sets in subsets based on the values of V_{med} , or in other words condition on variable V_{med} . Then we compare each of the smaller data sets $D_{exp,i}|v_{med}$ and $D_{obs}|v_{med}$ with each other and this for all values of V_{med} . By conditioning on a potentially mediating variable, we block the causal influence that might go through that variable and we obtain the immediate relation between V_{exp} and V_j .

Note that it might seem that if the mediating variable is a collider, this approach will fail, because conditioning on a collider on a path between two variables creates a dependence between those two variables. However, this approach will still be valid and this is best understood with an example: imagine the underlying DAG is of the form $V_{exp} \cdots \rightarrow V_{med} \leftarrow \cdots V_j$. In this case, when we compare each $D_{exp,i}$ and D_{obs} conditional on V_{med} , we will find no significant difference between both data sets, and this for all the values of V_{med} . This is because the dependence that is created between V_{exp} and V_j by conditioning on the collider V_{med} is present in both the original underlying DAG and in the post-experimental DAG, and thus this is also reflected in the data sets $D_{exp,i}$ and D_{obs} .

Table 9.1. An overview of how to complete edges of type $o \rightarrow$

$Ao \rightarrow B$	Type 1(a)	Type 1(b)	Type 1(c)
Exper. result	$exp(A) \not\rightsquigarrow B$	$exp(A) \rightsquigarrow B$ \nexists p.d. path $A \dashrightarrow B$ (length ≥ 2)	$exp(A) \rightsquigarrow B$ \exists p.d. path $A \dashrightarrow B$ (length ≥ 2)
Orient. result	$A \leftrightarrow B$	$A \rightarrow B$	Block all p.d. paths by conditioning on blocking set Z : $exp(A) Z \rightsquigarrow B: A \rightarrow B$ $exp(A) Z \not\rightsquigarrow B: A \leftrightarrow B$

In order not to overload that what follows with unnecessary complicated notation we will denote performing an experiment at variable V_i or a set of variables W by $exp(V_i)$ or $exp(W)$ respectively, and if we have to condition on some other set of variables Z on the data obtained by performing the experiment, we denote it as $exp(V_i)|Z$ and $exp(W)|Z$.

In general if a variable V_i is experimented on and another variable V_j is affected by this experiment, i.e. has another distribution after the experiment than before, we say that V_j varies with $exp(V_i)$, denoted by $exp(V_i) \rightsquigarrow V_j$. If there is no variation in V_j we note $exp(V_i) \not\rightsquigarrow V_j$.

Before going to the actual solutions we have to introduce the notion of potentially directed paths:

Definition 12. A *potentially directed path* (p.d. path) in a CPAG is a path made only of edges of types $o \rightarrow$ and \rightarrow , with all arrowheads in the same direction. A p.d. path from V_i to V_j is denoted as $V_i \dashrightarrow V_j$.

9.5.3 Solving $o \rightarrow$

An overview of the different rules for solving $o \rightarrow$ is given in Table 9.1.

For any edge $V_i o \rightarrow V_j$, there is no need to perform an experiment at V_j because we know that there can be no immediate influence of V_j on V_i , so we will only perform an experiment on V_i .

If $exp(V_i) \not\rightsquigarrow V_j$, then there is no influence of V_i on V_j so we know that there can be no directed edge between V_i and V_j and thus the only remaining possibility is $V_i \leftrightarrow V_j$ (Type 1(a)).

If $exp(V_i) \rightsquigarrow V_j$, then we know for sure that there is an influence of V_i on V_j , we now need to discover whether this influence is immediate or via some intermediate variables. Therefore we make a difference whether there is a potentially directed (p.d.) path between V_i and V_j of length ≥ 2 , or not. If no such path exists, then the influence has to be immediate and the edge is found $V_i \rightarrow V_j$ (Type 1(b)).

If at least one p.d. path $V_i \dashrightarrow V_j$ exists, we need to block the influence of those paths on V_j while performing the experiment, so we try to find a blocking set Z for all these paths. If $exp(V_i)|Z \rightsquigarrow V_j$, then the influence has to be immediate,

Table 9.2. An overview of how to complete edges of type $o-o$

$Ao-oB$	Type 2(a)	Type 2(b)	Type 2(c)
Exper. result	$exp(A) \not\rightsquigarrow B$	$exp(A) \rightsquigarrow B$ \exists p.d. path $A \dashrightarrow B$ (length ≥ 2)	$exp(A) \rightsquigarrow B$ \exists p.d. path $A \dashrightarrow B$ (length ≥ 2)
Orient. result	$A \leftarrow oB$ (\Rightarrow Type 1)	$A \rightarrow B$	Block all p.d. paths by conditioning on blocking set Z : $exp(A) Z \rightsquigarrow B: A \rightarrow B$ $exp(A) Z \not\rightsquigarrow B: A \leftarrow oB$ (\Rightarrow Type 1)

because all paths of length ≥ 2 are blocked, so $V_i \rightarrow V_j$. On the other hand if $exp(V_i)|Z \not\rightsquigarrow V_j$, there is no immediate influence and the edge is $V_i \leftrightarrow V_j$ (Type 1(c)).

A blocking set Z consists of one variable for each p.d. path. This variable can be chosen arbitrarily as we have explained before that conditioning on a collider does not invalidate our experimental approach.

9.5.4 Solving $o-o$

An overview of the different rules for solving $o-o$ is given in Table 9.2.

For any edge $V_i o-o V_j$, we have no information at all, so we might need to perform experiments on both variables.

If $exp(V_i) \not\rightsquigarrow V_j$, then there is no influence of V_i on V_j so we know that there can be no directed edge between V_i and V_j and thus the edge is of the following form: $V_i \leftarrow oV_j$, which then becomes a problem of Type 1.

If $exp(V_i) \rightsquigarrow V_j$, then we know for sure that there is an influence of V_i on V_j , and like with Type 1(b) we make a difference whether there is a potentially directed path between V_i and V_j of length ≥ 2 , or not. If no such path exists, then the influence has to be immediate and the edge becomes $V_i \rightarrow V_j$.

If at least one p.d. path $V_i \dashrightarrow V_j$ exists, we need to block the influence of those paths on V_j while performing the experiment, so we find a blocking set Z like with Type 1(c). If $exp(V_i)|Z \rightsquigarrow V_j$, then the influence has to be immediate, because all paths of length ≥ 2 are blocked, so $V_i \rightarrow V_j$. On the other hand if $exp(V_i)|Z \not\rightsquigarrow V_j$, there is no immediate influence and the edge is of the following form: $V_i \leftarrow oV_j$, which again becomes a problem of Type 1.

9.5.5 Removing Inducing Path Edges

In the previous phase only o -parts of edges of a CPAG have been oriented. The graph that is obtained in this way can contain both directed and bi-directed edges, each of which can be of two types. For the directed edges:

- an immediate causal edge that is also present in the underlying DAG
- an edge that is due to an inducing path in the underlying DAG.

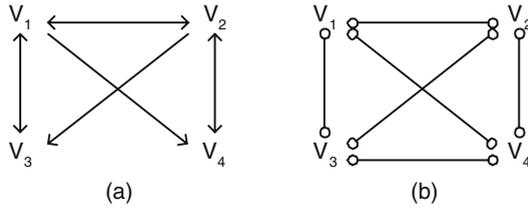


Fig. 9.7. (a) A SMCM. (b) Result of FCI, with an i-false edge $V_3 \circ - o V_4$.

For the bi-directed edges:

- an edge that represents a latent variable in the underlying DAG
- an edge that is due to an inducing path in the underlying DAG.

When representing the same underlying DAG, a SMCM and the graph obtained after orienting all unknown endpoints of the CPAG have the same connections except for edges due to inducing paths in the underlying DAG, these edges are only represented in the experimentally oriented graph.

Definition 13. We will call an edge between two variables V_i and V_j **i-false** if it was created due to an inducing path, i.e. because the two variables are dependent conditional on any subset of observable variables.

For instance in Figure 9.6(a), the path V_1, V_2, L_1, V_6 is an inducing path, which causes the FCI algorithm to find an i-false edge between V_1 and V_6 , see Figure 9.6(c). Another example is given in Figure 9.7 where the SMCM is given in (a) and the result of FCI in (b). The edge between V_3 and V_4 in (b) is a consequence of the inducing path through the observable variables V_3, V_1, V_2, V_4 .

In order to be able to apply a causal inference algorithm we need to remove all i-false edges from the learned structure. The substructures that can indicate this type of edges can be identified by looking at any two variables that a) are connected by an edge, and, b) have at least one inducing path between them.

To check whether the immediate connection needs to be present we have to block all inducing paths by performing one or more experiments on an inducing path blocking set (i-blocking set) Z^{ip} and block all other open paths by conditioning on a blocking set Z . Note that the set of variables Z^{ip} are the set of variables which get an assigned value during the experiments, the set of variables Z are used when looking for independences in the interventional data. If V_i and V_j are dependent, i.e. $(V_i \not\perp V_j)$, under these circumstances then the edge is correct and otherwise it can be removed.

In the example of Figure 9.6(c), we can block the inducing path by performing an experiment on V_2 , and hence can check that V_1 and V_6 do not covary with each other in these circumstances, so the edge can be removed.

An i-blocking set consists of a collider on each of the inducing paths connecting the two variables of interest. Here a blocking set Z is a set of variables that blocks each of the other open paths between the two variables of interest.

Table 9.3 gives an overview of the actions to resolve i-false edges.

Table 9.3. Removing i-false edges

Given	A MAG with a pair of connected variables V_i, V_j , and a set of inducing paths V_i, \dots, V_j
Action	Block all inducing paths V_i, \dots, V_j by performing experiments on i-blocking set Z^{ip} . Block all other open paths between V_i and V_j by conditioning on blocking set Z . When performing all $exp(Z^{ip}) Z$: if $(V_i \not\perp V_j)$: - confounding is real - else remove edge between V_i, V_j

9.5.6 Example

We will demonstrate a number of steps to discover the completely oriented SMCM (Figure 9.6(b)) based on the result of the FCI algorithm applied on observational data generated from the underlying DAG in Figure 9.6(a). The result of the FCI algorithm can be seen in Figure 9.8(a). We will first resolve problems of Type 1 and 2, and then remove i-false edges. The result of each step is explained in Table 9.4 and indicated in Figure 9.8.

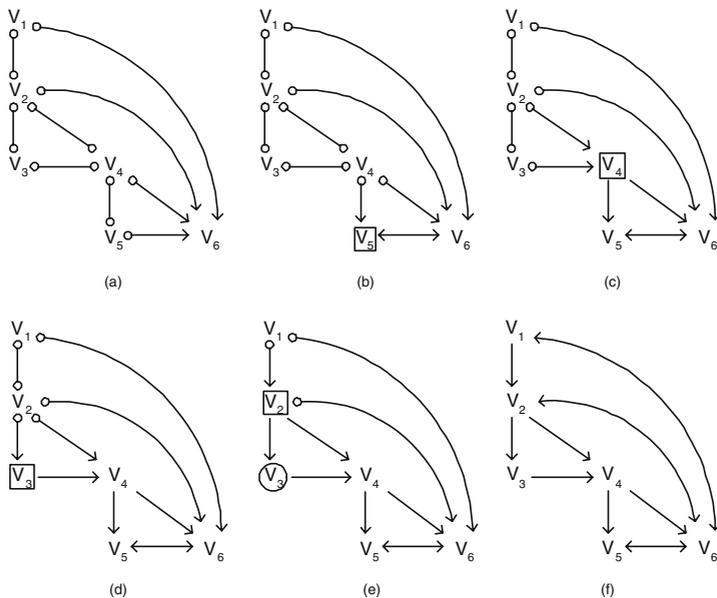


Fig. 9.8. (a) The result of FCI on data of the underlying DAG of Figure 9.6(a). (b) Result of an experiment at V_5 . (c) Result after experiment at V_4 . (d) Result after experiment at V_3 . (e) Result after experiment at V_2 while conditioning on V_3 . (f) Result of resolving all problems of Type 1 and 2.

Table 9.4. Example steps in disambiguating edges by performing experiments

Exper.	Edge before	Experiment result	Edge after	Type
$exp(V_5)$	$V_5 \circ - o V_4$	$exp(V_5) \not\rightsquigarrow V_4$	$V_5 \leftarrow o V_4$	Type 2(a)
	$V_5 \circ \rightarrow V_6$	$exp(V_5) \not\rightsquigarrow V_6$	$V_5 \leftrightarrow V_6$	Type 1(a)
$exp(V_4)$	$V_4 \circ - o V_2$	$exp(V_4) \not\rightsquigarrow V_2$	$V_4 \leftarrow o V_2$	Type 2(a)
	$V_4 \circ - o V_3$	$exp(V_4) \not\rightsquigarrow V_3$	$V_4 \leftarrow o V_3$	Type 2(a)
	$V_4 \circ \rightarrow V_5$	$exp(V_4) \rightsquigarrow V_5$	$V_4 \rightarrow V_5$	Type 1(b)
	$V_4 \circ \rightarrow V_6$	$exp(V_4) \rightsquigarrow V_6$	$V_4 \rightarrow V_6$	Type 1(b)
$exp(V_3)$	$V_3 \circ - o V_2$	$exp(V_3) \not\rightsquigarrow V_2$	$V_3 \leftarrow o V_2$	Type 2(a)
	$V_3 \circ \rightarrow V_4$	$exp(V_3) \rightsquigarrow V_4$	$V_3 \rightarrow V_4$	Type 1(b)
$exp(V_2)$	$V_2 \circ - o V_1$	$exp(V_2) \not\rightsquigarrow V_1$	$V_2 \leftarrow o V_1$	Type 2(a)
	$V_2 \circ \rightarrow V_3$	$exp(V_2) \rightsquigarrow V_3$	$V_2 \rightarrow V_3$	Type 1(b)
	$V_2 \circ \rightarrow V_4$	$exp(V_2) V_3 \rightsquigarrow V_4$	$V_2 \rightarrow V_4$	Type 1(c)

After resolving all problems of Type 1 and 2 we end up with the structure shown in Figure 9.8(f), this representation is no longer consistent with the MAG representation since there are bi-directed edges between two variables on a directed path, i.e. V_2, V_6 . However, this structure is not necessarily a SMCM yet, as there is a potentially i-false edge $V_1 \leftrightarrow V_6$ in the structure with inducing path V_1, V_2, V_6 , so we need to perform an experiment on V_2 , blocking all other paths between V_1 and V_6 (this is also done by $exp(V_2)$ in this case). Given that the original structure is as in Figure 9.6(a), performing $exp(V_2)$ shows that V_1 and V_6 are independent, i.e. $exp(V_2) : (V_1 \perp\!\!\!\perp V_6)$. Thus the bi-directed edge between V_1 and V_6 is removed, giving us the SMCM of Figure 9.6(b).

9.6 Parametrisation of SMCMs

As mentioned before, in his work on causal inference, Tian provides an algorithm for performing causal inference given knowledge of the structure of an SMCM and the joint probability distribution (JPD) over the observable variables. However, a parametrisation to efficiently store the JPD over the observables is not provided.

We start this section by discussing the factorisation for SMCMs introduced in Tian and Pearl (2002a). From that result we derive an additional representation for SMCMs and a parametrisation of that representation that facilitates probabilistic and causal inference. We will also discuss how these parameters can be learned from data.

9.6.1 Factorising with Latent Variables

Consider an underlying DAG with observable variables $V = \{V_1, \dots, V_n\}$ and latent variables $L = \{L_1, \dots, L_{n'}\}$. Then the joint probability distribution can be written as the following mixture of products:

$$P(v) = \sum_{\{l_k | L_k \in L\}} \prod_{V_i \in V} P(v_i | Pa(v_i), LPa(v_i)) \prod_{L_j \in L} P(l_j), \tag{9.6}$$

where $LPa(v_i)$ are the latent parents of variable V_i and $Pa(v_i)$ are the observable parents of V_i .

Remember that in a SMCM the latent variables are implicitly represented by bi-directed edges, then consider the following definition.

Definition 14. *In a SMCM, the set of observable variables can be partitioned into disjoint groups by assigning two variables to the same group iff they are connected by a bi-directed path. We call such a group a **c-component** (from "confounded component") (Tian and Pearl, 2002a).*

E.g. in Figure 9.6(b) variables V_2, V_5, V_6 belong to the same c-component. Then it can be readily seen that c-components and their associated latent variables form respective partitions of the observable and latent variables. Let $Q[S_i]$ denote the contribution of a c-component with observable variables $S_i \subset V$ to the mixture of products in equation 9.6. Then we can rewrite the JPD as follows:

$$P(v) = \prod_{i \in \{1, \dots, k\}} Q[S_i] \tag{9.7}$$

Finally, Tian and Pearl (2002a) proved that each $Q[S]$ could be calculated as follows. Let $V_{o_1} < \dots < V_{o_n}$ be a topological order over V , and let $V^{(i)} = \{V_{o_1}, \dots, V_{o_i}\}$, $i = 1, \dots, n$ and $V^{(0)} = \emptyset$.

$$Q[S] = \prod_{V_i \in S} P(v_i | (T_i \cup Pa(T_i)) \setminus \{V_i\}) \tag{9.8}$$

where T_i is the c-component of the SMCM G reduced to variables $V^{(i)}$, that contains V_i . The SMCM G reduced to a set of variables $V' \subset V$ is the graph obtained by removing all variables $V \setminus V'$ from the graph and the edges that are connected to them.

In the rest of this section we will develop a method for deriving a DAG from a SMCM. We will show that the classical factorisation $\prod P(v_i | Pa(v_i))$ associated with this DAG, is the same as the one that is associated with the SMCM as above.

9.6.2 Parametrised Representation

Here we first introduce an additional representation for SMCMs, then we show how it can be parametrised and finally, we discuss how this new representation could be optimised.

PR-representation

Consider $V_{o_1} < \dots < V_{o_n}$ to be a topological order O over the observable variables V , and let $V^{(i)} = \{V_{o_1}, \dots, V_{o_i}\}$, $i = 1, \dots, n$ and $V^{(0)} = \emptyset$. Then Table 9.5

Table 9.5. Obtaining the parametrised representation from a SMCM

Given a SMCM G and a topological order O , the PR-representation has these properties:
1. The nodes are V , the observable variables of the SMCM. 2. The directed edges that are present in the SMCM are also present in the PR-representation. 3. The bi-directed edges in the SMCM are replaced by a number of directed edges in the following way: Add an edge from node V_i to node V_j iff: a) $V_i \in (T_j \cup Pa(T_j))$, where T_j is the c -component of G reduced to variables $V^{(j)}$ that contains V_j , b) except if there was already an edge between nodes V_i and V_j .

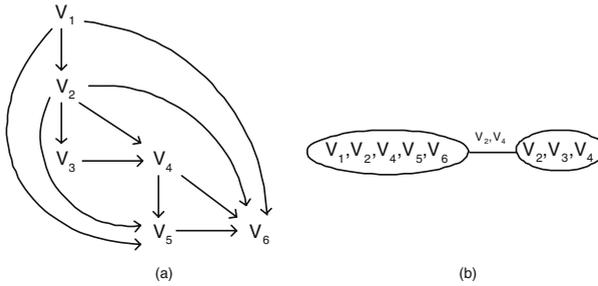


Fig. 9.9. (a) The PR-representation applied to the SMCM of Figure 9.6(b). (b) Junction tree representation of the DAG in (a).

shows how the parametrised (PR-) representation can be obtained from the original SMCM structure.

What happens is that each variable becomes a child of the variables it would condition on in the calculation of the contribution of its c -component as in Equation (9.8).

In Figure 9.9(a), the PR-representation of the SMCM in Figure 9.6(a) can be seen. The topological order that was used here is $V_1 < V_2 < V_3 < V_4 < V_5 < V_6$ and the directed edges that have been added are $V_1 \rightarrow V_5$, $V_2 \rightarrow V_5$, $V_1 \rightarrow V_6$, $V_2 \rightarrow V_6$, and, $V_5 \rightarrow V_6$.

The resulting DAG is an I -map (Pearl, 1988), over the observable variables of the independence model represented by the SMCM. This means that all the independencies that can be derived from the new graph must also be present in the JPD over the observable variables. This property can be more formally stated as the following theorem.

Theorem 2. *The PR-representation PR derived from a SMCM S is an I-map of that SMCM.*

Proof. Proving that PR is an I -map of S amounts to proving that all independences represented in PR (A) imply an independence in S (B), or $A \Rightarrow B$. We will prove that assuming both A and $\neg B$ leads to a contradiction.

Assumption $\neg B$: consider that two observable variables X and Y are dependent in the SMCM S conditional on some (possible empty) set of observable variables Z : $X \not\perp_S Y | Z$.

Assumption A : consider that X and Y are independent in PR conditional on Z : $X \perp_{PR} Y | Z$.

Then based on $X \not\perp_S Y | Z$ we can discriminate two general cases:

1. \exists a path C in S connecting variables X and Y that contains no colliders and no elements of Z .
2. \exists a path C in S connecting variables X and Y that contains at least one collider Z_i that is an element of Z . For the collider there are three possibilities:
 - a) $X \dots C_i \rightarrow Z_i \leftarrow C_j \dots Y$
 - b) $X \dots C_i \leftrightarrow Z_i \leftarrow C_j \dots Y$
 - c) $X \dots C_i \leftrightarrow Z_i \leftrightarrow C_j \dots Y$

Now we will show that each case implies $\neg A$:

1. Transforming S into PR only adds edges and transforms double-headed edges into single headed edges, hence the path C is still present in S and it still contains no collider. This implies that $X \perp_{PR} Y | Z$ is false.
2. a) The path C is still present in S together with the collider in Z_i , as it has single headed incoming edges. This implies that $X \perp_{PR} Y | Z$ is false.
 - b) The path C is still present in S . However, the double-headed edge is transformed into a single headed edge. Depending on the topological order there are two possibilities:
 - $C_i \rightarrow Z_i \leftarrow C_j$: in this case the collider is still present in PR , this implies that $X \not\perp_{PR} Y | Z$
 - $C_i \leftarrow Z_i \leftarrow C_j$: in this case the collider is no longer present, but in PR there is the new edge $C_i \leftarrow C_j$ and hence $X \not\perp_{PR} Y | Z$
 - c) The path C is still present in S . However, both double-headed edges are transformed into single headed edges. Depending on the topological order there are several possibilities. For the sake of brevity we will only treat a single order here, for the others it can easily be checked that the same holds.

If the order is $C_i < Z_i < C_j$, the graph becomes $C_i \rightarrow Z_i \rightarrow C_j$, but there are also edges from C_i and Z_i to C_j and its parents $Pa(C_j)$. Thus the collider is no longer present, but the extra edges ensure that $X \not\perp_{PR} Y | Z$.

This implies that $X \perp_{PR} Y | Z$ is false and therefore we can conclude that PR is always an I -map of S under our assumptions. □

Parametrisation

For this DAG we can use the same parametrisation as for classical BNs, i.e. learning $P(v_i | Pa(v_i))$ for each variable, where $Pa(v_i)$ denotes the parents in the

new DAG. In this way the JPD over the observable variables factorises as in a classical BN, i.e. $P(v) = \prod P(v_i|Pa(v_i))$. This follows immediately from the definition of a c -component and from Equation (9.8).

Optimising the Parametrisation

Remark that the number of edges added during the creation of the PR-representation depends on the topological order of the SMCN.

As this order is not unique, giving precedence to variables with a lesser amount of parents, will cause less edges to be added to the DAG. This is because added edges go from parents of c -component members to c -component members that are topological descendants.

By choosing an optimal topological order, we can conserve more conditional independence relations of the SMCN and thus make the graph more sparse, leading to a more efficient parametrisation.

Note that the choice of the topological order does not influence the correctness of the representation, Theorem 2 shows that it will always be an I -map.

Learning Parameters

As the PR-representation of SMCNs is a DAG as in the classical Bayesian network formalism, the parameters that have to be learned are $P(v_i|Pa(v_i))$. Therefore, techniques such as ML and MAP estimation (Heckerman, 1995) can be applied to perform this task.

9.6.3 Probabilistic Inference

Two of the most famous existing probabilistic inference algorithms for models without latent variables are the $\lambda - \pi$ algorithm (Pearl, 1988) for tree-structured BNs, and the *junction tree* algorithm (Lauritzen and Spiegelhalter, 1988) for arbitrary BNs.

These techniques cannot immediately be applied to SMCNs for two reasons. First of all until now no efficient parametrisation for this type of models was available, and secondly, it is not clear how to handle the bi-directed edges that are present in SMCNs.

We have solved this problem by first transforming the SMCN to its PR-representation which allows us to apply the junction tree (JT) inference algorithm. This is a consequence of the fact that, as previously mentioned, the PR-representation is an I -map over the observable variables. And as the JT algorithm only uses independencies in the DAG, applying it to an I -map of the problem gives correct results. See Figure 9.9(b) for the junction tree obtained from the parametrised representation in Figure 9.9(a).

Note that any other classical probabilistic inference technique that only uses conditional independencies between variables could also be applied to the PR-representation.

9.6.4 Causal Inference

In Tian and Pearl (2002a), an algorithm for performing causal inference was developed, however as mentioned before they have not provided an efficient parametrisation.

In Spirtes et al. (2000); Zhang (2006), a procedure is discussed that can identify a limited amount of causal inference queries. More precisely only those whose result is equal for all the members of a Markov equivalence class represented by a CPAG.

In Richardson and Spirtes (2003), causal inference in AGs is shown on an example, but a detailed approach is not provided and the problem of what to do when some of the parents of a variable are latent is not solved.

By definition in the PR-representation, the parents of each variable are exactly those variables that have to be conditioned on in order to obtain the factor of that variable in the calculation of the c -component, see Table 9.5 and Tian and Pearl (2002a). Thus, if we want to apply Tian's causal inference algorithm, the PR-representation provides all the necessary quantitative information, while the original structure of the SMCM provides the necessary structural information.

9.7 Conclusions and Perspectives

In this chapter we have introduced techniques for causal graphical modeling with latent variables. We have discussed all classical steps in a modeling process such as learning the structure from observational and experimental data, model parametrisation, probabilistic and causal inference.

More precisely we showed that there is a big gap between the models that can be learned from data alone and the models that are used in causal inference theory. We showed that it is important to retrieve the fully oriented structure of a SMCM, and discussed how to obtain this from a given CPAG by performing experiments.

As the experimental learning approach relies on randomized controlled experiments, in general it is not scalable to problems with a large number of variables, due to the associated large number of experiments. Furthermore, it cannot be applied in application areas where such experiments are not feasible due to practical or ethical reasons.

For future work we would like to relax the assumptions made in this chapter. First of all we want to study the implications of allowing two types of edges between two variables, i.e. confounding as well as a immediate causal relationship. Another direction for possible future work would be to study the effect of allowing multiple joint experiments in other cases than when removing inducing path edges.

Furthermore, we believe that applying the orientation and tail augmentation rules of Zhang and Spirtes (2005a) after each experiment, might help to reduce the number of experiments needed to fully orient the structure. In this way we could extend our previous results (Meganck et al., 2006) on minimising the total number of experiments in causal models without latent variables, to SMCMs. This allows to compare practical results with the theoretical bounds developed in Eberhardt et al. (2005).

SMCMs have not been parametrised in another way than by the entire joint probability distribution, we showed that using an alternative representation, we can parametrise SMCs in order to perform probabilistic as well as causal inference. Furthermore this new representation allows to learn the parameters using classical methods.

We have informally pointed out that the choice of a topological order, when creating the PR-representation, influences the size and thus the efficiency of the PR-representation. We would like to investigate this property in a more formal manner. Finally, we have started implementing the techniques introduced in this chapter into the structure learning package (SLP)⁴ of the Bayesian networks toolbox (BNT)⁵ for MATLAB.

Acknowledgements

This work was partially funded by a IWT-scholarship. This work was partially supported by the IST Programme of the European Community, under the PASCAL network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

References

- Ali, A., Richardson, T.: Markov equivalence classes for maximal ancestral graphs. In: Proc. of the 18th Conference on Uncertainty in Artificial Intelligence (UAI), pp. 1–9 (2002)
- Ali, A.R., Richardson, T., Spirtes, P., Zhang, J.: Orientation rules for constructing markov equivalence classes of maximal ancestral graphs. Technical Report 476, Dept. of Statistics, University of Washington (2005)
- Chickering, D.: Learning equivalence classes of Bayesian-network structures. *Journal of Machine Learning Research* 2, 445–498 (2002)
- Cooper, G.F., Yoo, C.: Causal discovery from a mixture of experimental and observational data. In: *Proceedings of Uncertainty in Artificial Intelligence*, pp. 116–125 (1999)
- Eberhardt, F., Glymour, C., Scheines, R.: On the number of experiments sufficient and in the worst case necessary to identify all causal relations among n variables. In: *Proc. of the 21st Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 178–183 (2005)
- Friedman, N.: Learning belief networks in the presence of missing values and hidden variables. In: *Proc. of the 14th International Conference on Machine Learning*, pp. 125–133 (1997)
- Heckerman, D.: A tutorial on learning with bayesian networks. Technical report, Microsoft Research (1995)
- Huang, Y., Valtorta, M.: Pearl's calculus of intervention is complete. In: *Proc. of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 217–224 (2006)
- Jordan, M.I. (ed.): *Learning in Graphical Models*. MIT Press, Cambridge (1998)

⁴ <http://banquiseasi.insa-rouen.fr/projects/bnt-slp/>

⁵ <http://www.cs.ubc.ca/~murphyk/Software/BNT/bnt.html>

- Jordan, M.I., Ghahramani, Z., Jaakkola, T., Saul, L.K.: An introduction to variational methods for graphical models. *Machine Learning* 37(2), 183–233 (1999)
- Lauritzen, S.L., Spiegelhalter, D.J.: Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, series B* 50, 157–244 (1988)
- Mackay, D.: Introduction to monte carlo methods. In: Jordan, M.I. (ed.) *Learning in Graphical Models*, pp. 175–204. MIT Press, Cambridge (1999)
- Meganck, S., Leray, P., Manderick, B.: Learning causal bayesian networks from observations and experiments: A decision theoretic approach. In: *Modeling Decisions in Artificial Intelligence. LNCS*, pp. 58–69 (2006)
- Murphy, K.P.: Active learning of causal bayes net structure. Technical report, Department of Computer Science, UC Berkeley (2001)
- Neapolitan, R.: *Learning Bayesian Networks*. Prentice Hall, Englewood Cliffs (2003)
- Pearl, J.: *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Francisco (1988)
- Pearl, J.: *Causality: Models, Reasoning and Inference*. MIT Press, Cambridge (2000)
- Richardson, T., Spirtes, P.: Ancestral graph markov models. Technical Report 375, Dept. of Statistics, University of Washington (2002)
- Richardson, T., Spirtes, P.: Causal inference via Ancestral graph models. In: *Highly Structured Stochastic Systems. Oxford Statistical Science Series*, ch. 3. Oxford University Press, Oxford (2003)
- Russell, S.J., Norvig, P. (eds.): *Artificial Intelligence: A Modern Approach*. Prentice Hall, Englewood Cliffs (1995)
- Shpitser, I., Pearl, J.: Identification of conditional interventional distributions. In: *Proc. of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 437–444 (2006)
- Spirtes, P., Glymour, C., Scheines, R.: *Causation, Prediction and Search*. MIT Press, Cambridge (2000)
- Spirtes, P., Meek, C., Richardson, T.: An algorithm for causal inference in the presence of latent variables and selection bias. In: *Computation, Causation, and Discovery*, pp. 211–252. AAAI Press, Menlo Park (1999)
- Tian, J.: Generating markov equivalent maximal ancestral graphs by single edge replacement. In: *Proc. of the 21st Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 591–598 (2005)
- Tian, J., Pearl, J.: On the identification of causal effects. Technical Report (R-290-L), UCLA C.S. Lab (2002a)
- Tian, J., Pearl, J.: On the testable implications of causal models with hidden variables. In: *Proc. of the 18th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 519–527 (2002b)
- Tong, S., Koller, D.: Active learning for structure in bayesian networks. In: *Seventeenth International Joint Conference on Artificial Intelligence* (2001)
- Zhang, J.: *Causal Inference and Reasoning in Causally Insufficient Systems*. PhD thesis, Carnegie Mellon University (2006)
- Zhang, J., Spirtes, P.: A characterization of markov equivalence classes for ancestral graphical models. Technical Report 168, Dept. of Philosophy, Carnegie-Mellon University (2005a)
- Zhang, J., Spirtes, P.: A transformational characterization of markov equivalence for directed acyclic graphs with latent variables. In: *Proc. of the 21st Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 667–674 (2005b)

Use of *Explanation Trees* to Describe the State Space of a Probabilistic-Based Abduction Problem^{*}

M. Julia Flores¹, José A. Gámez¹, and Serafín Moral²

¹ Computing Systems Department & SIMD - *i³A*
University of Castilla-La Mancha
Campus Universitario s/n. Albacete. 02071
{julia,jgamez}@dsi.uclm.es

² Departamento de Ciencias de la Computación e I. A.
Universidad de Granada
Granada, 18071, Spain
smc@decsai.ugr.es

Abstract. This chapter presents a new approach to the problem of obtaining the most probable explanations given a set of observations in a Bayesian network. The method provides a set of possibilities ranked by their probabilities. The main novelties are that the level of detail of each one of the explanations is not uniform (with the idea of being as simple as possible in each case), the explanations are mutually exclusive, and the number of required explanations is not fixed (it depends on the particular case we are solving). Our goals are achieved by means of the construction of the so called *explanation tree* which can have asymmetric branching and that will determine the different possibilities. This chapter describes the procedure for its computation based on information theory criteria and shows its behaviour in some examples.

To test the procedure we have used a couple of examples that can be intuitively interpreted and understood. Moreover, we have carried out a set of experiments to make a comparison with other existing abductive techniques that were designed with goals similar to those we pursue.

10.1 Abductive Inference in Bayesian Networks. Total and Partial Abduction Techniques

When dealing with inference in Bayesian networks, the most typical and classical approach is to compute the posterior probability distributions of the variables when some evidence has been observed. This is indeed the most common probabilistic inference task in Bayesian networks (BNs) for which normally probability or evidence propagation [1, 2, 3] is used. In this case, the computation of the posterior probability for each non-observed variables given a set of observations ($X_O = x_O$) (the *evidence*) is the main goal:

$$P(X_i | X_O = x_O) \quad \forall X_i \in X_I \quad (10.1)$$

^{*} This chapter is an extended version of [27].

This is normally known as *evidence propagation* or *probability propagation*. We call evidence those facts which are observed and therefore the involved variables are fixed to a certain value each one.

But this is not the only existing problem we can solve with BNs computation, and there are also other interesting inference tasks. In this category we can find abductive reasoning, which represents another type of propagation and has also a great relevance within this BNs field. Abduction is a method for inference used for finding/generating explanations to some observed facts. Generating explanations in Bayesian networks can be understood in two (main) different ways:

1. *Explaining the reasoning process* (see [4] for a review). That is, trying to justify *how* a conclusion was obtained, *why* new information was asked, etc.
2. *Diagnostic explanations* or *abductive inference* (see [5] for a review). In this case the explanation reduces to *factual* information about the state of the world, and the best explanation for a given evidence is the state of the world (configuration) that is the most probable given the evidence [1].

In this case we will focus on the second approach. Therefore, given a set of observations or evidence ($X_O = x_O$ or x_O in short) known as the *explanandum*, we aim to obtain the best configuration of values for the explanatory variables (the *explanation*) which is consistent with the *explanandum* and which could be assumed to predict it.

As the abductive task is mostly considered as the search of explanations its major application has been done for diagnosis and analysis problems [6, 7], where medical diagnosis stands out especially. In the last years various authors have directed their research endeavours to the study of performing abductive inference for the formalism of Bayesian networks. Two main abductive tasks in BNs are identified:

- *Most Probable Explanation* (MPE) or *total abduction*. In this case all the unobserved variables (X_U) are included in the explanation [1]. The *best* explanation is the assignment $X_U = x_U^*$ which has maximum a posteriori probability given the *explanandum*, i.e.,

$$x_U^* = \arg \max_{x_U \in \Omega_{X_U}} P(x_U | x_O). \quad (10.2)$$

Searching for the best explanation has the same complexity (NP-hard [8]) as probability propagation, in fact the best MPE can be obtained by using probability propagation algorithms but replacing summation by maximum in the marginalisation operator [9]. However, as it is expected to have several competing hypothesis accounting for the *explanandum*, our goal usually is to get the K best MPEs. Nilsson [10] showed that using algorithm in [9] only the first three MPEs can be correctly identified, and proposed a clever method to identify the remaining $(4, \dots, K)$ explanations. This kind of problems has been studied by several authors who developed exact algorithms [11, 12] and there are also works using approximate methods [13] in order to solve them.

Nilsson proved that under certain assumptions, his algorithm to obtain K -MPEs has a similar complexity to the computation of posterior probabilities by Hugin algorithm [10, 14]

One of the main drawbacks of the MPE definition is that as it produces complete assignments, the explanations obtained can exhibit the *overspecification* problem [15] because some non-relevant variables have been used as explanatory.

- *Maximum a Posteriori Assignment* (MAP) or *partial abduction* [16, 15]. The goal of this task is to alleviate the overspecification problem by considering as target variables only a subset of the unobserved variables called the *explanation set* (X_E). Sometimes certain variables clearly have no explanatory value. This could be the case in a network that represents a car, where for the observation $x_O = \{\text{"car does not start"}\}$ one configuration such as "radio does not work" is not explanatory, since the status of the radio will not probably be an answer by itself to the observed fault. Others will be just intermediate nodes, let us imagine a network modelling the final mark of a student. We could have an intermediate node, again not explanatory by itself, which is a combination of (=whose parents are) the really observable variables that express *Theory* and *Practice* marks. In general there could be variables useless in an explanation because they simply not give any information from an abductive point of view or we are just not interested in them.

Then, we look for the maximum a posteriori assignment only for those variables given the explanandum, i.e.,

$$x_E^* = \arg \max_{x_E} P(x_E | x_O) = \arg \max_{x_E} \sum_{x_R} P(x_E, x_R | x_O), \quad (10.3)$$

where $X_R = X_U \setminus X_E$.

In principle it could seem that the solution for this problem would arise by means of an adaptation of those methods developed for total abduction. However, a deeper study of the problem reveals that in general this does not hold, and moreover those resolution methods for partial abduction will be generally less efficient than total abduction. Therefore, algorithms are almost always approximate (based on search methods). This problem is therefore more complex than the MPE problem, because it can be NP-hard even for cases in which MPE is polynomial (e.g., polytrees) [17, 18], although recently Park and Darwiche [19, 17] have proposed exact and approximate algorithms to enlarge the class of *efficiently* solved cases. With respect to looking for the K best explanations, exact and approximate algorithms which combine Nilsson algorithm [10] with probability trees [20] have been proposed in [21]. In general, we could say that even being a more interesting 'problem partial abduction has been less studied than total abduction [22].

The problem in both cases is set out as the search of the instantiation values for all (total) or a subset of (partial) non-observed variables in such a way that the joint probability of the configuration is maximum. Besides, since it is

not generally enough by finding the best explanation, but we need the K best explanations, this K number is usually added to the problem statement.

The question now is which variables should be included in the explanation set. Many algorithms avoid this problem by assuming that the explanation set is provided as an input, e.g., given by the experts or users. Many others interpret the BN as a causal one and only ancestors of the *explanandum* are allowed to be included in the explanation set (sometimes only root nodes are considered) [11]. However, including all the ancestors in the explanation set does not seem to avoid the overspecification problem and even so, what happens if the network does not have a causal interpretation?, e.g., it has been learnt from a data base or it represents an agent's beliefs [23]. Shimony [15, 24] goes one step further and describes a method which tries to identify the relevant variables (among the *explanandum* ancestors) by using independence and relevance based criteria. However, as pointed out in [23] the explanation set identified by Shimony's method is not as concise as expected, because for each variable in the *explanandum* all the variables in at least one path from it to a root variable are included in the explanation set. Henrion and Druzdzel [25] proposed a model called *scenario-based explanation*. In this model a tree of propositions is assumed, where a path from the root to a leaf represents a scenario, and they look for the scenario with highest probability. In this model, partial explanations are allowed, but they are restricted to come from a set of predefined explanations.

As stated in [23] *conciseness* is a desirable feature in an explanation, that is, the user usually wants to know only the most influential elements of the complete explanation, and does not want to be burdened with unnecessary detail. This follows the logical principle known as *Occam's razor*¹ which can be stated as *one should not increase, beyond what is necessary, the number of entities required to explain anything*. This criterium for deciding among scientific theories or explanations is also said to be parsimonious. Because of this conception of choosing the simplest explanation for a phenomenon, a different approach is taken in [26]. The idea is that even when only the relevant variables to the *explanandum* are included in the explanation set, the explanations can be simplified due to context-specific irrelevance. This idea is even more interesting when we look for the K MPEs, because it allows us to obtain explanations with different number of literals. In [26] the process is divided into two stages: (1) the K MPEs are obtained for a given prespecified explanation set, and (2) then they are simplified by using different independence and relevance based criteria.

In our work we try to obtain simplified explanations directly. The reason is that the second stage in [26] requires to carry out several probabilistic propagations and so its computational cost is high (and notice that this process is carried out after a complex- MAP computation). Another drawback of the procedure in [26] is that it is possible, that after simplification, the explanations are not mutually exclusive, we can have even the case of two explanations such that one is a subset of the other. Here, our basic idea is to start with a predefined explanation set X_E , and then we build a tree in which variables (from X_E)

¹ Attributed to the medieval philosopher William of Occam (or Ockham).

are added in function of their explanatory power with respect to the *explanandum* but taken into account the current context, that is, the partial assignment represented by the path obtained from the root to the node currently analysed. Variables are selected based on the idea of *stability*, that is, we can suppose that our system is (more or less) stable, and that it becomes unstable when some (*unexpected*) observations are entered into the system. The instability of a variable will be measured by its entropy or by means of its (im)purity (GINI index). Therefore, we first select those variables that reduce most the uncertainty of the non-observed variables of the explanation set, i.e., the variables better determining the value of the explanation variables. Of course, the tree does not have to be symmetric and we can decide to stop the growing of a branch even if not all the variables in X_E have been included. In any case, our set of explanations will be mutually exclusive, and will have the additional property of being exhaustive, i.e., we will construct a full partition of the set of possible configurations or scenarios of the values of the variables in the explanation set. We will see how this complete partition of the space can lead to some configurations that should not be understood as real explanations, on the contrary they may be just regarded as configurations compatible with the evidence.

Along the subsequent sections we will describe our method in detail (section 10.2) and afterwards illustrate it: first (section 10.3), by using some (toy) study cases. These two first points will be a slightly extended version of the published paper [27]. Section 10.4 will do a further analysis on the obtained trees and another example (used in [26]) is added to our experiments in order to compare both techniques.

10.2 On the Search for Minimal Explanations: The *Explanation Tree*

Our method aims to find the best explanation(s) for the observed variables that do not necessarily have a fixed number of literals and we want to achieve that **directly**. The provided explanations will adapt to the current circumstances. Sometimes the fact that a variable X takes a particular value is an explanation by itself (Occam's razor) and including other variables to this explanation will not add any new significant information.

We have then decided to represent our solutions by a tree, the *Explanation Tree* (ET). In the ET, every inner node will denote a variable of the explanation set and every branch from this variable will indicate the instantiation of this variable to one of its possible states. Each node of the tree will determine an assignment for the variables in the path from the root to it: each variable is equal to the value on the edge followed by the path. This assignment will be called the *configuration* of values associated to the node. In the explanation tree, we will store for each leaf the probability of its associated configuration given the evidence. The set of explanations will be the set of configurations associated to the leaves of the explanation tree ordered by their posterior probability given the evidence.

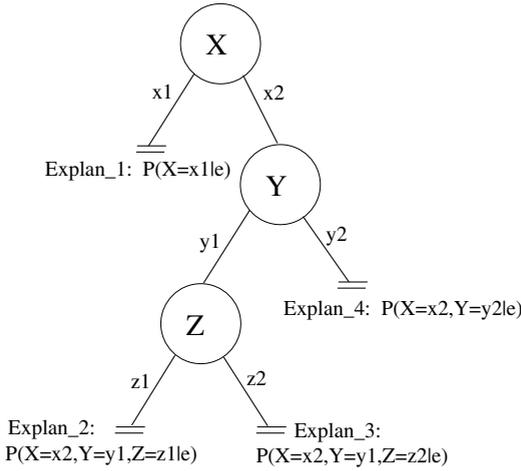


Fig. 10.1. An example of explanation tree

For example, in Fig. 10.1 we can see three variables X, Y and Z that belong to the explanation set, since they are nodes in the tree. In this particular example there are four leaf nodes, that is, four possible explanations, each of them labelled as Explan_i . What this ET indicates is that, given the observed evidence, X has the value $x1$ is a valid explanation for such situation (with its probability/weight associated). But if it is not the case then we should look into other factors, in this case Y . For example, we can see that adding $Y = y2$ to the explanation will be enough. Otherwise, when Y takes value $y1$ the node needs to be *expanded* and we have to look for other involved factors in order to find a valid explanation (in this example, variable Z).

Although the underlying idea is simple, how to obtain this tree is not so evident. There are two major points that have to be answered:

- As the ET is created in a top-down way, given a branch of the tree, how to select the next variable?
- Given our goals, i.e. allow asymmetry and get concise explanations, how to decide when to stop branching?

To solve the two previous questions we have used information measures. For the first one, we look for the variable that once instantiated the uncertainty of the rest explanation variables is reduced at maximum. In other words, given the context provided by the current branch, we identify the most explicative as the one that helps to determine the values of the other variables as much as possible.

Algorithm 1 (`CREATE-NEW-NODE`) recursively creates our ET. In this algorithm we assume the existence of an inference engine that provides us with the probabilities needed during tree growing. We comment on such engine in Section 10.2.1. The algorithm is called with the following parameters:

1. The evidence/observations to be explained x_O .
2. The *path* corresponding to the branch we are growing. In the first call to this algorithm, i.e. when deciding the root node, this parameter will be null.
3. The current explanation set (X_E). That is, the set of explanatory variables already available given the context (path). In the first call X_E is the original explanation set. Notice also that if $X_E = X_U$ in the first call, i.e., all non-observed variables belong to the explanation set, then the method has to select those variables relevant to the explanation without prior information.
4. Two real numbers α and β used as thresholds (on information and probability respectively) to stop growing.
5. The final explanation tree that will be recursively and incrementally constructed as an accumulation of branches (paths). Empty in the initial call.

Algorithm 1. Creates a new node for the explanation tree

```

1: procedure CREATE_NEW_NODE( $x_O, path, X_E, \alpha, \beta, ET$ )
2:   for all  $X_j, X_k \in X_E$  do
3:      $Info[X_j, X_k] = Inf(X_j, X_k | x_O, path)$ 
4:   end for
5:    $X_j^* = \arg \max_{X_j \in X_E} \sum_{X_k} Info[X_j, X_k]$ 
6:   if CONTINUE( $Info[], X_j^*, \alpha$ ) and  $P(path | x_O) > \beta$  then
7:     for all state  $x_j$  of  $X_j^*$  do
8:        $new\_path \leftarrow path + X_j^* = x_j$ 
9:       CREATE_NEW_NODE( $x_O, new\_path, X_E \setminus X_j^*, \alpha, \beta, ET$ )
10:    end for
11:   else
12:      $ET \leftarrow ET \cup \langle path, P(path | x_O) \rangle$             $\triangleright$  update the ET adding path
13:   end if
14: end procedure

```

In algorithm 1, for each variable in the explanation set, X_j , we compute the sum of the amount of information that this variable provides about all the current explanation variables conditioned to the current observations $x_O^* = (x_O, path)$. We are interested in the variable that maximises this value. In our study we have considered two classical measures:

– *mutual information*:

$$Inf(X_j, X_k | x_O^*) = I(X_j, X_k | x_O^*) = \sum_{x_j, x_k} P(x_j, x_k | x_O^*) \log \left(\frac{P(x_j, x_k | x_O^*)}{P(x_j | x_O^*) \cdot P(x_k | x_O^*)} \right)$$

– *GINI index*:

$$Inf(X_j, X_k | x_O^*) = GINI(X_j, X_k | x_O^*) = 1 - \sum_{x_j, x_k} P(x_j, x_k | x_O^*)^2.$$

Thus, there are different instances of the algorithm depending on the criterion used as *Inf*.

Once we have selected the next variable to be placed in a branch, we have to decide whether or not to expand this node. Again, we will use the measure *Inf*.

The procedure CONTINUE is the responsible to take this decision by considering the vector `Info []`. This procedure considers the list of values `Info[Xj*, Xk]` for $X_k \neq X_j^*$, then it computes the maximum, minimum, or average of them, depending on the particular criterion we are using. If this value is greater than α it decides to continue. Of course the three criteria give rise to different behaviours, being minimum the most restrictive, maximum the most permissive and having average and intermediate behaviour.

Notice that when only two variables remain in the explanation set, the one selected in line 5 is in fact that having greater entropy ($I(X, X) = H(X)$) if mutual information (or MI) is used. Also, when only one variable is left, it is of course the selected one, but it is necessary to decide whether or not it should be expanded. For that purpose, we use the same information measure, that is, $I(X, X)$ or $\text{GINI}(X, X)$, and only expand this variable if it is at least as uncertain (unstable) as the distribution $[1/3, 2/3]$ (normalising in the case of more than two states²). That is, we only add a variable if it has got more uncertainty than a given threshold.

10.2.1 Computation

Our inference engine is (mainly) based on Shenoy Shafer propagation algorithm running over a binary join tree [28]. Furthermore, we have forced the existence of a single cluster (being a leaf) for each variable in X_E , i.e. a clique which contains only a variable. We use these clusters to enter *as evidence* the value to which an explanatory variable is instantiated, as well as to compute its posterior probability.

Here we comment on the computation of the probabilities needed to carry out the construction of the explanation tree. Let us assume that we are considering to expand a new node in the tree which is identified by the configuration (path) $C = c$. Let x_O^* be the configuration obtained by joining the observations $X_O = x_O$ and $C = c$. Then, we need to calculate the following probabilities:

- $P(X_i, X_j | x_O^*)$ for $X_i, X_j \in X_E \setminus C$. To do this we use a two stage procedure:
 1. Run a full propagation over the join tree with x_O^* entered as evidence. In fact, many times only the second stage (i.e., *DistributeEvidence*) of Shenoy-Shafer propagation is needed. This is due to the single cliques included in the join tree, because if only one evidence item (say X) has changed³ from the last propagation, we locate the clique containing X , modify the evidence entered over it and run *DistributeEvidence* by using it as root.
 2. For each pair (X_i, X_j) whose joint probability is required, locate the two closest cliques (C_i and C_j) containing X_i and X_j . Pick all the potentials

² This normalisation has been done by using $\frac{H(X_i)}{\log|\Omega_{X_i}|}$ for entropy (MI) and $\frac{\text{GINI}(X_i)}{\frac{|\Omega_{X_i}|-1}{|\Omega_{X_i}|}}$

for GINI index.

³ Which happens frequently because we build the tree in depth, and (obviously) the create-node algorithm and the probabilistic inference engine are synchronised.

in the path between C_i and C_j and obtain the joint probability by using variable elimination [29]. In this process, we can take as basis the deletion sequence implicit in the joint tree (but without deleting the required variables) and then the complexity is not greater than the complexity of sending a series of messages along the path connecting C_i with C_j for each possible value of X_i . But, the implicit triangulation has been optimised to compute marginal distributions for single variables, and it is possible to improve it to compute the marginal of two variables as in our case. The complexity of this phase is also decreased by using caching/hashing techniques, because some sub-paths can be shared between different pairs, or even a required potential can be directly obtained by marginalisation from one previously cached.

- $P(C = c|x_O) = \frac{P(C=c,x_O)}{P(x_O)}$. This probability can be easily obtained from previously described computations. We just use $P(x_O)$ that is computed in the first propagation (when selecting the variable to be placed in the root of our explanation tree) and $P(x_O^*) = P(C = c, x_O)$ which is computed in the current step (full propagation with x_O^* as evidence).

Though this method requires multiple propagations, all of them are carried out over a join tree obtained without constraining the triangulation sequence, and so it (generally) has a size considerably smaller than the join tree used for partial abductive inference over the same explanation set [17, 18]. Besides, the join tree can be pruned before starting the propagations [18].

10.3 Initial Testing: Preliminary Study Cases

In order to show how it works and the features of the provided explanations, we found interesting to use some (toy) networks having a familiar meaning for us, to test whether the outputs are reasonable.

We used the following two cases:

1. **academe network**: it represents the evaluation for a subject in an academic environment, let us say, university, for example. This simple network has got seven variables, as Fig. 10.2 shows. Some of them are intermediate or auxiliary variables. What this network tries to model is the final mark for a student, depending on her practical assignments, her mark in a theoretical exam, on some possible extra tasks carried out by this student, and on other factors such as behaviour, participation, attendance... We have chosen this particular topic because the explanations are easily understandable from an intuitive point of view.

In this network we consider as evidence that a student has failed the subject, i.e., $x_O \equiv \{finalMark=failed\}$, and we look for the best explanations that could lead to this fact. We use $\{Theory, Practice, Extra, OtherFactors\}$ as the explanation set. In this first approach we run our ET-based algorithm

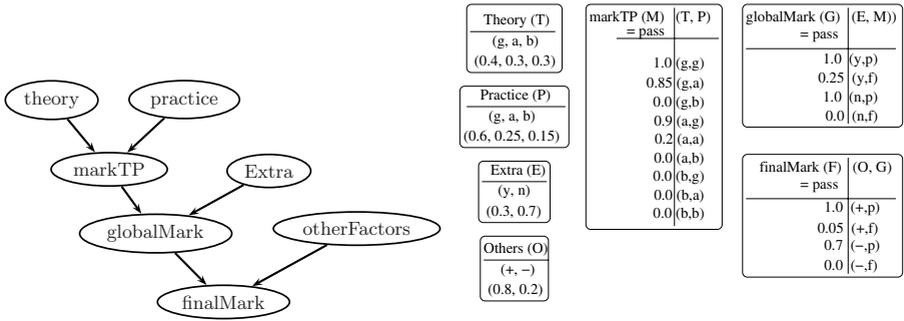


Fig. 10.2. Case of study 1: academe network

with $\beta = 0.0$ (i.e. the growing of the tree is not limited when the explanations have very little probability), $\alpha=0.05|0.07$ and $\text{criterion} = \max|\min|\text{avg}$. Fig. 10.4 summarises the obtained results (variables are represented by using their initials).

- gates network: this second net represents a logical circuit (Fig. 10.3.a). The network (Fig. 10.3.b) is obtained from the circuit by applying the method described in [30]. The network has a node for every input, output, gate and intermediate output. Again, we use an example easy to follow, since the original circuit only has got seven gates (two NOT-gates, two OR-gates and three AND-gates) and the resulting network has 19 nodes.

In this case, we consider as evidence one possible input for the circuit (ABCDE = 01010) plus an erroneous output (given such input), KL=00. Notice that the correct output for this case is KL=01, and also notice that from the transformation carried out to build the network, even when some gates are

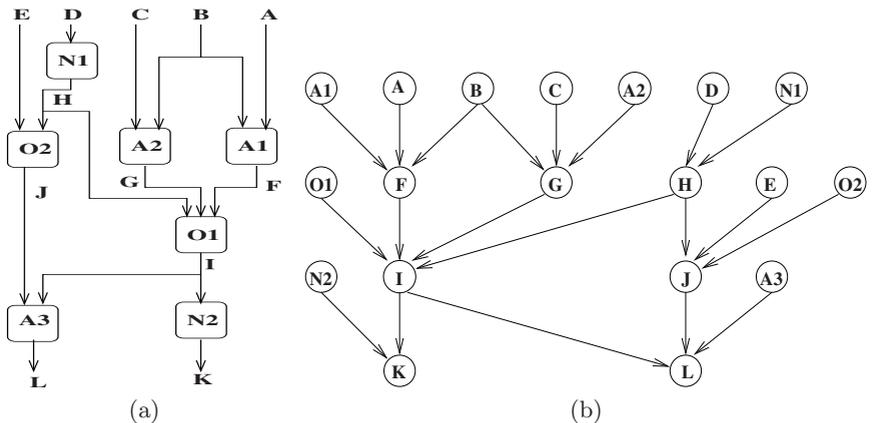


Fig. 10.3. (a) Original logic circuit (b) Network gates obtained from (a) by using the transformation described in [30]

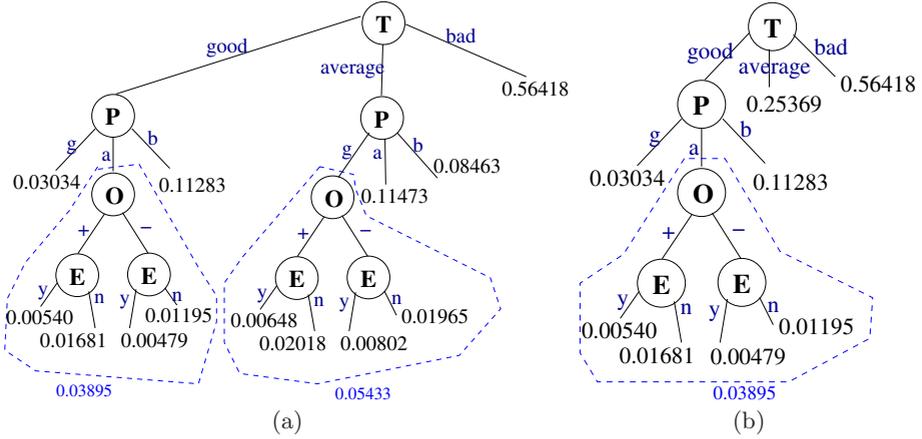


Fig. 10.4. Results for *academe*: (a) is the obtained tree for all MI cases except (MI, $\alpha=0.05,\min$) which produces tree (b) together with all (gini, $\alpha=0.05$) cases and (gini, $\alpha=0.07,\max$). Finally it is necessary to remark that (gini, $\alpha=0.07,\min|\text{avg}$) leads to an empty tree, \emptyset , that is no node is expanded. β is 0.0.

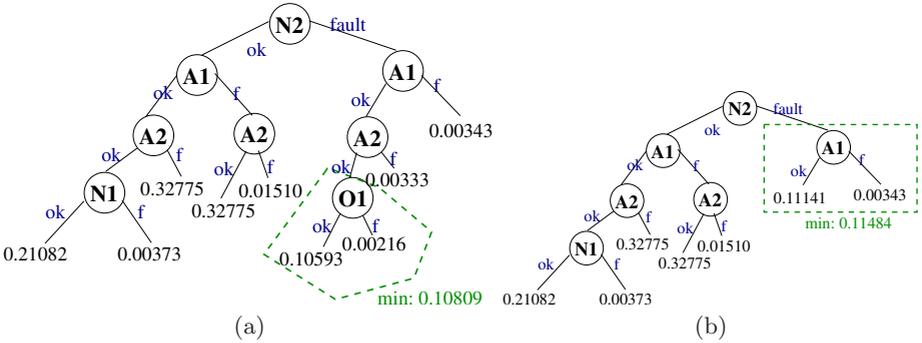


Fig. 10.5. Results for *gates* and MI: (a) is the obtained ET for (MI, $\alpha=0.05,\max|\text{avg}$) and also (MI, $\alpha=0.07,\max$); (b) is for (MI, $\alpha=0.07,\text{avg}$). In both cases min prunes more the tree than avg, so the dotted area would not be expanded. β is 0.05.

wrong the output could be correct (see [30]). So our evidence is ABCDEKL = 0101000 and we consider $X_E = \{A1, A2, A3, O1, O2, N1, N2\}$ as the explanation set with the purpose of detecting which gate(s) is(are) faulty. Figures 10.5 and 10.6 show the trees obtained for mutual information (I) and GINI respectively. The same parameters as in the previous study case are used but $\beta = 0.05$.

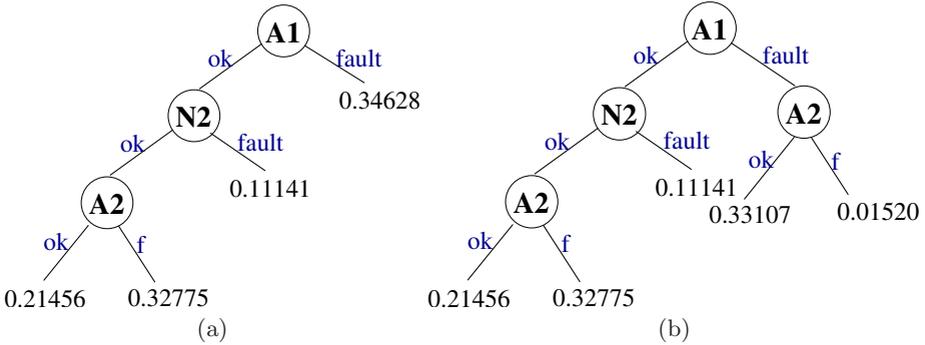


Fig. 10.6. Results for gates and GINI: (a) represents the tree for all gini cases, except $(gini, \alpha=0.05, \max)$ which produces tree in part (b). β is 0.05.

10.3.1 Analysis of the Obtained Trees

The first thing we can appreciate from the obtained trees is that they are *reasonable*, i.e., the produced explanations are those that could be expected.

Regarding the academe network, when a student is failed, it seems reasonable that the most explicative variable is *theory* because of the probability tables introduced in the network. Thus, in all the cases *Theory* is the root node, and also in all the cases $\{theory=bad\}$ constitutes an explanation by itself, being in fact the most probable explanation (0.56).

The other common point for the obtained ETs is that the branch with *theory* as good is always expanded. It is clear that being *theory* ok another reason must explain the failure. On the other hand, the main difference between the two ETs is that 10.4.(a) expands the branch $\{theory=average\}$ and (b) does not. It is obvious that a bigger α makes the tree more restrictive. If this tree is expanded, as $\alpha=0.05$ does, is because when *theory* is average it can be interesting to explore what happens with the *practical* part of the subject.

In some cases, it can be useful to add to the explanation tree those variables that are not part of an explanation, but that change their 'a priori' usual value or that have an important change in its 'a priori' probability distribution could be added to the explanation as this could be useful to the final user to fully understand some situations. An example can be the case of academe network with $\{theory = good, practice = good\}$. This branch is not expanded. The reason is that in this situation, the other variables have small entropy: *Extra* should be 'no' and *OtherFactors* '-', with high probability. This implies an important change with respect to 'a priori' probabilities for these values, and then these variables with their respective values could be added to the explanation $\{theory = good, practice = good\}$, making its meaning more evident.

We also used this case to show the influence of β . As $\beta = 0.0$ was used, we can see that some branches represent explanations with a very low posterior probability (those in the dashed area in Fig. 10.4), and so they will not be

useful. The dashed areas in Fig. 10.4 represent the parts of the tree that are not constructed if we use $\beta \simeq 0.05$, which apart of producing a simpler and more understandable tree is also of advantage to reduce the computational effort (probabilistic propagations) required to construct the tree.

With respect to the resulting trees for the **gates** case, we can appreciate two clear differences: (1) GINI produces simpler trees than MI, and (2) the most explicative variable is different depending on the used measure. Regarding this last situation, we can observe in the circuit that there are many independent causes⁴ (faults) that can account for the erroneous output. Choosing the AND gate A1 as GINI does is reasonable (as well as choosing A2) because AND gates have (in our network) greater a priori fault probability. On the other hand, choosing N2 as MI does is also reasonable (and perhaps closer to human behaviour) because its physical proximity to the wrong output. If we were a technician this would probably be the first gate to test. In this way, it seems that MI is more sensitive to the fact that the impact a node has in the value of the remaining nodes is attenuated with the distance in the graph.

Once the first variable has been decided, the algorithm tries to grow the branches until they constitute a good explanation. In some cases, it seems that some branches could be stopped early (i.e. once we know that `N2=fault`), but these situations depend on the thresholds used and it is clear that studying how to fix them is one of the major research lines for this work.

Perhaps an interesting point is to think about why O1 is not selected by MI when `N2=ok` as could be expected given the distance-based preference previously noticed. But, in this case AND gates have more prior of failing than OR gates.

Of course, we get different explanations depending on the used measure, the value of α or the criterion, but in general we can say that all the generated explanations encode reasonable descriptions of the possible scenarios. Finally, in all the trees there is a branch, and so an explanation which indicates that a set of gates are ok. Perhaps this cannot be understood as an explanation to a fault, but we leave it in the tree in order to provide a full partitioning. Some advice about these explanations can be given to the user by indicating for example if such explanations raise or not the probability of the fault with respect to its prior probability.

10.4 Further Experimentation

From the previous study and the examined examples we can extract that the explanations given by our *ET*-method are quite reasonable. Anyway, this reasonability term might be a quite vague concept and we wish to reinforce the belief on the goodness of the generated explanation tree. Hence, we have figured out a more systematic way to contrast the obtained results to an already

⁴ However, it is interesting to observe that applying probability propagation, the posterior probability of each gate given the evidence, e.g. $P(A1|x_O)$, indicates that that for all the gates it is more probable to be ok.

existing technique with the same purpose: K Most Probable Explanations search. In subsection 10.4.1 we will apply both techniques to the *academe* and *gates* problem in order to compare the quality of explanations, and the differences in the format of giving them (always using global configurations or not). With the aim of going into greater detail, the next step will be a comparison with the already cited method using simplification of explanations [26]. To do so, we will use again the *gates* network and another of the networks employed in [26] which models the start mechanism for a car (subsection 10.4.2).

10.4.1 *Explanation Tree vs. Partial Abduction*

In this part of the chapter we intend to see the behaviour of our method and set the given solution against the K -best explanations generated by partial abduction. To be *fair*, and since the output of both methods is different, we will try to make a kind of translation from one to the other, in such a way that:

1. $ET \rightarrow K$ -best explanations: From an Explanation Tree we will indicate the corresponding ranking of explanations, ordered by probability. In our method every leaf node represented one explanation, that one from the root until this leaf. Thus, this is almost immediate to be done, but that will make easier the search of similarities/differences when it is contrasted with the K -best explanations. With this, we can see if the ET is able of representing these K -best explanations and in which form.
2. K -best explanations $\rightarrow ET$: The K -best explanations provided by the abduction task will be reflected in a tree structure similar to the resulting ET. So we will annotate, how many explanations are included in each branch (and which ones). In this case we will measure the distributions of explanations along the tree, and see that, as expected, in partial abduction there are sets of configurations that could be aggregated in only one solution for the sake of simplicity (Principle of parsimony).

In this case, we do not deem necessary to go through all examples, covering the whole bunch of versions for ET provided before. We have decided to perform this illustrative proof with the most representative examples.

Academical Example

For the *academe* network, we will look at the tree in figure 10.4.(a). We assume β -pruning has been performed, so the dotted area is removed as figure 10.7 shows. We have chosen this tree (the largest one) on purpose, to avoid starting from an advantageous situation. To get the K -best explanations, we have fixed the K value to 20. We think it is a big enough number to guarantee that we allow partial abduction to have quite a lot explanations, more than the number of leaves, by far. In this *academe* problem MI and GINI happened to behave quite similarly, so we have not considered relevant to distinguish both cases.

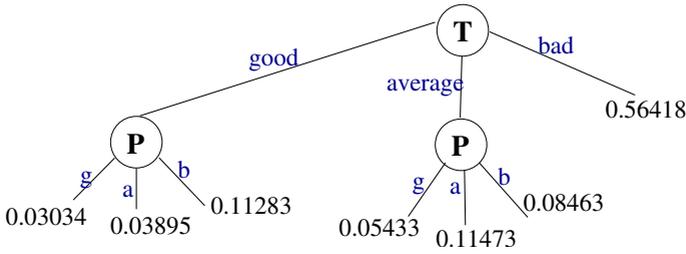


Fig. 10.7. ET for academe and MI used for comparison and which is α, β -pruned

#	Theory	Practice	Other	Extra	Prob.
1	bad	good	+	no	0.201776
2	bad	avg	+	no	0.084076
3	avg	avg	+	no	0.067259
4	good	bad	+	no	0.067259
5	bad	good	+	yes	0.064857
6	bad	good	-	no	0.053099
7	avg	bad	+	no	0.050044
8	bad	bad	+	no	0.050044
9	bad	avg	+	yes	0.027024
10	avg	avg	+	yes	0.027024
11	bad	avg	-	no	0.022124
12	good	bad	+	yes	0.021619
13	good	good	-	no	0.021240
14	avg	good	+	no	0.020178
15	bad	good	-	yes	0.019877
16	avg	good	-	no	0.019647
17	avg	avg	-	no	0.019027
18	good	bad	-	no	0.017699
19	good	avg	+	no	0.016815
20	avg	bad	+	yes	0.016214

Fig. 10.8. 20-best explanations for the academe model when $X_O = \{finalMark = failed\}$ and $X_E = \{theory, practice, otherFactors, Extra\}$

In Fig. 10.8 we reproduce the output of running partial abduction (Elvira software [31]) on academe network with $K = 20$, just for the same example we went through in the previous section. On the other hand, Fig. 10.9 has transformed the corresponding tree (fig. 10.7) into another ordered list of explanations to have the same arrangement style. In the second figure we have preferred not to add a column for every variable, since most of them will not be instantiated to a particular value.

First, if we just watch the first explanation in both cases, they coincide on the value of *Theory* which is equal to bad. This is a good sign in the sense that the most probable explanation is the same with the two techniques, but is

#	Configuration	Prob.
1	{theory = bad}	0.56418
2	{theory = average, practice = average}	0.11473
3	{theory = good, practice = bad}	0.11283
4	{theory = average, practice = bad}	0.08463
5	{theory = average, practice = good}	0.05433
6	{theory = good, practice = average}	0.03895
7	{theory = good, practice = good}	0.03034

Fig. 10.9. Explanation ranking corresponding to *ET* in Fig. 10.7

this really exactly the same? Since the initial problem solving assumptions are different, these explanations are different too. In MAP the first explanation is {*Theory* = bad, *Practice* = good, *OtherFactors* = +, *Extra* = no}. It seems quite logical that a student having a bad theoretical part had failed, but why should we assume he has done good practical assignments as this explanations says? That could lead to think that having bad practical assignments would not be a so nice explanation for this fact, and actually that would give even more reasons to think that the student has failed. The same happens with the *otherFactors* value, it is positive, but what if *otherFactors* would have been negative? Even worse, that will increase the possibilities of the student to fail the subject. That leads us to think that ET-explanations are more appropriate than K-MPE ones.

Once the explanation content has been regarded, we should also look at the associated strength. In MAP the most probable explanation has a probability of 0.2. But in our ET the first explanation “*Theory* is bad” covers more than the half (56%) of the explanation space. We find this second number much more accurate: in most of the cases where a student has failed, a bad theoretical exam seems a good enough explanation. If we observe the prior probability tables *Theory* was precisely the most influential factor. This confirms our believe that when abduction requires configurations always with $|X_E|$ elements, there are some unnecessary variables $X_{unnecessary} \subset X_E$ ⁵ that change their states in the configurations in a counter-intuitive way (they are in fact being tuned), since this variable instantiation is not really significant for the explanation.

To understand better this important point, we jump to the other related illustration (translation K-best expl. → ET) in Fig. 10.10. Here we tried to make a picture on how the *K*-best explanations would be distributed in the ET. So, the structure is identical to the ET, i.e. a tree node is a variable, a branch indicates the state associated to this variable and leaf nodes represent explanations. Precisely at this lower level, where explanations are implicitly depicted, we have included the *K*-best explanations data:

⁵ Notice that the set of unnecessary variables can vary depending on the given configuration, that is, on the other variables and values in the current explanation/configuration. Apart from this, it could happen that some of them are in all cases unnecessary.

- Together with the last branch configuration, and between brackets [], we annotate the explanations that would fall in that branch/explanation. The number indicates the ranking, as in column # of Fig. 10.8.
- Just on the leaves a triangle indicates the number M of explanations that are included in this position ($Mexpl.$) and below it we write the sum of their probabilities in parentheses, that is:

$$\sum_{x_E \in Expl_{path}} P(x_E|x_O)$$

where $Expl_{path}$ is the set of explanations in path going from root to the leaf and $|Expl_{path}| = M$.

For example explanation #14: {*Theory*=avg, *Practice*=good, *OtherFactors*=+, *Extra*=yes} belongs to path <*Theory*=avg, *Practice*=good>.

- Finally, in braces we indicate the branch/path probability as used in the ET representation.

Then, in Fig. 10.10 we can see how {*Theory* = bad} piles up the biggest number of explanations (8 expl.), even if we look at all the branches at the same detail level {*Theory* = good} has 5 and {*Theory* = avg } presents 7. However it is not so a question of quantity, but mainly *quality* and what is clear is that branch {*Theory* = bad} includes 7 of the 10-best explanations!!! Notice that in Fig. 10.8 if we apply a *similar* criterion to our β -pruning, with $\beta = 0.05$ we could have just taken the eighth first explanations, the other could be considered as too little relevant. Anyhow, as we have remarked above, we show them because we wanted to take a big *mass* of explanation in order to make the comparison advantageous to partial abduction in the way that many explanations are regarded. In a practical case, the user would not normally need to have up to 20 different reasons.

The main conclusion from the previous discussed point is that when we perform the search of the K -best explanations, they are not necessarily K . With

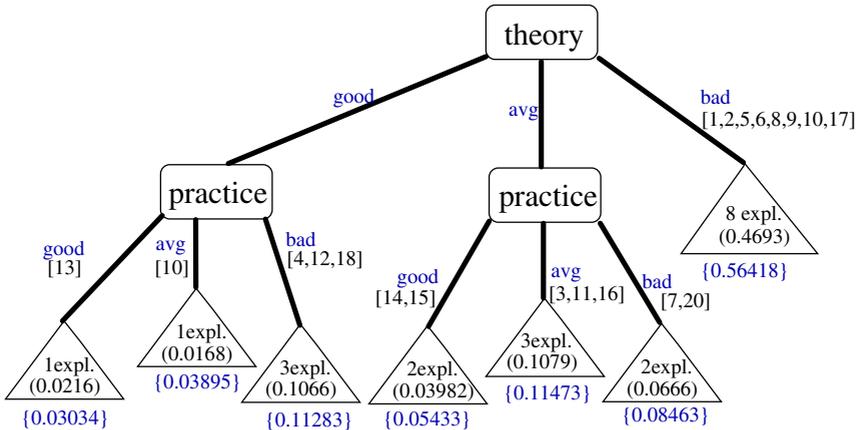


Fig. 10.10. ET structure for academe including 20-best explanations

the resulting ET we can see that for example 7 out of 10 explanations could be (and should be in our opinion) amalgamate in only one.

Even if we think that our method manages to outperform partial abduction in both efficiency and results, there are still doubtful issues in its own nature. For example, with the same principle as the previously explained, we could think that having *Theory* as average could also be an explanation by itself. Observe that in Fig. 10.9, ET-explanations #2,#4 and #5 present that configuration and they differ in the practical part value. But here there could be nuances to take into account. For example having *theory* and *practice* ordinary (avg) is the second explanation, very near from having bad *practice* and good *theory* whereas average *theory* and good *practice* decrease considerably the probability when we know that the student has failed. This instability is even more clear when *theory* is good, we need to distinguish the possible states for *Practice* to reach a valid explanation. Look at the different probabilities for these three branches: $\{Practice = bad\}$ is 4 times more probable than the other two. Nevertheless, in the 20-best explanations this one would appear in the forth place without standing out specially from the rest, and later this explanation reappears in #12 and #18.

Circuit Example

Unlike the first example, the provided ETs differ a lot from using MI or GINI as the *Info* measure both in size and also in the nodes situation along the tree. So, in this second case, we are going to divide this comparative study into two parts: one for tree in Fig. 10.5.(a) and the other for tree in fig. 10.6.(a).

●When Info is Mutual Information

The notation for figures is exactly the same as for the previous example. So, we can find the 20-best explanations in fig. 10.11, the ranking for ET in fig. 10.13 and the *integrated* tree in fig. 10.14.

Looking at the two rankings, again we can detect that not only the first explanation, but the two first (which are equiprobable) are alike in the sense that the extracted anomaly is that either *A1* does not work properly or the failure is in *A2*. But again, since the K-MPEs need a value for every variable, this method burdens this explanation with values that are not relevant, all the other gates are set to *ok*. Here the overspecification problem is again quite clear. And, from the quantitative point of view (probability ordering) the same situation repeats: as the significant data is that gate *A1* (or *A2*) fails, the rest of configurations when $A1 = fault$ are regarded here, considering only one of these configurations instead of integrating them as when using the ET model. Luckily, in this example, this difference is slightly greater than 0.015 (0.32775 - 0.311406), because the accumulation of two gates faults is quite improbable and only one of the configurations has most of the probability mass. To check that, we should just look explanations from #6 to #20, where the total probability barely reaches

#	N1	N2	A1	A2	A3	O1	O2	Prob.
1	ok	ok	ok	fault	ok	ok	ok	0.311406
2	ok	ok	fault	ok	ok	ok	ok	0.311406
3	ok	ok	ok	ok	ok	fault	ok	0.205486
4	ok	fault	ok	ok	ok	ok	ok	0.101705
5	ok	ok	fault	fault	ok	ok	ok	0.014447
6	ok	ok	ok	fault	ok	fault	ok	0.006355
7	ok	ok	fault	ok	ok	fault	ok	0.006355
8	ok	ok	ok	fault	fault	ok	ok	0.004815
9	ok	ok	fault	ok	fault	ok	ok	0.004815
10	ok	ok	ok	fault	ok	ok	fault	0.003178
11	ok	ok	fault	ok	ok	ok	fault	0.003178
12	ok	ok	ok	ok	fault	fault	ok	0.003178
13	ok	fault	ok	fault	ok	ok	ok	0.003145
14	ok	fault	fault	ok	ok	ok	ok	0.003145
15	ok	ok	ok	ok	ok	fault	fault	0.002097
16	ok	fault	ok	ok	ok	ok	fault	0.002076
17	ok	fault	ok	ok	ok	fault	ok	0.002076
18	fault	ok	ok	fault	ok	ok	ok	0.001573
19	fault	ok	fault	ok	ok	ok	ok	0.001573
20	fault	ok	ok	ok	fault	ok	ok	0.001573

Fig. 10.11. 20-best explanations for gates with $X_O = \{ABCDEKL = 0101000\}$ and $X_E = \{A1, A2, A3, O1, O2, N1, N2\}$

0.07. Also, it is curious to see how three faults are not even considered within the 20 best explanations.

In *ET* depending on the side of the tree, there could be three or four gates included in the explanation. This gives us the hint that this designed algorithm is also able of discerning those variables within the explanation set which are in fact relevant to explain the given observations. *A1* and *A2* (with a symmetrical behaviour for this example) together with *N2* are clearly the three involved gates that could have caused the given error. In this particular example, even without considering the threshold $\beta(=0.0)$, gates *O2* and *A3* never appear in the explanation tree. We can see that the erroneous output is in signal *K* and these two gates do not play a role for that value. In certain cases *O1* (depending on the thresholds' values) could appear. As we already commented, this gate, even if it is related to the *bad* output *K*, increases its belief of working properly (or, the other way round, decreases its belief of being faulty) because it also participates in producing signal *L*, which is correct. In *K*-best explanations this happens to be the third possible faulty gate, so we find that this implication (*L* signal valid then *O1* is likely to work properly) is not caught there.

The nice feature about *ET* selecting the explanatory variables has also been shown in the *academe* example where *Theory* and *Practice* stood out as the significant variables.

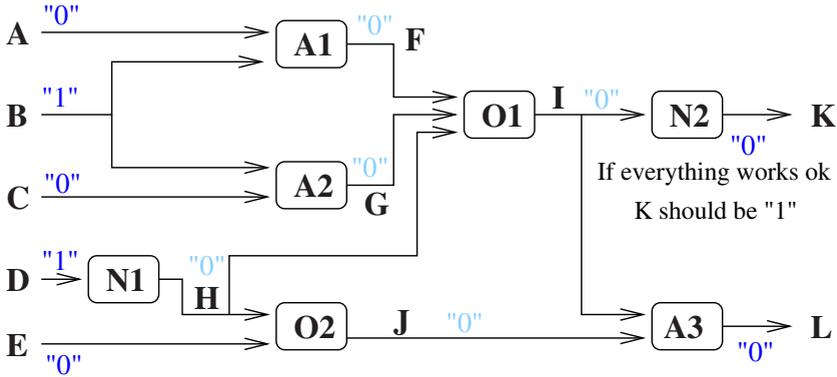


Fig. 10.12. Circuit with the evidence (ABCDEKL=0101000) incorporated and the assumed signal values when if everything works ok annotated

#	Configuration	Prob.
1	{N2 = ok, A1 = ok, A2 = fault}	0.32775
2	{N2 = ok, A1 = fault, A2 = ok}	0.32775
3	{N2 = ok, A1 = ok, A2 = ok, N1 = ok}	0.21082
4	{N2 = fault, A1 = ok, A2 = ok}	0.10809
5	{N2 = ok, A1 = fault, A2 = fault}	0.01510
6	{N2 = ok, A1 = ok, A2 = ok, N1 = fault}	0.00373
7	{N2 = fault, A1 = fault}	0.00343
8	{N2 = fault, A1 = ok, A2 = fault}	0.00333

Fig. 10.13. Explanation ranking corresponding to ET in figure 10.5.(a)

Also, let us comment that the set of *paired* explanations (for instance #1 and #2, #8 and #9, #10 and #11, ...) comes from the symmetrical influence of gates A1 and A2. From figure 10.12 this symmetry can easily be observed.

A glance at the combined tree in fig. 10.14 reinforces the conjectures previously remarked. The two branches that accumulate a larger number of (MPE) explanations (5 of them each) are exactly the two first candidates: A1 is faulty or A2 is faulty. Next, unexpectedly maybe, we have that gates N2, A1, A2, N1 are ok. But in this case, we have that O1 is faulty with probability close to 1.0. So, we have again an example in which it seems reasonable to expand some given explanations with the variables that has experiment an important modification of their prior probabilities.

Again, notice that if we had taken for example the 10-best explanations, our tree would have given even a deeper level of detail just with 8 leaves/explanations because some leaves will not correspond to any of the 10-best explanations, ET adds then more information.

Finally, we would like to add again that the small probabilities of explanations from #6 to #20 prove that they were unnecessary. Even though, with a few

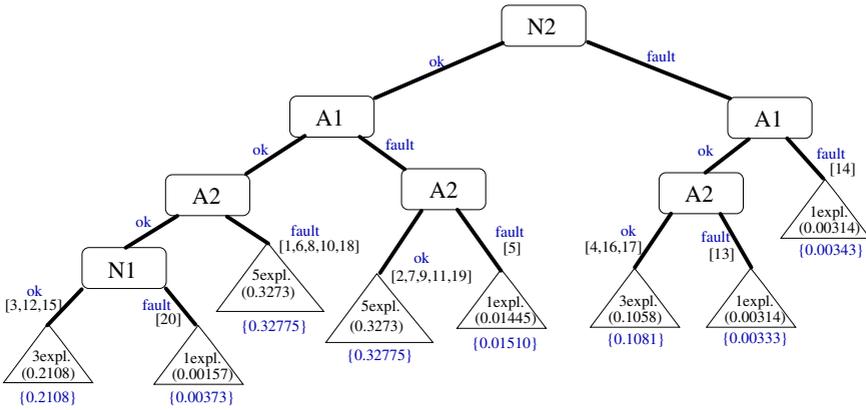


Fig. 10.14. ET structure for gates with MI including 20-best explanations

#	Configuration	Prob.
1	{A1 = fault}	0.34628
2	{A1 = ok, N2 = ok, A2 = fault}	0.32775
3	{A1 = ok, N2 = ok, A2 = ok}	0.21456
4	{A1 = ok, N2 = fault, A2 = ok}	0.11141

Fig. 10.15. Explanation ranking corresponding to ET in Fig. 10.6.(a)

propagation steps with the explanation tree we are able of capturing all of them. Besides, these explanations are presented to the user in a more intuitive and simpler manner.

●When Info is GINI Index

Using GINI index instead of MI has changed the order of selected variables in the tree, as reviewed in section 10.3. Also, see that in Fig. 10.6.(b) the second level variable is distinct depending on the state taken by the root variable A1. Here we are going to examine tree 10.6.(a) and when incorporating the 20-best explanations on it we obtain the one depicted in fig. 10.16. As this tree shows and as the corresponding ranking (fig. 10.15) does too, again the two main explanations are that either A1 or A2 does not work. In this case A1 has been first selected, but we think is a question of tie breaks. Apart from the tree level of the involved gates and the slight difference between ET-explanations #1 and #2 the same conditions as with MI accomplish. But this time we have the advantage of presenting a more compact tree, where the system determines that the relevant gates to be tested are A1, A2 and N2 in this order (according to the probability ranking). This is influenced by the thresholds α and β ⁶, as the

⁶ And by the min, avg or max criteria as well.

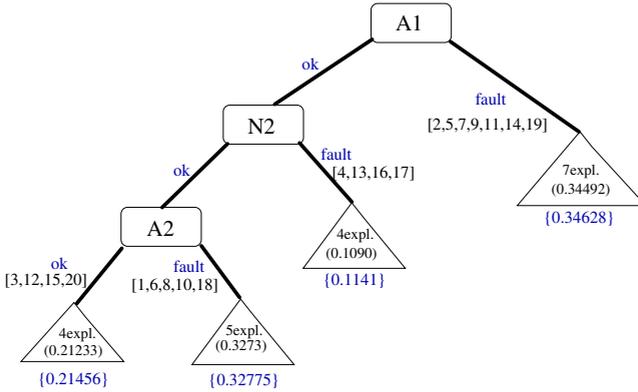


Fig. 10.16. ET structure for gates with GINI including 20-best explanations

other technique was influenced by the K number. The user can make a choice on these values depending the level of detail s/he requests. But for a general case these explanations seem good enough and they have been reached with a reasonably low effort, just a few propagations on the join tree. Once again, we can also say, that the method has made a reduction of the explanation set from seven variables down to only three, which also simplifies quite a lot the problem.

10.4.2 Looking into Simplification Methods. A New Example: Car-Start

To finish in this evaluation of the *Explanation Tree* technique, we find interesting to add some detail about the already mentioned method of simplifying explanations [22, 26]. We just would like to set out the basic ideas in order to see similarities and differences of this method with our proposal. In these two works, the objective was to simplify the explanations given after applying MPE as figure 10.17 illustrates. They also state the process as follows: “Let $expl(x_O) = \{x_E^1, x_E^2, \dots, x_E^k\}$ be the K MPEs obtained for evidence $X_O = x_O$. Then, for all $x_E \in expl(x_O)$ we are looking for a sub-configuration $x'_E [X'_E \subset X_E]$, so that x'_E is still accounting for the observed evidence.”

So, the process is differentiated into two main steps:

1. Generation of complete explanations (configurations of X_E with $|X_E|$ literals), ordered by their posterior probabilities given the observations.
2. Simplification of these explanations by removing *unimportant* literals.

To design a mechanism for this *reduction* of explanations, the authors define and propose two important criteria: **Independence** (I~simplification) and **Relevance** (R~simplification)⁷. We avoid theoretical details and formulas,

⁷ In this work the authors develop other simplification techniques such as those induced by the graph, that we will not touch on here.



Fig. 10.17. Process followed in the simplification of explanations in [26]

aiming the conception of both criteria. The first one will try to detect those variables that are useless in the explanation since the values these variables take do not affect the evidence. In the case of relevance-based criteria they attempted to remove those literals that could be considered as insignificant in the sense that they are (almost) irrelevant for the observed evidence.

There is a notable difference between ET-based simplifications and this two stage process: in *ET* the root variable will always be in all the explanations whereas when simplifying explanations it could happen that one explanation is $\{A = a_1, B = b_1\}$ and another one $\{C = c_1, D = d_1\}$.

It is clear that both approaches come up from the same concern: avoid the overspecification problem when dealing with abductive inference in BNs. In fact, the current work was inspired on the previous one, but attempting to skip its two main drawbacks: (1) little efficiency.- since it requires a two-step process and the first one includes K-MPEs and (2) being this simplification *K-MPE guided* the obtained solutions are somehow influenced by them. As we have just verified, in many occasions the *K*-best explanations do not correspond to *K* different scenarios, because some explanations should have been aggregated into one. Remember the previous example, where 7 of the 10th best explanations could be summarised in only one. We observe that these 7 explanations are just the combination of the simple explanation extended with different configurations of other variables. We also believe this happens very frequently when performing partial abduction. Actually, the experiments executed in [22] discovered that after applying simplification to the *K*-best explanations, most of the times the same subset of simplified explanations were repeated following a certain pattern.

To confirm our feeling that ET succeeds not only in giving explanations with different number of literals, but also in improving the simplification procedure in [26] we are going to take a couple of examples in this work and apply the ET-algorithm on them.

To start with, we take again the *gates* network, but this time we fix the evidence to $X_O = \{D = 0, K = 1, L = 1\}$ and the explanation set to $X_E = \{F, G, H, I, J, A1, A2, A3, O1, O2, N1, N2\}$. To have a visualisation of this situation we have depicted fig. 10.18. Weq have selected the explanation set as those variables that are not observable in the network.

For this example in [22] it is applied a technique called successive explanations search that after several stages gives rise to explanation $X_S^5 = \{A3 = ok\}$.

If we execute the ET-algorithm on this same data and some *standard* thresholds (for example $\alpha = 0.05, \beta = 0.05$) with MI the given tree is as simple as the one depicted in figure 10.19.

Well, we could say that both explanations are not inconsistent, *A3* can work properly at the same time as *N2* presents a fault. Besides, if *A3* works, a failure in

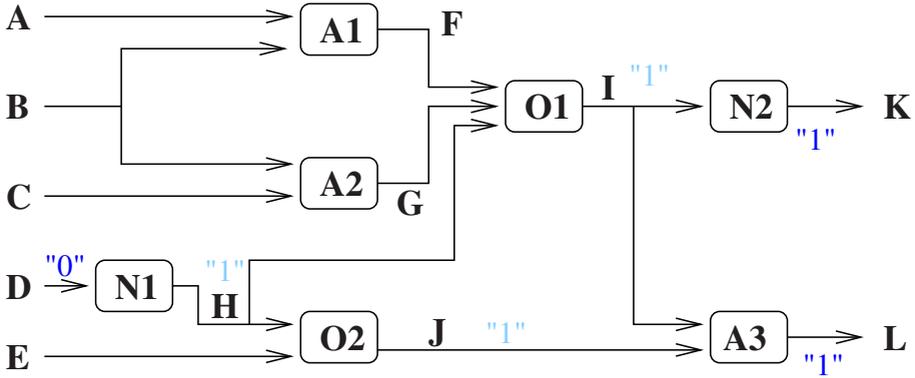


Fig. 10.18. Circuit with the evidence (DKL=011) incorporated and the assumed signal values when everything works ok annotated. Notice that here there is a conflict: if output L is 1, and being $H = 1$ an input of the OR-gate $O1$, then I should be 1, but that will imply $K = 0$ which is inconsistent with the evidence.

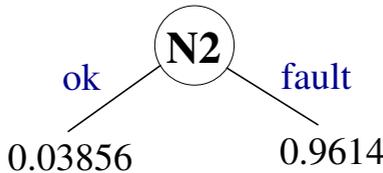


Fig. 10.19. ET obtained for the gates with (MI,0.05,max) and $\beta = 0.05$ in the situation of fig. 10.18

$N2$ is the fact that better would explain this circumstance, because the output of $A3$ should be correct. Thus, if $L = 1$ that means that $I = 1$, but this is an input for the NOT-gate $N2$ that should have been the opposite signal, 0, according to the introduced evidence. We think that the second explanation ($\{N2 = fault\}$ with 96% of strength) is quite more informative in a diagnosis task. So we really find that our answer is of better quality than the other one.

Another Example: Car-Start Problem

As a second example we have taken the car-start network from [26]⁸. We are going to firstly introduce a simple and intuitive example, where the evidence is that the car does not start ($Starts = No$), and the explanation set is $\{X_E = Alternator, FanBelt, Leak, Charge, BatteryAge, BatteryState, BatteryPower, GasInTank, Starter, EngineCranks, Leak2, FuelPump, Distributor, SparkPlugs\}$.

⁸ They indicate that this network has been originally found in JavaBayes package. <http://www.cs.cmu.edu/~javabayes> is the web site.

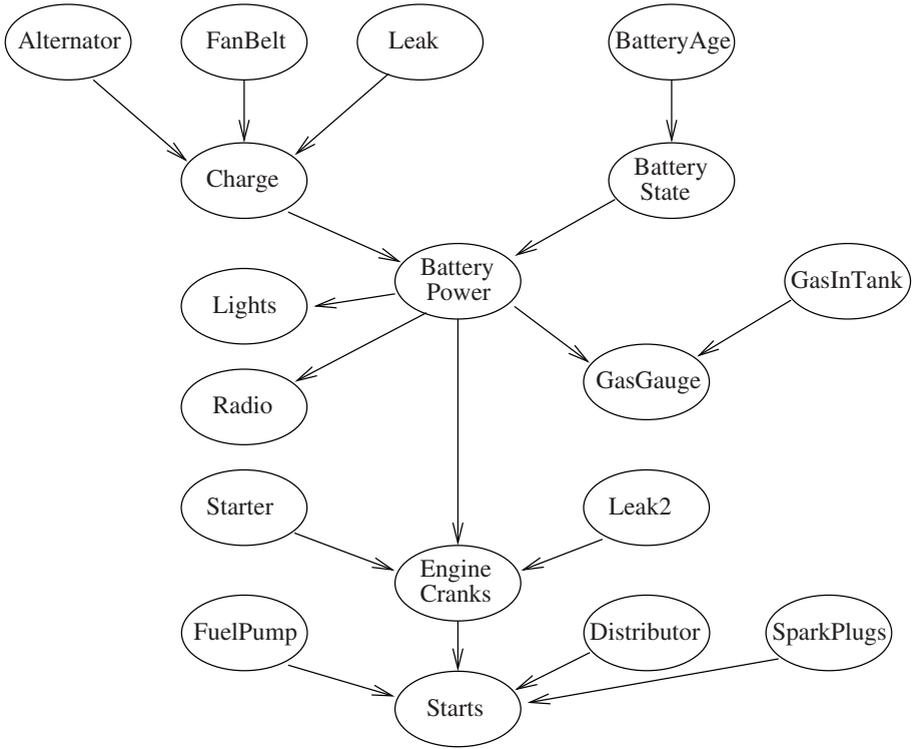


Fig. 10.20. Network modelling the car-start problem

With an *ET*-execution of kind $[\text{Info}, \text{criterion}, \alpha, \beta] = [\text{MI}, \text{min}, 0.07, 0.05]$ the resulting tree is the one in fig. 10.21.

Even with real probability tables unknown, any person with a little knowledge about cars could try to interpret this answer which just says that the two most probable explanations are that the battery state is weak (0.77) or as a second and less probable explanation (0.13) that the starter could be faulted.

Let us now show a more complex case studied in [22] where evidence is $X_O = \{GasGauge = Gas, Lights = Work, Radio = Works, Starts = No\}$ and the explanation set is the same of the previous example. With the simplification method the final obtained explanation is $\{GasInTank = Yes, Starter = Faulted\}$. We have performed an execution of the Explanation Tree algorithm of kind $(\text{MI}, \text{min}, 0.07, 0.05)$ as before, and the given tree is drawn in fig. 10.22. In this case we find that both simplified explanations are quite close. It is clear that the starter does not work properly, which is probably the main explanation. But both simplification and ET go further adding also that there is no problem with the gas status. In simplification it says that there is gas in the tank while ET has *checked* the possibility of presenting a leak of kind “2”, but this is quite

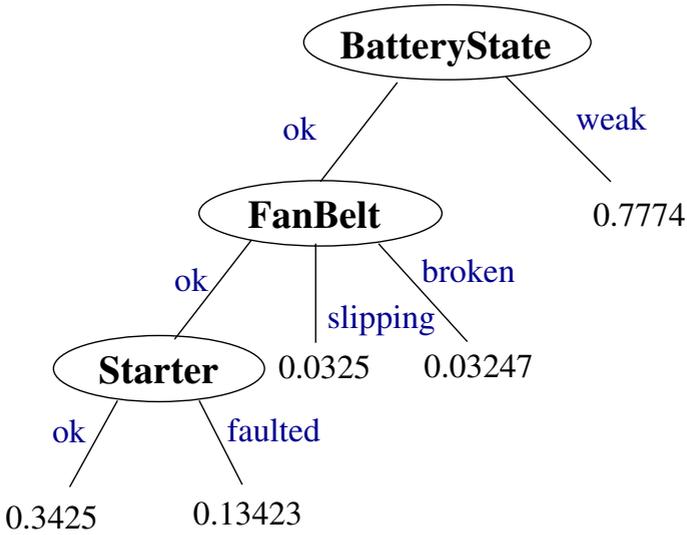


Fig. 10.21. ET for the car-start problem, when $\{Starts = No\}$ and $(MI, min, 0.07, 0.05)$

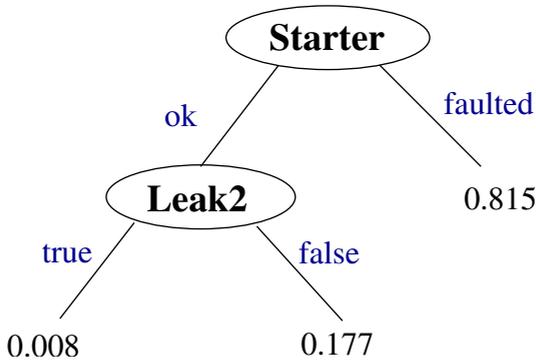


Fig. 10.22. ET for the car-start problem, when $\{GasGauge = Gas, Lights = Work, Radio = Works, Starts = No\}$ and $(MI, min, 0.07, 0.05)$

improbable. Since the evidence included that the gas gauge indicates enough gas, and this device is not determined as faulted, we could have supposed that it works properly and therefore $\{GasInTank = Yes\}$.

We have performed a final comparison “*simplification vs. ET*” taking randomly selected examples from [26]. In this study they just indicate the explanation set and the observed variables and, using different types of algorithms for simplification, they show the obtained number of literals. We have then reproduced these tests with all the possible configurations for the observed variables

with (MI,max,0.07,0.05) and in all cases either the tree was empty or it presented one variable. In simplification experiments the number of literals could vary from 1.5 until 6. In the cases where an empty tree was obtained, it could indicate that all variables have a very low entropy and they present a fixed value. Then, there is only one possible scenario that will be the most probable explanation. On the other hand, in the cases with one variable the distribution was quite clear giving probability numbers such as 0.933 or 0.8146.

10.5 Discussion and Main Conclusions

This chapter has proposed a new approach to the problem of obtaining the most probable explanations given a set of observations in a Bayesian network. The method provides a set of possibilities ordered by their probabilities. The main novelties are three:

1. The level of detail of each one of the explanations is not uniform (with the idea of being as simple as possible in each case).
2. The generated explanations are mutually exclusive.
3. The number of required explanations is not fixed (it depends on the particular case we are solving).

Our goals are achieved by means of the construction of the so called *explanation tree* which can have asymmetric branching and that will determine the different possibilities.

We have described the procedure for its computation based on information theoretic criteria. To show its behaviour some simple examples have been proved and we have presented a comparison with the K -best explanations and with the simplification of explanations. From this experimental analysis, we can conclude that our method outperforms the K -best explanations search by far both in quality and efficiency, since our method minimises the number of propagations and manage to do them in a very quick form. We find that the description given by an ET is more useful for a user than the one given by K complete explanations because in the former case the possibilities are given in a more compact and structured way.

With respect to efficiency we have that ET construction ($\beta > 0$) is polynomial⁹ when probabilistic propagation is polynomial as in the case of polytrees, whereas K -MPEs are NP-hard in polytrees.

Regarding the simplification of explanations, we could say that the obtained explanations are different, but not always of different quality. Nevertheless, *ET* is able of giving simpler explanations and what is more important it achieves the task in a direct way, avoiding the cost of performing *traditional* abduction first.

⁹ Each expansion of a node in the tree implies a polynomial number of probability propagations and the number of nodes in the tree is bounded by a function of β .

10.6 Further Work

Our main conclusion is that this technique behaves in a very satisfactory way, but we would like to run more sophisticated experiments in order to evaluate it in depth. We believe it is quite promising, even though we are conscious that a better refinement should be done for some issues such as the determination of α and β thresholds or the characterisation of those explanations given for partitioning the explanation space, but not relevant as an explanation.

As further work we plan also to study the possibility of using Kullback-Leibler distance as the Info measure. In addition, very useful comments have suggested us to explore the connection about this method and the so-called *Hitting set* which is quite broadly studied by researchers on the logical field together with theories of diagnosis [6]. Finally, we would like to analyse if our diagnosis method could be benefitted from the conflict analysis theory, as long as conflicts are applicable to a particular problem.

We also consider that clustering is an important field of application/extension for this method. So, we have already started a new research line oriented to that specific task where the Explanation Tree has constituted an inspiration. The idea and preliminaries about this technique and its structure, the *Independency Tree*, can be found in [32].

References

1. Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann, San Mateo (1988)
2. Castillo, E., Gutiérrez, J.M., Hadi, A.S.: Expert Systems and Probabilistic Network Models. Springer, Heidelberg (1997)
3. Jensen, F.V.: Bayesian Networks and Decision Graphs. Springer, Heidelberg (2001)
4. Lacave, C., Díez, F.J.: A review of explanation methods for Bayesian networks. *The Knowledge Engineering Review* 17, 107–127 (2002)
5. Gámez, J.A.: Abductive inference in Bayesian networks: A review. In: Gámez, J.A., Moral, S., Salmerón, A. (eds.) *Advances in Bayesian Networks*, pp. 101–120. Springer, Heidelberg (2004)
6. Reiter, R.: A theory of diagnosis from first principles. *Artificial Intelligence* 32(1), 57–95 (1987)
7. Peng, Y., Reggia, J.A.: A probabilistic causal model for diagnostic problem solving. *IEEE Transactions on Systems, Man, and Cybernetics* 17(2), 146–162 (1987)
8. Shimony, S.E.: Finding MAPs for belief networks is NP-hard. *Artificial Intelligence* 68, 399–410 (1994)
9. Dawid, A.P.: Applications of a general propagation algorithm for probabilistic expert systems. *Statistics and Computing* 2, 25–36 (1992)
10. Nilsson, D.: An efficient algorithm for finding the M most probable configurations in Bayesian networks. *Statistics and Computing* 8, 159–173 (1998)
11. Li, Z., D’Ambrosio, B.: An efficient approach for finding the MPE in belief networks. In: *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence*, pp. 342–349. Morgan Kaufmann, San Francisco (1993)

12. Seroussi, B., Goldmard, J.L.: An algorithm directly finding the k most probable configurations in Bayesian networks. *International Journal of Approximate Reasoning* 11, 205–233 (1994)
13. Gelsema, E.S.: Abductive reasoning in Bayesian belief networks using a genetic algorithm. *Pattern Recognition Letters* 16, 865–871 (1995)
14. Nilsson, D.: An algorithm for finding the most probable configurations of discrete variables that are specified in probabilistic expert systems. MSc.Thesis, University of Copenhagen, Copenhagen, Denmark (1994)
15. Shimony, S.E.: Explanation, irrelevance and statistical independence. In: *Proc. of the National Conf. in Artificial Intelligence*, pp. 482–487 (1991)
16. Neapolitan, R.E.: *Probabilistic Reasoning in Expert Systems. Theory and Algorithms*. Wiley Interscience, New York (1990)
17. Park, J.D., Darwiche, A.: Complexity results and approximation strategies for MAP explanations. *Journal of Artificial Intelligence Research* 21, 101–133 (2004)
18. de Campos, L.M., Gámez, J.A., Moral, S.: On the problem of performing exact partial abductive inference in Bayesian belief networks using junction trees. In: Bouchon-Meunier, B., Gutierrez, J., Magdalena, L., Yager, R.R. (eds.) *Technologies for Constructing Intelligent Systems 2: Tools*, pp. 289–302. Springer, Heidelberg (2002)
19. Park, J.D., Darwiche, A.: Solving MAP exactly using systematic search. In: *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence (UAI 2003)*, pp. 459–468 (2003)
20. Salmerón, A., Cano, A., Moral, S.: Importance sampling in Bayesian networks using probability trees. *Computational Statistics and Data Analysis* 34, 387–413 (2000)
21. de Campos, L.M., Gámez, J.A., Moral, S.: Partial abductive inference in Bayesian networks by using probability trees. In: Camp, O., Filipe, J., Hammoudi, S., Piatini, M. (eds.) *Enterprise Information Systems*, vol. V, pp. 146–154. Kluwer Academic Publishers, Dordrecht (2004)
22. Gámez, J.A.: *Inferencia abductiva en redes causales (Abductive inference in casual networks)*. Doctoral thesis, Dpto. de Ciencias de la Computación e I.A. Universidad de Granada (June 1998)
23. Chajewska, U., Halpern, J.Y.: Defining explanation in probabilistic systems. In: *Proc. of 13th Conf. on Uncertainty in Artificial Intelligence (UAI 1997)*, pp. 62–71 (1997)
24. Shimony, S.E.: The role of relevance in explanation I: Irrelevance as statistical independence. *International Journal of Approximate Reasoning* 8, 281–324 (1993)
25. Henrion, M., Druzdzal, M.J.: Qualitative propagation and scenario-based schemes for explaining probabilistic reasoning. In: Bonissone, P.P., Henrion, M., Kanal, L.N., Lemmer, J.F. (eds.) *Uncertainty in Artificial Intelligence*, vol. 6, pp. 17–32. Elsevier Science, Amsterdam (1991)
26. de Campos, L.M., Gámez, J.A., Moral, S.: Simplifying explanations in Bayesian belief networks. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems* 9, 461–489 (2001)
27. Flores, M.J., Gámez, J.A., Moral, S.: Abductive inference in Bayesian networks: finding a partition of the explanation space. In: Godo, L. (ed.) *ECSQARU 2005. LNCS (LNAI)*, vol. 3571, pp. 63–75. Springer, Heidelberg (2005)
28. Shenoy, P.P.: Binary join trees for computing marginals in the Shenoy-Shafer architecture. *International Journal of Approximate Reasoning* 17(2-3), 239–263 (1997)

29. Dechter, R.: Bucket elimination: A unifying framework for probabilistic inference. In: Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence (UAI 1996), pp. 211–219 (1996)
30. deKleer, J., Williams, B.C.: Diagnosing multiple faults. *Artificial Intelligence* 32(1), 97–130 (1987)
31. Consortium, E.: Elvira: An Environment for Probabilistic Graphical Models. In: Gámez, J.A., Salmerón, A. (eds.) Proceedings of the 1st European Workshop on Probabilistic Graphical Models, pp. 222–230 (2002)
32. Flores, M.J., Gámez, J.A., Moral, S.: The Independency tree model: a new approach for clustering and factorisation. In: Proceedings of the Third European Workshop on Probabilistic Graphical Models, PGM 2006, pp. 83–90 (2006)

Toward a Generalized Bayesian Network*

Dawn E. Holmes

Department of Statistics and Applied Probability, South Hall,
University of California, Santa Barbara,
CA 93106, USA

Abstract. The author's past work in this area has shown that the probability of a state of a Bayesian network, found using the standard Bayesian techniques, could be equated to the Maximum Entropy solution and that this result enabled us to find minimally prejudiced estimates of missing information in Bayesian networks. In this paper we show that in the class of Bayesian networks known as Bayesian trees, we are able to determine missing constraint values optimally using only the maximum entropy formalism. Bayesian networks that are specified entirely within the maximum entropy formalism, whether or not information is missing, are called generalized Bayesian networks. It is expected that further work will fully generalize this result.

Keywords: Bayesian networks, maximum entropy, d -separation.

PACS: 02.50.Cw, 89.70.+c, 05.70.-a, 65.40.Gr.

11.1 Introduction

One of the major drawbacks of using Bayesian networks is that complete information, in the form of marginal and conditional probabilities must be specified before the usual updating algorithms are applied. Holmes [1] has shown that when all or some of this information is missing, it is possible to determine unbiased estimates using maximum entropy. The techniques thus developed depend on the property that the probability of a state of a fully-specified Bayesian network, found using the standard Bayesian techniques, can be equated to the maximum entropy solution. A fully-constrained Bayesian network is clearly a special case, both theoretically and practically, and a general theory has yet to be provided. As a first step toward a general theory a *generalized* Bayesian network is defined as one in which some, all or none of the essential information is missing. It is then shown that missing information can be estimated using the maximum entropy formalism (MaxEnt) alone, thus divorcing these results from their dependence on Bayesian techniques.

The techniques required for the current problem are substantially different to those used previously in that, although we still use the method of undetermined multipliers, we no longer equate the joint probability distributions given by the Bayesian and maximum entropy models in order to determine the Lagrange multipliers. Two preliminary results are described here. Firstly, we extend the 2-valued work of Holmes [2] and of Markham and Rhodes [3] by developing an iterative algorithm for updating

* Re-printed with kind permission of the American Institute of Physics.

probabilities in a multivalued multiway tree, Secondly, we use the Lagrange multiplier technique to find the probability of an arbitrary state in a Bayesian tree using only MaxEnt. We begin by defining a Bayesian network.

11.2 Bayesian Networks

A Bayesian network is essentially a system of constraints; those constraints being determined by *d*-separation. Formally, a Bayesian network is defined as follows. Let:

- (i) **V** be a finite set of vertices
- (ii) **B** be a set of directed edges between vertices with no feedback loops. The vertices together with the directed edges form a directed acyclic graph $\mathbf{G} = \langle \mathbf{V}, \mathbf{B} \rangle$
- (iii) a set of events be depicted by the vertices of **G** and hence also represented by **V**, each event having a finite set of mutually exclusive outcomes
- (iv) E_i be a variable which can take any of the outcomes e_i^j of the event $i, j = 1 \dots n_i$
- (v) **P** be a probability distribution over the combinations of events, i.e. **P** consists of all possible $P \left(\bigcap_{i \in \mathbf{V}} E_i \right)$.

Let **C** be the following set of constraints:

- (2i) the elements of **P** sum to unity.
- (2ii) for each event *i* with a set of parents M_i there are associated conditional probabilities $P \left(E_i \mid \bigcap_{j \in M_i} E_j \right)$ for each possible outcome that can be assigned to E_i and E_j .
- (2iii) those independence relationships implied by *d*-separation in the directed acyclic graph.

Then $\mathbf{N} = \langle \mathbf{G}, \mathbf{P}, \mathbf{C} \rangle$ is a causal network if **P** satisfies **C**.

In a Bayesian network the property of *d*-separation identifies all the constraints as independencies and dependencies. In classical Bayesian network theory a prior distribution must be specified in order to apply the updating algorithms developed, for example, by Pearl [4] or Lauritzen and Spiegelhalter [5]. By working with the same set of constraints as those implied by *d*-separation, the MaxEnt formalism provides a means of determining the prior distribution when information is missing. The author has previously shown that the MaxEnt model with complete information is identical to the Bayesian model and has used this property to estimate the optimal prior

distribution when information is missing. We now show that the MaxEnt model is not dependent on the Bayesian model for a class of Bayesian networks.

11.3 A Generalized Bayesian Network with Maximum Entropy

Consider the knowledge domain represented by a set, K , of multivalued events a_i . Associated with each event is a variable E_v . The general state S of the causal tree is the conjunction $\bigcap_{v \in V} E_v$. A particular state is obtained by assigning some e_v^j to each E_v . It is assumed that the probability of a state is non-zero. The number of states N_S in the tree is given by:

$$N_S = \prod_{i \in V} n_i$$

where n_i is the number of values possessed by the i th event. States are numbered from $1, \dots, N_S$ and denoted by $S_i : i = 1, \dots, N_S$, and the probability of a state is denoted by $P(S_i)$. To determine a minimally prejudiced probability distribution \mathbf{P} , using the maximum entropy formalism, we maximize

$$H = - \sum_{i=1}^{N_S} P(S_i) \ln P(S_i) \tag{11.1}$$

in accordance with the constraints implied by d -separation. These constraints are given in the form of marginal or conditional probabilities that represent the current state of knowledge of the domain.

Let a sufficient set of constraints be denoted by \mathbf{C} , where each constraint $C_j \in \mathbf{C}$. Each constraint is assigned a unique Lagrange multiplier λ_j , where j represents the subscripts corresponding to the events on the associated edge. For the edge $\langle a_1, b_1 \rangle$, the Lagrange multipliers are $\lambda(b_1^1, a_1^1), \lambda(b_1^1, a_1^2), \dots, \lambda(b_1^m, a_1^p)$ where event a_1 has p outcomes and event b_1 has m outcomes. Without loss of generality we consider the constraints arising from a typical edge $\langle a_1, b_1 \rangle$ thus:

$$P(e_b^j | e_a^i) = \beta(b_j, a_i) \quad i = 1, \dots, N_S; \quad j = 1, \dots, m \tag{11.2}$$

Since \mathbf{P} is a probability distribution we also require the normalization constraint:

$$\sum_{i=1}^{N_S} P(S_i) = 1 \tag{11.3}$$

The Lagrange multiplier λ_0 is associated with the sum to unity. Applying the theory of Lagrange multipliers transforms the problem into that of maximizing:

$$F = H - \sum_{\text{all } j} \lambda_j C_j \tag{11.4}$$

By partially differentiating (11.4) with respect to $P(S_i)$ and λ_j , we see that the contribution to the expression for a maximum from H is given by:

$$-(1 + \ln P(S_i)) \quad i = 1, \dots, N_s \tag{11.5}$$

Similarly, the contribution made by each causal constraint and the sum to unity to the expression for a maximum is given by

$$- \sum_{\substack{C_j \in C \\ i=1, \dots, N_s}} \lambda_j \frac{\partial C_j}{\partial P(S_i)} = 0 \tag{11.6}$$

resulting in a combined expression:

$$-(1 + \ln P(S_i)) - \sum_{\substack{C_j \in C \\ i=1, \dots, N_s}} \lambda_j \frac{\partial C_j}{\partial P(S_i)} = 0 \tag{11.7}$$

and hence

$$P(S_i) = e^{-1} \prod_{\substack{C_j \in C \\ i=1, \dots, N_s}} \exp\left(\left(-\lambda_j\right) \frac{\partial C_j}{\partial P(S_i)}\right) \tag{11.8}$$

In order to further consider the probability of a state, as given in (11.8), we first need to transform the given constraints into expressions containing the sums of probabilities of states. These causal constraints given in (11.2) are thus expressed in the form:

$$\left(1 - \beta(b_1^j, a_1^i)\right) \sum_{x \in X} P(S_x) - \beta(b_1^j, a_1^i) \sum_{y \in Y} P(S_y) = 0 \tag{11.9}$$

where $X = \left\{ x \mid \sum_x P(S_x) = P(e_{a_1}^i e_{b_1}^j) \right\}$ and

$$Y = \left\{ y \mid \sum_y P(S_y) = \sum_{\substack{k=1 \\ k \neq j}}^{k=m} P(e_{a_1}^i e_{b_1}^k) \right\}$$

This defines a family of constraint equations for the arbitrary edge $\langle a_1, b_1 \rangle$. The root node is a special case of equations (11.2) since the information is given in the form of marginal probabilities and hence they need not be considered separately.

Substituting (11.8) into (11.9) gives:

$$(1 - \beta(b_1^j, a_1^i)) \sum_{x \in X} \prod_{C_j \in C} \exp\left((-\lambda_{b_1^j, a_1^i}) \frac{\partial C_j}{\partial P(S_x)} \right) - \beta(b_1^j, a_1^i) \sum_{y \in Y} \prod_{C_j \in C} \exp\left((-\lambda_{b_1^j, a_1^i}) \frac{\partial C_j}{\partial P(S_y)} \right) = 0 \tag{11.10}$$

Now consider the probability of the state with event a_1 instantiated with its i th outcome and event b_1 with its j th outcome, denoted by $P(S_{b_1^j, a_1^i})$. We see that when $x \in X$, $P(S_{b_1^j, a_1^i})$ contains the expression:

$$\exp\left((-\lambda(b_1^j, a_1^i))(1 - \beta(b_1^j, a_1^i)) \right)$$

Similarly, when $y \in Y$, $P(S_{b_1^j, a_1^i})$ contains the terms:

$$\exp(-\lambda(b_1^j, a_1^i)) \prod_{k=1}^{k=m-1} \exp\left((-\lambda(b_1^k, a_1^i))(-\beta(b_1^k, a_1^i)) \right)$$

Hence $P(S_{b_1^j, a_1^i})$ contains the terms

$$\exp\left((-\lambda(b_1^j, a_1^i))(1 - \beta(b_1^j, a_1^i)) \right) \prod_{\substack{k=1 \\ k \neq j}}^{k=m-1} \exp\left((-\lambda(b_1^k, a_1^i))(-\beta(b_1^k, a_1^i)) \right)$$

arising from the edge $\langle a_1, b_1 \rangle$. Re-arranging gives

$$\exp\left((-\lambda(b_1^j, a_1^i)) \right) \prod_{k=1}^{k=m-1} \exp\left(-\lambda(b_1^k, a_1^i)(-\beta(b_1^k, a_1^i)) \right)$$

Since this constraint is typical we see that for all states belonging to $X \in x$:

$$\exp\left((-\lambda(b_1^j, a_1^i))(1 - \beta(b_1^j, a_1^i)) \right) \tag{11.11}$$

and for all states belonging to $Y \in y$:

$$\exp(-\lambda(b_1^j, a_1^i)) \prod_{k=1}^{k=m-1} \exp\left((-\lambda(b_1^k, a_1^i))(-\beta(b_1^k, a_1^i)) \right) \tag{11.12}$$

From equations (11.11) and (11.12) we see that (11.10) becomes:

$$\begin{aligned} & \exp(-\lambda(b_1^j, a_1^i))(1 - \beta(b_1^j, a_1^i)) \sum_{x \in X} \prod_{C_j \in \mathbf{C} - C_{(b_1^j, a_1^i)}} \exp\left(\left(-\lambda_{b_1^j, a_1^i}\right) \frac{\partial C_j}{\partial P(S_x)}\right) - \\ & \beta(b_1^j, a_1^i) \sum_{y \in Y} \prod_{C_j \in \mathbf{C} - C_{(b_1^j, a_1^i)}} \exp\left(\left(-\lambda_{b_1^j, a_1^i}\right) \frac{\partial C_j}{\partial P(S_y)}\right) = 0 \end{aligned}$$

and hence

$$\begin{aligned} \exp(-\lambda(b_1^j, a_1^i)) = & \frac{\beta(b_1^j, a_1^i) \sum_{y \in Y} \prod_{C_j \in \mathbf{C} - C_{(b_1^j, a_1^i)}} \exp\left(\left(-\lambda_{b_1^j, a_1^i}\right) \frac{\partial C_j}{\partial P(S_y)}\right)}{\left(1 - \beta(b_1^j, a_1^i)\right) \sum_{x \in X} \prod_{C_j \in \mathbf{C} - C_{(b_1^j, a_1^i)}} \exp\left(\left(-\lambda_{b_1^j, a_1^i}\right) \frac{\partial C_j}{\partial P(S_x)}\right)} \end{aligned} \tag{11.13}$$

This expression enables us to update Lagrange multipliers using an iterative algorithm. However, as we show in the next section, we can solve for the Lagrange multipliers algebraically, thus producing a solution identical to that given in earlier papers, using techniques outside of the MaxEnt formalism. See for example, Holmes [6]

11.4 Solving for the Lagrange Multipliers: Example

For the purposes of illustration we consider a three valued causal binary tree with three nodes A, B and C. Let

$$E_a = \{e_a^1 e_a^2 e_a^3\}, E_b = \{e_b^1 e_b^2 e_b^3\}, E_c = \{e_c^1 e_c^2 e_c^3\}$$

denote the outcomes of events a, b and c respectively, which are mutually exclusive and collectively exhaustive. The required information, given by conditional probabilities associated with each outcome, is as follows:

$$\sum_{i=0}^{26} P(S_i) = 1 \quad (\text{constraint 0})$$

$$P(e_a^i) = \alpha(a_i); \quad i = 1, 2; \quad (\text{constraints 1 and 2})$$

$$P(e_b^j | e_a^i) = \beta(b_j a_i), \quad P(e_b^j | e_a^i) = \beta(b_j a_i); \quad i = 1, 2, 3; \quad j = 1, 2; \quad (\text{constraints 3 - 8})$$

$$P(e_c^j | e_a^i) = \beta(c_j a_i), \quad P(e_c^j | e_a^i) = \beta(c_j a_i); \quad i = 1, 2, 3; \quad j = 1, 2; \quad (\text{constraints 9 - 14})$$

$$\tag{11.14}$$

This system can be in any of 27 states, labeled 0-26, as follows:

$S_0 : e_a^1 e_b^1 e_c^1$	$S_1 : e_a^1 e_b^1 e_c^2$	$S_2 : e_a^1 e_b^1 e_c^3$	$S_3 : e_a^1 e_b^2 e_c^1$	$S_4 : e_a^1 e_b^2 e_c^2$	$S_5 : e_a^1 e_b^2 e_c^3$
$S_6 : e_a^1 e_b^3 e_c^1$	$S_7 : e_a^1 e_b^3 e_c^2$	$S_8 : e_a^1 e_b^3 e_c^3$			

The remaining states are similarly defined but with $E_a = e_a^2$ for states 9 – 17 and $E_a = e_a^3$ for states 18 – 26. Each constraint in (11.14) can be expressed in terms of state probabilities, as in (11.9). For example, constraint 3 gives:

$$(1 - \beta(b_1 a_1)) \sum_{i=0,1,2} P(S_i) - \beta(b_1 a_1) \left(\sum_{i=3,4,5} P(S_i) + \sum_{i=6,7,8} P(S_i) \right) = 0 \quad (11.15)$$

In (11.14), sets X and Y as defined in (11.9), contain states 3,4,5 and 6,7,8 respectively. Using the equation for probability of a state given by (11.6) together with (11.15) enables us to find the values of all the Lagrange multipliers. Expanding (11.15) and simplifying gives an expression for $\exp(-\lambda_3)$ in terms of known information, together with certain unknown Lagrange multipliers thus:

$$\exp(-\lambda_3) = \left(\frac{\beta(b_1 a_1)}{1 - \beta(b_1 a_1)} \right) \times \left(\frac{1 + \exp(-\lambda_4) + \exp(-\lambda_5) + \exp(-\lambda_6) + \exp(-\lambda_4) \exp(-\lambda_5) + \exp(-\lambda_4) \exp(-\lambda_6)}{1 + \exp(-\lambda_5) + \exp(-\lambda_6)} \right) \quad (11.16)$$

Following the same procedure but with

$$(1 - \beta(b_2 a_1)) \sum_{i=3,4,5} P(S_i) - \beta(b_2 a_1) \left(\sum_{i=0,1,2} P(S_i) + \sum_{i=6,7,8} P(S_i) \right) = 0 \quad (11.17)$$

leads to

$$\exp(-\lambda_4) = \left(\frac{\beta(b_2 a_1)}{1 - \beta(b_2 a_1)} \right) \times \left(\frac{1 + \exp(-\lambda_3) + \exp(-\lambda_5) + \exp(-\lambda_6) + \exp(-\lambda_3) \exp(-\lambda_5) + \exp(-\lambda_3) \exp(-\lambda_6)}{1 + \exp(-\lambda_5) + \exp(-\lambda_6)} \right) \quad (11.18)$$

Using equations (1.16) and (1.17) we find, by factorization and substitution, that:

$$\exp(-\lambda_3) = \left(\frac{\beta(b_1 a_1)}{1 - \beta(b_1 a_1)} \right) \left(1 + \frac{\beta(b_2 a_1)}{1 - \beta(b_2 a_1)} (1 + \exp(-\lambda_3)) \right) \quad (11.19)$$

hence

$$\exp(-\lambda_3) = \frac{\beta(b_1 a_1)}{\beta(b_1 a_1)\beta(b_2 a_1) + (1 - \beta(b_2 a_1))(1 - \beta(b_1 a_1))} \quad (11.20)$$

and so

$$\exp(-\lambda_3) = \frac{\beta(b_1 a_1)}{\beta(b_2 a_1)} \quad (11.21)$$

The remaining Lagrange multipliers are found similarly, and so the probability of each state can be determined.

11.5 Remarks

For the class of Bayesian networks discussed here, the non-linear independence constraints implied by d -separation are preserved by the maximum entropy formalism and do not need to be explicitly stated.

Having shown how to find the Lagrange multipliers and thus the probability of each state, methods previously developed by Holmes and Rhodes [1] can be used to determine missing information since these depend only on the maximum entropy formalism. We have seen in this paper how to derive expressions for estimating missing information in tree-like Bayesian networks without equating the maximum entropy and Bayesian models.

The next step in the current project will be to develop the theory required to deal with the non-linear constraints inherent in singly connected networks without recourse to methods outside of the maximum entropy formalism.

References

1. Holmes, D.E., Rhodes, P.C.: Reasoning with Incomplete Information in a Multivalued Multiway Causal Tree Using the Maximum Entropy Formalism. *International Journal of Intelligent Systems* 13(9), 841–859 (1998)
2. Holmes, D.E.: Maximizing Entropy for Inference in a Class of Multiply Connected Networks. In: *The 24th Conference on Maximum Entropy and Bayesian methods*, American Institute of Physics (2004)
3. Markham, M.J., Rhodes, P.C.: Maximizing Entropy to deduce an Initial Probability Distribution for a Causal Network. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems* 7(1), 63–80 (1999)
4. Pearl, J.: *Probabilistic Reasoning in Intelligent Systems*. In: *Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Francisco (1988)
5. Lauritzen, S.L., Spiegelhalter, D.J.: Local Computations with Probabilities on Graphical Structures and their Applications to Expert Systems. *J. Royal Statist. Soc. B* 50(2), 154–227 (1988)
6. Holmes, D.E.: Efficient Estimation of Missing Information in Multivalued Singly Connected Networks Using Maximum Entropy. In: von der Linden, W., Dose, V., Fischer, R., Preuss, R. (eds.) *Maximum Entropy and Bayesian Methods*, pp. 289–300. Kluwer Academic, Dordrecht (1999)

A Survey of First-Order Probabilistic Models

Rodrigo de Salvo Braz*, Eyal Amir, and Dan Roth

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801

Summary. There has been a long standing division in Artificial Intelligence between logical and probabilistic reasoning approaches. While probabilistic models can deal well with inherent uncertainty in many real-world domains, they operate on a mostly propositional level. Logic systems, on the other hand, can deal with much richer representations, especially first-order ones, but treat uncertainty only in limited ways. Therefore, an integration of these types of inference is highly desirable, and many approaches have been proposed, especially from the 1990s on. These solutions come from many different subfields and vary greatly in language, features and (when available at all) inference algorithms. Therefore their relation to each other is not always clear, as well as their semantics. In this survey, we present the main aspects of the solutions proposed and group them according to language, semantics and inference algorithm. In doing so, we draw relations between them and discuss particularly important choices and tradeoffs.

For decades after the field of Artificial Intelligence (AI) was established, its most prevalent form of representation and inference was logical, or at least symbolic representations that were in a deeper sense equivalent to a fragment of logic. While highly expressive, this type of model lacked a sophisticated treatment of *degrees* of uncertainty, which permeates real-world domains, especially the ones usually associated with intelligence, such as language, perception and common sense reasoning.

In time, probabilistic models became an important part of the field, incorporating probability theory into reasoning and learning AI models. Since the 1980s the field has seen a surge of successful solutions involving large amounts of data processed from a probabilistic point of view, applied especially to Natural Language Processing and Pattern Recognition.¹

* Currently at the Computer Science Division of University of California, Berkeley.

¹ Strictly speaking, this tendency has not been only probabilistic, including machine learning methods such as neural networks that did not claim to be modeling probabilities. However, a link to probabilities can usually be found and the methods are used in similar ways.

This success, however, came with a price. Typically, probabilistic models are less expressive and flexible than logical or symbolic systems. Usually, they involve propositional, rather than first-order representations. When required, more expressive, higher level representations are obtained by ad hoc manipulations of lower level, propositional systems.

Starting in the 1970s but having greatly increased from the 1990s on, a line of research sought to integrate those two important modes of reasoning. In this chapter we give a survey of this research, and try to show some general lines separating different approaches.

We have roughly divided this research in different stages. The 1970s and 1980s saw great interest in expert systems [1, 2]. As these systems were applied to real-world domains, coping with uncertainty became more desirable, giving rise to the certainty factors approach, which uses rules with attached numbers (representing degrees of certainty) that get propagated to conclusions during inference.

Certainty factors systems did not have clear semantics, and often produced surprising and nonintuitive results [3]. The search for clearer semantics for rules with varying certainty gave rise, among other things, to approaches such as Bayesian Networks. These however were essentially propositional, and thus had much less expressivity than logic systems.

The search for clear semantics of probabilities in logic systems resulted in works such as Nilsson [4], Bacchus [5] and Halpern [6], which laid out the basic theoretic principles supporting probabilistic logic. These works, however, did not include efficient inference algorithms.

Works aiming at efficient inference algorithms for first-order probabilistic inference (FOPI) can be divided in two groups, which Pearl [3] calls *extensional* and *intensional* systems. In the first one, statements in the language are more procedural in nature, standing for licenses for propagating truth values that have been generalized from true or false to a gray scale of varying degrees of certainty. In the second group, statements place restrictions on a probability distribution on possible worlds. They do not directly correspond to computing operations, nor can they typically be taken into account without regard to other rules (that is, inference is not completely modular). Efficient algorithms have to be devised for these languages that preserve their semantics while doing better than considering the entire model at every step.

Among intensional models, there are further divisions regarding the type of algorithm proposed. One group proposes inference rules similar to the ones used in first-order logic inference (for example, modus ponens). A second one computes, in more or less efficient manners, the possible derivations of a query given a model. A third one uses sampling to answer queries about a model. A fourth and more prevalent group constructs a (propositional) graphical model (Bayesian or Markov networks, for example) that answers queries, and uses general graphical model inference algorithms for solving them. Finally, a fifth one proposes *lifted* algorithms that directly operate on first-order representations in order to derive answers to queries.

We now present these stages in more detail.

12.1 Expert Systems and Certainty Factors

Expert systems are based on rules meant to be applied to existing facts, producing new facts as conclusions [1]. Typically, the context is a deterministic one in which facts and rules are assumed to be certain. Uncertainties from real-world applications are dealt with during the modeling stage where necessary (and often heavy-handed) simplifications are performed.

Certainty factors were introduced for the purpose of allowing uncertain rules and facts, making for more direct and accurate modeling. A rule $(A \leftarrow B) : c_1$, with $c_1 \in [0, 1]$, indicates that we can conclude A with a degree of certainty of $c_1 \times c_2$, if B is known to be true with a degree of certainty $c_2 \in [0, 1]$. Given a collection of rules and facts, inference is performed by propagating certainties in this fashion. There are also combination rules for the cases when more than one rule provide certainty factors for the same literal.

A paradigmatic application of certainty factors is the system MYCIN [7], an expert system dedicated to diagnosing diseases based on observed symptoms. Clark & McCabe [8] describe how to use Prolog with predicates containing an extra argument representing its certainty and being propagated accordingly. Shapiro [9] describes a Prolog interpreter that does the same but in a way implicit in the interpreter and language, rather than as an extra argument.

One can see that certainty factors have a probabilistic flavor to them, but formally they are not taken to be probabilistic. This is for good reason: should we interpret them as probabilities, results would be inconsistent with probability theory. Heckerman [10] and Lucas [11] discuss situations in which certainty factor computations can and cannot be correctly interpreted probabilistically. One reason they cannot is the incorrect treatment of bidirectional inference: two certainty factor rules $(A \leftarrow B) : c_1$ and $B : c_2$ imply nothing about inference from A to B , while $P(A|B)$ and $P(B)$ do place constraints on $P(B|A)$. These problems are further discussed in Pearl [3].

12.2 Probabilistic Logic Semantics

The semantic limitations of certainty factors is one of the motivations for defining precise semantics for probabilistic logics, but such investigations date from at least as far back as Carnap [12].

One of the most influential AI works in this regard is Nilsson [4] (a similar approach is given by Hailperin [13]). Nilsson establishes a systematic way of determining the probabilities of logic sentences in a query set, given the probabilities of logical sentences in an evidence set. To be more precise, the method determines *intervals* of probabilities to the query sentences, since in principle the evidence set may be consistent with an entire range of point probabilities for them. For example, knowing that A is true with probability 0.2 and B with probability 0.6 means that $A \wedge B$ is true with probability in $[0, 0.2]$, depending on whether A and B are mutually exclusive, or $A \rightarrow B$, or anything in between.

Given a set of sentences L , Nilsson considers the equivalence classes of possible worlds that assign the same truth values to the sentences in L (that is, as far as L is concerned, all possible worlds in the same class are the same). Formally, Nilsson's system is based on the following linear problem:

$$\begin{aligned} \Pi &= VP \\ 0 &\leq \Pi_j \leq 1 \\ 0 &\leq P_i \leq 1 \\ \sum_i P_i &= 1 \end{aligned}$$

where Π is the vector of probabilities of sentences in both query and evidence sets, P the vector of probabilities of possible worlds equivalence classes, and V is a matrix with $V_{ij} = 1$ if sentence j is true in possible world set i , and 0 otherwise. The probabilities of sentences in the knowledge base are incorporated as constraints in this system as well, and linear programming techniques can be used to determine the probability of novel sentences. However, as Nilsson points out, the problem becomes intractable even with a modest number of sentences, since all possible worlds equivalence classes need to be enumerated and this is an intractable problem. Therefore this framework cannot be directly used in practice.

Placing the probabilities on the possible worlds, as does Nilsson, makes it easy to express subjective probabilities such as “Tweety flies with probability 0.9” (that is, the sum of probabilities of all possible worlds in which Tweety flies is 0.9). However, probabilistic knowledge can also express statistical facts about the domain such as “90% of birds fly” (which says that, in each possible world, 90% of birds fly). Bacchus [5] provides an elaborate probabilistic logic semantics that includes both types of probabilistic knowledge, making it possible to use both statements above, as well as statements mixing them, such as “There is a probability of 0.8 that 90% of birds fly.” He also discusses the interplay between the two types, namely the question of when it is correct to use the fact that “90% of birds fly” in order to assume that “a randomly chosen bird flies with probability 0.9,” a topic that has both formal and philosophical aspects. Halpern [6] elaborates on the axiomatization of Bacchus, taking probabilities to be real numbers (Bacchus did not), and is often cited as a reference for this semantics with two types of probabilities. In subsequent work, the subjective type probability has been much more developed and used, and is also the type involved in propositional graphical models.

Fagin, Halpern, Meggido [14] present a logic to reason *about* probabilities, including their addition and multiplication by scalars. Other works discussing the semantics of probabilities on first-order structures are [15, 16, 17].

12.3 Extensional Approaches

Somewhat parallel to the works on the semantics of probabilistic logic, a different line of research proposed logic reasoning systems incorporating uncertainty in

the explicit form of probabilities (as opposed to certainty factors). These systems often stem from the fields of logic programming and deductive databases, and fit into the category described by [3] as *extensional* systems, that is, systems in which rules work as “procedural licenses” for a computation step instead of a constraint on possible probability distributions. Most of these systems operate on a collection of rules or clauses that propagate generalized truth values (typically, a value or interval in $[0, 1]$).

Kiefer and Li [18] provide a probabilistic interpretation and a fixpoint semantics to Shapiro [9]. Wüthrich [19] elaborates on their work, taking into account partial dependencies between clauses. For example, if each of them atoms a, b and c has a prior probability of 0.5 and we have two rules $p \leftarrow a \wedge b$ and $p \leftarrow b \wedge c$, Kiefer and Li will assume the rules independent and assign a probability $0.25 + 0.25 - 0.25 * 0.25 = 0.4375$ to p . Wüthrich’s system, however, takes into account the fact that b is shared by the clauses and computes instead $0.25 + 0.25 - 0.5^3 = 0.375$ (that is, it avoids double counting of the case where the two rules fire at the same time, which occurs only when the three atoms are true at once).

One of the most influential works within the extensional approach is Ng and Subrahmanian [20]. Here, a logic programming system uses generalized truth values in the form of intervals of probabilities. They define probabilistic logic program as sets of *p-clauses* of the form

$$A : \mu \leftarrow F_1 : \mu_1 \wedge \dots \wedge F_n : \mu_n,$$

where A is an atom, F_1, \dots, F_n are basic formulas (conjunctions or disjunctions) and μ, μ_1, \dots, μ_n are probability intervals. A clause states that if the probability of each formula F_i is in μ_i , then the probability of A is in μ . For example, the clause

$$path(X, Y) : [0.8, 0.95] \leftarrow a(X, Z) : [1, 1] \wedge path(Z, Y) : [0.85, 1]$$

states that, if $a(X, Z)$ is certain (probability in interval $[1, 1]$ and therefore 1) and $path(Z, Y)$ has probability in $[0.85, 1]$, then $path(X, Y)$ has probability in $[0.8, 0.95]$. Probabilities of basic formulas F_i are determined from the probability intervals of their conjuncts (or disjuncts) by taking into account the possible correlations between them (similarly to what Nilsson does). The authors present a fixpoint semantics where clauses are repeatedly applied and probability intervals successively narrowed up to convergence. They also develop a model theory determining what models (sets of distributions on possible worlds) satisfy a probabilistic logic program, and a refutation procedure for querying a program.

Lakshmanan and Sadri [21] propose a system similar to Ngo and Subrahmanian, while keeping track of both the probability of each atom as well of its negation. Additionally, it uses configurable independence assumptions for different clauses, allowing the user to declare whether two atoms are independent, mutually exclusive, or even the lack of an assumption (as in Nilsson). Lakshmanan [22] separates the qualitative and quantitative aspects of probabilistic logic. Dependencies between atoms are declared in terms of the boolean truth

values of a set of *support* atoms. Only later is a distribution assigned to the support atoms, consequently defining distributions on the remaining atoms as well. The main advantage of the approach is the possibility of investigating different total distributions, based on distributions on the support set, without having to recalculate the relationship between atoms and support set. The algorithms works in ways similar to Ngo and Haddawy [23] and Lakshmanan and Sadri [21], but propagates support set conditions rather than probabilities. Support sets are also a concept very similar to the *hypotheses* used in Probabilistic Abduction by Poole [24] (see next section).

12.4 Intensional Approaches

We now discuss intensional approaches to probabilistic logic languages, where statements (often in the form of rules) are interpreted as restrictions on a globally defined probability distribution. This probability distribution is over all possible worlds or, in other words, on assignments to the set of all possible random variables in the language. Statements typically pose constraints in the form of conditional probabilities, and often also as conditional independence relations. (As mentioned in Sect. 12.2, another possibility would be *statistical* constraints, but this has not been explored in any works to our knowledge.)

The algorithms in intensional approaches, when available, are arguably more complex than extensional approaches, since their steps do not directly correspond to the application of rules in the language and need to be consistent with the global distribution while being as local as possible (for efficiency reasons).

We cover five different types of intensional approaches: deduction rules, exhaustive computation of derivations, sampling, Knowledge Based Model Construction (KBMC) and Lifted inference.

12.4.1 Deduction Rules

Classical logic deduction systems often work by receiving a model specified in a particular language and using deduction rules to derive new statements (guaranteed to be true) from subsets of previous statements. Some work has been devoted to devising similar systems when the language is that of probabilistic logic.

This method is particularly challenging in probabilistic systems because probabilistic inference is not as modular as classical logical inference. For example, while the logical knowledge of $A \rightarrow B$ allows us to deduce B given that $A \wedge \varphi$ is true for any formula φ , knowing $P(B|A)$ in itself does not tell us *anything* about $P(B|A \wedge \varphi)$. In principle, one needs to consider *all* available knowledge when establishing the conditional probability of B . Classical logic reasoning shows a modularity that is harder to achieve in a probabilistic setting.

One way of making probabilistic inference more modular is to use knowledge about conditional independencies between random variables. If we know that B is independent of any other random variable given A , then we know that $P(B|A \wedge \varphi)$

is equal to $P(B|A)$ for any φ . This has been the approach of graphical models such as Bayesian and Markov networks [3], where independencies are represented by the structure of a graph over the set of random variables.

The computation steps of specific inference algorithms for graphical models (such as Variable Elimination [25]) could be cast as deduction rules, much like in classical logic. However this is not traditionally done, mostly because inference rules are typically described in a logic-like language and graphical models are not. When dealing with a *first-order* probabilistic logic language, however, this approach becomes more natural.

Luckasiewicz [26] uses inference rules for solving trees of probabilistic conditional constraints over basic events. These trees are similar to Bayesian networks, with each node being a random variable and each edge being labeled by a conditional probability table. However, these trees are not meant to encode independence assumptions. Besides, conditional probabilities can also be specified in intervals.

Frisch and Haddawy [27] present a set of inference rules for probabilistic propositional logic with interval probabilities. They characterize it as an anytime system since inference rules will increasingly narrow those intervals. They also provide more modular inference by allowing statements on conditional independencies of random variables, which are used by certain rules to derive statements based on local information.

Koller and Halpern [28] investigate the use of independence information for FOPI based on inference rules. They use this notion to discuss the issue of *substitution* in probabilistic inference. While substitution is fundamental to classical logic inference, it is not sound in general in a probabilistic context. For example, inferring $P(q(A)) = \frac{1}{3}$ given $\forall P(q(X)) = \frac{1}{3}$ is not sound. Consider three possible worlds w_1, w_2, w_3 containing the three objects o_1, o_2, o_3 each, where $q(o_i)$ is 1 in w_i and 0 otherwise. If each possible world has a probability $\frac{1}{3}$ of being the actual world, then $\forall P(q(X)) = \frac{1}{3}$ holds. However, if A refers to o_i in each w_i , then $P(q(A)) = 1$. While this problem can be solved by requiring constants to be rigid designators (that is, each of them refers to the same object in all worlds), the authors argue that this is too restrictive. Their solution is to use information on independence. They show that when the statements $\forall P(q(X)) = \frac{1}{3}$ and $x = A$ are independent, one can derive $P(q(A)) = \frac{1}{3}$. Finally, they discuss the topic of using statistical probabilities as a basis for subjective ones (the two types discussed by Bacchus [5] and Halpern [6]) based on independencies.

12.4.2 Exhaustive Computation of Derivations

Another type of intensional system is the one in which the available algorithms exhaustively compute the set of derivations or proofs for a query, in the same way proofs are found for queries in logic programming. However, while in logic programming it is often only necessary to find one proof for a certain query, in probabilistic models all proofs will typically influence the query's result, and therefore need to be computed.

Riezler [29] presents a probabilistic account of Constraint Logic Programs (CLPs) [30]. In regular logic programming, the only constraints over logical variables are equational constraints coming from unification. CLPs generalize this by allowing other constraints to be stated over those variables. These constraints are managed by special-purpose constraint solvers as the derivation proceeds, and failure in satisfying a constraint determines failure of the derivation. Probabilistic Constraint Logic Programs (PCLPs) are a stochastic generalization of CLPs, where clauses are annotated with a probability and chosen for the expansion of a literal according to that probability, among the available clauses with matching heads. The probability of a derivation is determined by the product of probabilities associated to the stochastic choices. In fact, PCLPs are a generalization of Stochastic Context-Free Grammars (SCFGs) [31], the difference between them being that PCLP symbols have arguments in the form of logical variables with associated constraints while grammar symbols do not. For this reason, PCLP derivations can fail while SCFGs will always succeed. This presents a complication for PCLP algorithms because the probability has to be normalized with respect to the sum of *successful* derivations only. It also makes the use of efficient dynamic programming techniques such as the inside-outside algorithm [32] not adequate for PCLPs, forcing us to compute all possible derivations of a query. Riezler focuses on presenting an algorithm for learning the parameters of a PCLP from incomplete data, in what is a generalization of the Baum-Welch algorithm for HMMs [33].

Stochastic Logic Programs [34, 35] are very similar to PCLPs, restricting themselves to regular logic programming (e.g., Prolog). This line of work is more focused on the development of an actual system on top of a Prolog interpreter and to be used with Inductive Logic Programming techniques such as Progol [36]. Like Riezler, in [35] Cussens develops methods for learning parameters of SLPs using Improved Iterative Scaling [37] and the EM algorithm [38].

Lucasiewicz [39] presents a form of Probabilistic Logic Programming that complements Nilsson's [4] approach. Nilsson considers all equivalence classes of possible worlds with respect to the given knowledge and builds a linear program in order to assign probabilities to sentences. Lucasiewicz essentially does the same by using logic programming for both determining the the equivalence classes and the linear program.

Baral et al. [40] use answer set logic programming to implement a powerful probabilistic logic language. Its distinguishing feature is the possibility of specifying observations and actions, with their corresponding implications with respect to causality, as studied by Pearl [41]. However, the implementation, using answer set Prolog, depends on determining all answer sets.

12.4.3 Sampling Approaches

Because building all derivations of a query given a program is very expensive, approximation solutions become an attractive alternative.

Sato [42] presents PRISM, a full-featured Prolog interpreter extended with probabilistic switches that can be used to encode probabilistic rules and facts.

These switches are special built-in predicates that randomly succeed or not, following a specific probability distribution. They can be placed in the body of a clause, which in consequence will succeed or not with the same distribution (when the rest of the body succeeds). Therefore, multiple executions of the program will yield different random results that can be used as samples. A query can then be answered by multiple executions which sample its possible outcomes. Sato also provides a way of learning the parameters of the switches by using a form of the EM algorithm [43].

BLOG [44] is a first-order language allowing the specification of a generative model on a first-order structure. It is similar in form to BUGS [45], a specification language for propositional generative models. The main distinction of BLOG is its open world assumption; it does not require that the number of objects in the world be set a priori, using instead a prior on this number and also keeping track of identities of objects with different names. BLOG computes queries by sampling over possible worlds.

12.4.4 Knowledge Based Model Construction

We now present the most prominent family of models in the field of FOPI models, Knowledge Based Model Construction (KBMC). These approaches work by generating a propositional graphical model from a first-order language specification that answers the query at hand. This construction is usually done in a way specific to the query, ruling out irrelevant portions of the graph so as to increase efficiency.

Many KBMC approaches use a first-order logic-like specification language, but some use different languages such as frame systems, parameterized fragments of Bayesian networks, and description logics. Some build Bayesian networks while others prefer Markov networks (and in one case, Dependency Networks [46]).

While KBMC approaches try to prune sections of underlying graphical models which are irrelevant to the current query, there is still potentially much wasted computation because they may replicate portions of the graph which require essentially identical computations. For example, a problem may involve many employees in a company, and the underlying graphical model will contain a distinct section with its own set of random variables for each of them (representing their properties), even though all these sections have essentially the same structure. Often the same computation will be repeated for each of those sections, while it is possible to perform it only once in a generalized form. Avoiding this waste is the object of Lifted First-Order Probabilistic Inference [47, 48], discussed in Sect. 12.4.5.

The most commonly referenced KBMC approach is that of Breese [49, 50], although Horsch and Poole [51] had presented a similar solution a year before. [49] defines a probabilistic logic programming language, with Horn clauses annotated by probabilistic dependencies between the clause's head and body. Once a query is presented, clauses are applied to it in order to determine the probabilistic dependencies relevant to it. These dependencies are then used to form a Bayesian network. Backward inference will generate the causal portion of the

network relative to the query; forward inference creates the diagnostic part. The construction algorithm uses the evidence in order to decide when to stop expanding the network – there is no need to generate portions that are d-separated from the query by the evidence. In fact, this work covers not only Bayesian networks, but influence diagrams as well, including decision and utility value nodes.

There are many works similar in spirit to [49] and differing only in some details; for example, the already mentioned Horsch and Poole [51], which also uses mostly Horn clauses (it does allow for universal and existential quantifiers over the entire clause body though) as the first-order language. One distinction in this work, however, is the more explicit treatment of the issue of *combination functions*, used to combine distributions coming from distinct clauses with the same head. One example of a combination function is noisy-or [3], which assumes that the probability provided by a single clause is the probability of it making the consequent true regardless of the other clauses. Suppose we have clauses $A \leftarrow B$ and $A \leftarrow C$ in the knowledge base, the first one dictating a probability 0.8 for A when B is true and the second one dictating a probability 0.7 for A when C is true. Then the combination function noisy-or builds a Conditional Probability Table (CPT) with B and C as parents of A , with entries $P(A|B, C) = \{1 - 0.2 \times 0.3, 1 - 0.8 \times 0.3, 1 - 0.2 \times 0.7, 1 - 0.8 \times 0.7\} = \{0.94, 0.76, 0.86, 0.47\}$ for $\{(B = \top, C = \top), (B = \perp, C = \top), (B = \top, C = \perp), (B = \perp, C = \perp)\}$, respectively.

In first-order models, noisy-or and other combination functions are especially useful when a random variable has a varying number of parents, which makes its CPT impossible to represent by a fixed-dimensions table. A clause $p \leftarrow q(X)$, for example, determines that p depends on all instantiations of $q(X)$, that is, all instantiations of $q(X)$ are parents of p . However, the number of such instantiations depends on how many values X can take. Without knowing this number, the only way of having a general specification of p 's CPT is to have a combination function on the instantiations of $q(X)$. In fact, even when this number is known it may be convenient to represent the CPT with a combination function for compactness sake.

Charniak and Goldman [52] expand a deductive database and truth maintenance system (TMS) in order to define a language for constructing Bayesian networks. The Bayesian networks come from the data-dependency network maintained by the TMS system, which is annotated with probabilities. There is also a notion of combination functions. The authors choose not to expand logical languages, justifying this choice by arguing that logic and probability do not correspond perfectly, the first being based on implication while the second on conditioning.

Poole [24] defines Probabilistic Abduction, a probabilistic logic language aimed at performing abduction reasoning. Probabilities are defined only for a set of predicates, called *hypotheses* (which is reminiscent of the support set in [22]), while the clauses themselves are deterministic. When a problem has naturally dependent hypotheses, one can redefine them as regular predicates and invent a new hypothesis to explain that dependence. While deterministic clauses

can seem too restrictive, one can always get the effect of probabilistic rules by using hypotheses as a condition of the rule (like switches in Sato's PRISM [42]). The language also assumes that the bodies of clauses with the same head are mutually exclusive, and again this is not as restrictive as it might seem since clauses with non-mutually exclusive bodies can be rewritten as a different set of clauses satisfying this. As in other works in this section, the actual computation of probabilities is based on the construction of a Bayesian network. In [53], Poole extends Probabilistic Abduction for decision theory, including both utility and decision variables, as well as negation as failure.

Glesner and Koller [54] present a Prolog-like language that allows the declaration of facts about a Bayesian network to be constructed by the inference process. The computing mechanisms of Prolog are used to define the CPTs as well, so they are not restricted to tables, but can be computed on the fly. This allows CPTs to be defined as decision trees, for example, which provides a means of doing automatic pruning of the resulting Bayesian network – if the evidence provides information enough to make a CPT decision at a certain tree node, the descendants of that node, along with parts of the network relevant to those descendants only, do not need to be considered or built. The authors focus on flexible dynamic Bayesian networks that do not necessarily have the same structure at every time slice.

Haddawy [55] presents a language and construction method very similar to [51, 49]. However, he focuses on defining the semantics of the first-order probabilistic logic language directly, and independently of the Bayesian network construction, and proceeds to use it to prove the correctness of the construction method. Breese [49] had done something similar by defining the semantics of the knowledge base as an abstract Bayesian network which does not usually get built itself in the presence of evidence, and by showing that the Bayesian network actually built will give the same result as the abstract one.

Koller and Pfeffer [56] present an algorithm for learning the probabilities of noisy first-order rules used for KBMC. They use the EM algorithm applied to the Bayesian networks generated by the model, using incomplete data. This works in the same way as the regular Bayesian network parameter learning with EM, with the difference that many of the parameters in the generated networks are in fact *instances* of the same parameter in a first-order rule. Therefore, all updates on these parameters must be accumulated in the original parameter.

Jaeger [57] defines a language for specifying a Bayesian network whose nodes are the extensions of first-order predicates. In other words, each node is the assignment to the set *all* atoms of a certain predicate. Needless to say, inference in such a network would be extremely inefficient since each node would have an extremely large number of values. However, it offers the advantage of making the semantics of the language very clear (it is just the usual propositional Bayesian network semantics – the extension of a predicate is just a propositional variable with a very large number of values). The author proposes, like other approaches here, to build a regular Bayesian network (with a random variable per ground atom) for the purpose of answering specific queries. He also presents

a sophisticated scheme for combination functions, including the possibility of their nesting.

Koller et al. [58, 59] define Probabilistic Relational Models (PRMs), a sharp depart from the logical-probabilistic models that had been proposed until then as solutions for FOPI models. Instead of adding probabilities to some logic-like language, the authors use the formalism of Frame Systems [60] as a starting point. The language of frames, similar also to relational databases, is less expressive than first-order logic, which is to the authors one of its main advantages since first-order logic inference is known to be intractable (which only gets worse when probabilities are added to the mix). By using a language that limits its expressivity to what is most needed in practical applications, one hopes to obtain more tractable inference, an argument commonly held in the Knowledge Representation community [61]. In fact, Pfeffer and Koller had already investigated adding probabilities to restricted languages in [62]. In that case, the language in question was that of description logics.

The language of Frame Systems consists of defining a set of objects described by *attributes* – binary predicates relating an object to a simple scalar value – or *relations* – binary predicates relating an object to another (or even self) object. PRMs add probabilities to frame systems by establishing distributions on attributes conditioned on other attributes (in the same object, or related object). In order to avoid declaring these dependencies for each object, this is done at a *scheme* level where classes, or template objects, stand for all instances of a class. This scheme describes the attributes of classes and the relations between them. Conditional probabilities are defined for attributes and can name the conditioning attributes via the relations needed to reach them.

As in the previous approaches, queries to PRMs are computed by generating an underlying Bayesian network. Given a collection of objects (a database *skeleton*) and the relationships between them, a Bayesian network is built with a random variable for each attribute in each object. The parents of these random variables in the network are the ones determined by the relations in the particular database, and the CPT filled with the values specified at the template level. An example of this process is shown in Fig. 12.1.

Note that the set of ancestors of attributes in the underlying network is determined by the relations from one object to another. One could imagine an attribute *rating* of an object representing a restaurant that depends on the attribute *training* of the object representing its chef (related to it by the relationship *chef*). In approaches following first-order representations, *chef* would be a binary predicate, and each of its instances a random variable. As a result, the ancestors of *rating* would be the attributes *training* of all objects potentially linked to the restaurant by the relationship *chef*, plus the random variables standing for possible pairs in the relationship *cook* itself, resulting in a large (and thus expensive) CPT. PRMs avoid this when they take data with a defined structure where the assignment to relations such as *cook* is known; in this case, the random variables in the relationship *chef* would not even be included in the network, and the attribute *rating* of each object would have a single

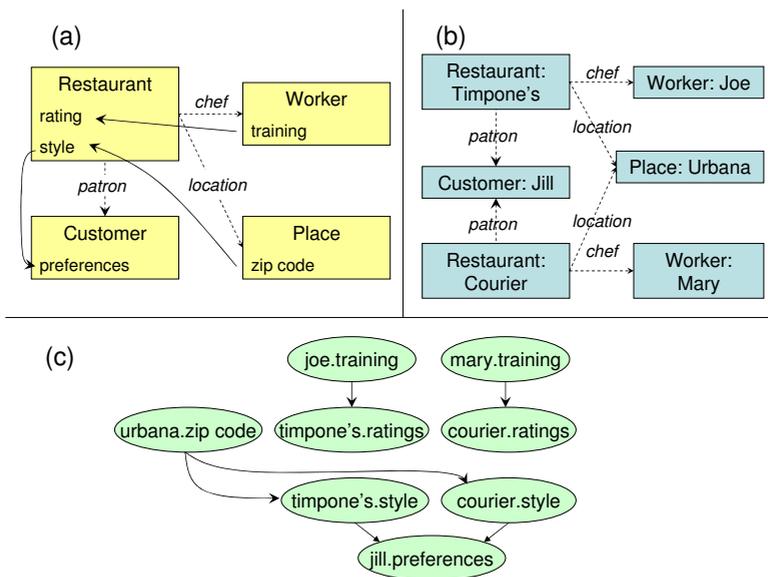


Fig. 12.1. (a) A PRM scheme showing classes of objects (rectangles), probabilistic dependencies between their attributes (full arrows) and relationships (dashed arrows). (b) A database skeleton showing a collection of objects, their classes and relationships. (c) The corresponding generated Bayesian network.

ancestor. When relationships are not fixed in advance, we have *structural uncertainty*, which was addressed by the authors in [63]. These papers have presented PRM learning of both parameters and structure (that is, the learning of the scheme level).

PRMs make use of Bayesian networks, a directed graphical model that brings a notion of causality. In relational domains it is often the case that random variables depend on each other without a clear notion of causality. Take for example a network of people linked by friendship relationships, with the attribute *smoker* for each person. We might want to state the first-order causal relationship $P(\text{smoker}(X) | \text{friends}(X, Y), \text{smoker}(Y))$ in such a model, but it would create cycles in the underlying Bayesian network (between each pair of *smoker* attributes such as *smoker(john)* and *smoker(mary)*). For this reason, Relational Markov Networks (RMNs) [64] recast PRMs so they generate undirected graphical models (Markov networks) instead of Bayesian networks. In RMNs, dependencies are stated as first-order features that get instantiated into potential function on cliques of random variables, without a notion of causality or conditional probabilities. The disadvantage of it, however, is that learning in undirected graphical models is harder than in directed ones, involving a full inference step at each expectation step of the EM algorithm.

Relational Dependency Networks (RDNs) [65] provide yet another alternative to this problem. They are the first-order version of Dependency Networks

(DNs) (Heckerman, [46]), which use conditional probabilities but do not require acyclicity. Using directed conditional probabilities avoids the expensive learning of undirected models. However, DNs have the downside of conditional probabilities being no longer guaranteed consistent with the joint probability defined by their normalized product. Heckerman shows that, as the amount of training data increases, conditional probabilities in a DN will asymptotically converge to consistency. RDNs are sets of first-order conditional probabilities which are used to generate an underlying regular dependency network. These first-order conditional probabilities are typically learned from data by relational learners (Sect. 12.5). RDNs are implemented in Proximity, a well-developed, publicly available software package.

Kersting and DeRaedt [66] introduce Bayesian Logic Programs. This work's motivation is to provide a language which is as syntactically and conceptually simple as possible while preserving the expressive power of works such as Ngo and Haddawy [23], Jaeger [57] and PRMs [58]. According to the authors, this is necessary so one understands the relationship between all these approaches, and also the fundamental aspects of FOPI models.

Fierens et al [67] define Logical Bayesian Networks (LBNs). LBNs are very similar to Bayesian Logic Programs, with the difference of having both random variables and deterministic logical literals in their language. A logic programming inference process is run for the construction of the Bayesian network, during which logical literals are used, but since they are not random variables, they are not included in the Bayesian network. This addresses the same issue of fixed relationships discussed in the presentation of PRMs, that is, when a set of relationships is deterministically known, we can create random variable nodes in the Bayesian network with significantly fewer ancestors. In the BLPs and LBNs framework, this is exemplified by a rule such as:

$$rating(X) \leftarrow cook(X, Y), training(Y) .$$

which has an associated probability, declaring that a restaurant X 's rating depends on their cook Y 's training. In Bayesian Logic Programs, the instantiations of $cook(X, Y)$ are random variables (just like the instantiations of $rating(X)$ and $training(Y)$). Therefore, since we do not know a priori which Y makes $cook(timpone, Y)$ true, $rating(timpone)$ depends on all instantiations of $cook(timpone, Y)$ and $training(Y)$ and has all of them as parents in the underlying Bayesian network. If in the domain at hand the information of $cook$ is deterministic, then this would be wasteful. We could instead determine Y such that $cook(timpone, Y)$, say $Y = joe$, and build the Bayesian network with only the relevant random variable $training(joe)$ as parent of $rating(timpone)$. This is precisely what LBNs do. In LBNs, one would define $cook$ as a deterministic literal that would be reasoned about, but not included in the Bayesian network as a random variable. This in fact is even more powerful than the PRMs

approach since it deals even with the situation where relationships are not directly given as data, but have to be reasoned about in a deterministic manner.

Santos Costa et al. [68] propose an elegant KBMC approach that smoothly leverages an already existing framework, Constraint Logic Programming (CLP). In regular logic programming, the only constraint over logical variables are equational constraints coming from unification. As explained in Sect. 12.4.2, CLP programs generalize this by allowing other constraints to be stated over those variables. These constraints are managed by special-purpose constraint solvers as the derivation proceeds, and failure in satisfying a constraint determines failure of the derivation. The authors leverage CLP by developing a constraint solver on probabilistic constraints expressed as CPTs, and simply plug it into an already existing CLP system. The resulting system can also use available logic programming mechanisms in the CPT specification, making it possible to calculate it dynamically, based on the context, rather than by fixed tables. The probabilistic constraint solver uses a Bayesian network internally in order to solve the posed constraints, so this system is also using an underlying propositional Bayesian network for answering queries. Santos Costa et al. indicate [69] as the closest approach to theirs, with the difference that the latter keeps hard constraints on Bayesian variables separate from probabilistic constraints. This allows hard constraints to be solved separately. It is also different in that it does not use conditional independencies (like Bayesian networks do), and therefore its inference is exponential on the number of random variables.

Markov Logic Networks (MLNs) [70] is a recent and rapidly evolving framework for probabilistic logic. Its main distinctions are that it is based on undirected models and has a very simple semantics while keeping the expressive

English / First-Order Logic	Clausal form	Weight
"Smoking causes cancer" $\forall X \text{ Smokes}(X) \Rightarrow \text{Cancer}(X)$	$\neg \text{Smokes}(X) \vee \text{Cancer}(X)$	1.5
"If two people are friends either both smoke or neither does" $\forall X \forall Y \text{ Fr}(X,Y) \Rightarrow (\text{Sm}(X) \Leftrightarrow \text{Sm}(Y))$	$\neg \text{Friends}(X,Y) \vee \text{Smokes}(X) \vee \text{Smokes}(Y)$	1.1

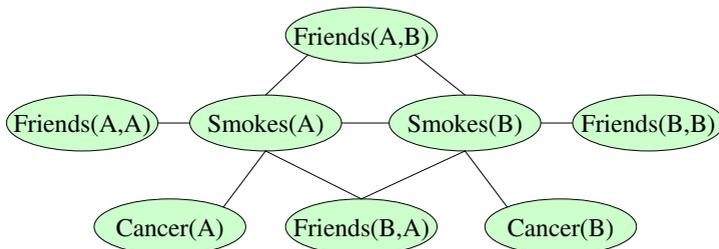


Fig. 12.2. A ground Markov network generated from a Markov Logic Network for objects Anna (A) and Bob (B) (example presented in [70])

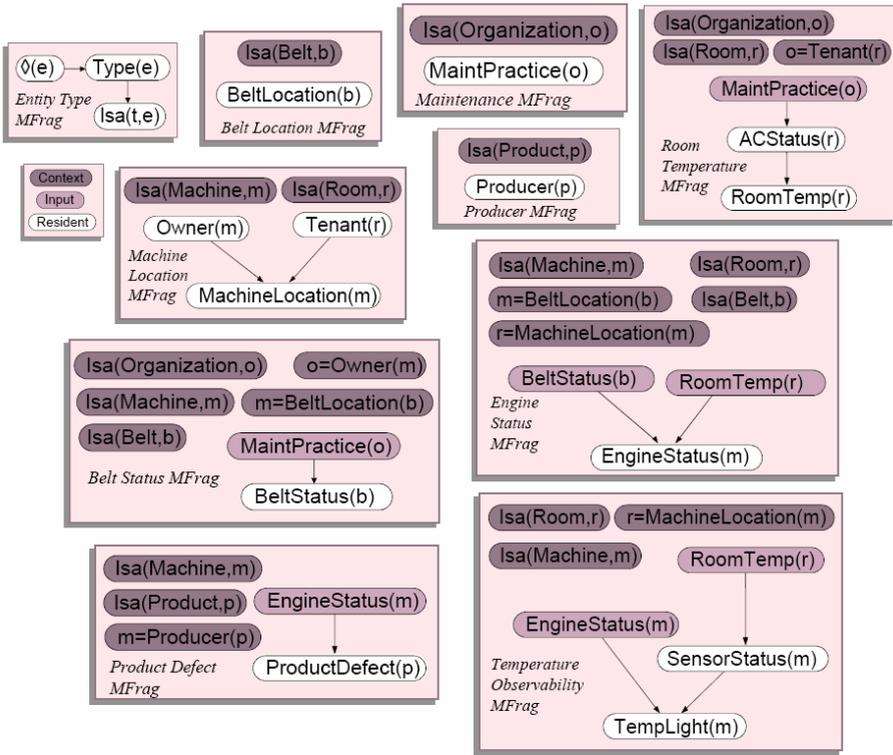


Fig. 12.3. An example of a MEBN, as shown in [72]

power of first-order logic. The downside to this is that its inference can become quite slow if complex constructs are present.

MLNs consist of a set of weighted first-order formulas and a universe of objects. Its semantics is simply that of a Markov network whose features are the instantiations of all these formulas given the universe of objects. The potential of a feature is defined as the exponential of its weight in case it is true. Figure 12.2 shows an example.

Formulas can be arbitrary first-order logic formulas, which are converted to clausal form for inference. Converting existentially quantified formulas to clausal form usually involves Skolemization, which requires uninterpreted functions in the language. Since MLNs do not include such functions, existentially quantified formulas are replaced by the disjunction of their groundings (this is possible because the domain is finite). The great expressivity of MLNs allows them to easily subsume other proposed FOPI languages. They are also a generalization of first-order logic, to which they reduce when weights are infinite.

Learning algorithms for MLNs have been presented from the beginning. Because learning in undirected models is hard, MLNs use the notion of pseudo-likelihood [71], an approximate but efficient method. When data is incomplete, EM is used.

MLNs are a powerful language and framework accompanied by well-supported software (called *Alchemy*) and which has been applied to real domains. The drawback of its expressivity is potentially very large underlying networks (for example, when existential quantification is used).

Laskey [72] presents multi-entity Bayesian networks (MEBNs), a first-order version of Bayesian networks, which rely on generalizing typical Bayesian network representations rather than a logic-like language. A MEBN is a collection of Bayesian network *fragments* involving parameterized random variables. As in the other approaches, the semantics of the model is the Bayesian network resulting from instantiating these fragments. Once they are instantiated, they are put together according to the random variables they share. A MEBN is shown in Fig. 12.3.

Laskey's language is indeed quite rich, allowing infinite models, function symbols and distributions on the parameters of random variables themselves. The work focus on defining this language rather than on the actual implementation, which is based on instantiating a Bayesian network containing the parts relevant to the query at hand. It does not provide a detailed account of this process, which can be especially tricky in the case of infinite models.

12.4.5 Lifted Inference

One of the major difficulties in KBMC approaches is that they must propositionalize the model in order to perform inference. This does not preserve the rich first-order structure present in the original model; the propositionalized version does not indicate anymore that CPTs are instantiations of the same original one, or that random variables are instantiations of an original parameterized random variable. In other words, it creates a potentially large propositional model with a great amount of redundancy that cannot be readily exploited.

Recent research on *lifted inference* [73, 47, 48] has addressed this point. A lifted inference algorithm receives a first-order specification of a probabilistic model and performs inference directly on it, without propositionalization. This can potentially yield an enormous gain in efficiency.

For example, a possible model can be formed by parameterized factors (or *parfactors*) $\phi_1(\textit{epidemic}(D))$ and $\phi_2(\textit{sick}(P, D))$ and a set of typed objects *flu*, *rubella*, and *john*, *mary* etc. The model is equivalent to a propositional graphical model formed by all possible instantiations of parfactors by the given objects, which is the set of regular factors $\phi_1(\textit{epidemic}(\textit{flu}))$, $\phi_1(\textit{epidemic}(\textit{rubella}))$, \dots , and $\phi_2(\textit{sick}(\textit{john}, \textit{flu}))$, $\phi_2(\textit{sick}(\textit{john}, \textit{rubella}))$, $\phi_2(\textit{sick}(\textit{mary}, \textit{flu}))$, $\phi_2(\textit{sick}(\textit{mary}, \textit{rubella}))$, etc.

What lifted inference does, instead of actually generating these instantiations, is to operate *directly* on the parfactors and obtain the *same* answer as the one obtained by instantiating and solving by a propositional algorithm. By operating directly on parfactors, the lifted algorithm can potentially be much more efficient, since the first-order structure is explicitly available to it. For example, suppose we want to compute the marginal of $P(\textit{epidemic}(\textit{flu}))$. Then we have

to sum out all the other random variables in the model. While a regular KBMC algorithm would instantiate them and then sum them out, a lifted inference algorithm will directly sum out the *parameterized epidemic(D)*, for $D \neq flu$, and *sick(P, D)*. The lifted elimination operation may not depend on the number of objects in the domain at all, greatly speeding up the process. The step in which an entire class of random variables is eliminated at once is possible because they all share the same structure, and this structure is explicitly available to the algorithm.

Figure 12.4 presents a simplified diagram of a lifted inference operation.

Poole [73] proposes a lifted algorithm that generalizes propositional Variable Elimination [25] but covers only some specific cases. de Salvo Braz et al. [47, 48] present a broader algorithm, called First-Order Variable Elimination (FOVE). FOVE includes Poole’s operation (which this work calls *Inversion Elimination*) a generalized version of it, called simply *Inversion*, and a second elimination operation called *Counting Elimination*. While *Inversion* does not depend on the domain size, *Counting Elimination* does, but still only exponentially less than propositionalization. The work also presents rigorous proofs of the correctness of these operations and shows how to solve the lifted version of the Most Probable Explanation (MPE) problem. While more general, FOVE still does not cover all possible cases, when it too must resort to propositionalization. When this

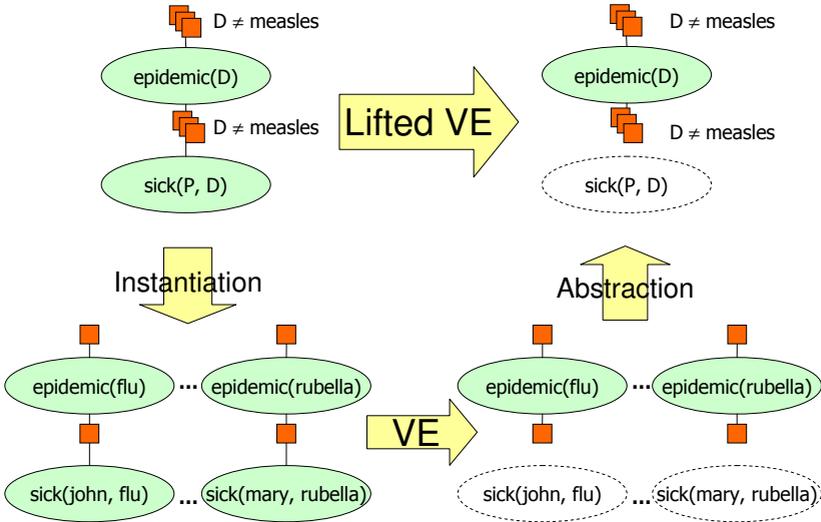


Fig. 12.4. A diagram of several possible operations involving first-order and propositional probabilistic models. The figure uses the notation of factor graphs, which explicitly shows potential functions as squares connected to their arguments. Parameterized factors are shown as piled up squares, since they compactly stand for multiple factors. Lifted inference operates solely on the first-order representation and can be much faster than propositional inference, while producing the same results.

happens, this propositionalization will be localized to the parfactors involved in the uncovered case.

Lifted FOPI is a further step towards closing the gap between logic and probabilistic inference, bringing to the latter the type of inference that does not require binding of parameters (which would be the logical variables in atoms, in logical terms), often seen in the former. However, it has its own disadvantages. It is relatively more complicated to implement, and requires a normalizing pre-processing of the model (called *shattering*) that can be very expensive. Further methods are being developed to circumvent these difficulties.

12.5 Relational Learning

In this section we discuss some first-order models developed from a machine learning perspective.

Machine learning algorithms have traditionally been defined as classification of *attribute-value vectors* [74]. In many applications, it is more natural and convenient to represent data as *graphs*, where each vertex represents an object and each edge represents a relation between objects. Vertices can be labeled with attributes of its corresponding object (unary predicates or binary predicates where the second argument is a simple value; this is similar to PRMs in Sect. 12.4.4), and edges can be labeled (the label can be interpreted as a binary predicate holding between the objects). This provides the typical data structure representations such as trees, lists and collections of objects in general. When learning from graphs, we usually want to form hypotheses that explain one or more of the attributes and (or) relations (the *targets*) of objects in terms of its neighbors. Machine learning algorithms which were developed to benefit from this type of representation have often been called *relational*. This is closely associated to probabilistic first-order models, since graph data can be interpreted as a set of ground literals using unary and binary predicates. Because the hypotheses explaining target attributes and relations apply to several objects, it is also convenient to represent the learned hypotheses as quantified (first-order) rules. And because most learners involve probabilities or at least some measure of uncertainty, probabilistic first-order rules provide a natural representation option. Figure 12.5 illustrates these concepts.

We now discuss three forms of relational learning: propositionalization (flattening), Inductive Logic Programming (ILP), and FOPI learning, which can be seen as a synthesis of the two.

12.5.1 Propositionalization

A possible approach to relational machine learning is that of using a relational structure for generating propositional attribute-value vectors for each of its objects. For this reason, the approach has been called *propositionalization*. Because it transforms graph-like data into vector-like data, it is also often called *flattening*.

Cumby & Roth [75] provide a language for transforming relational data into attribute-value vectors. Their concern is not forming a first-order hypothesis,

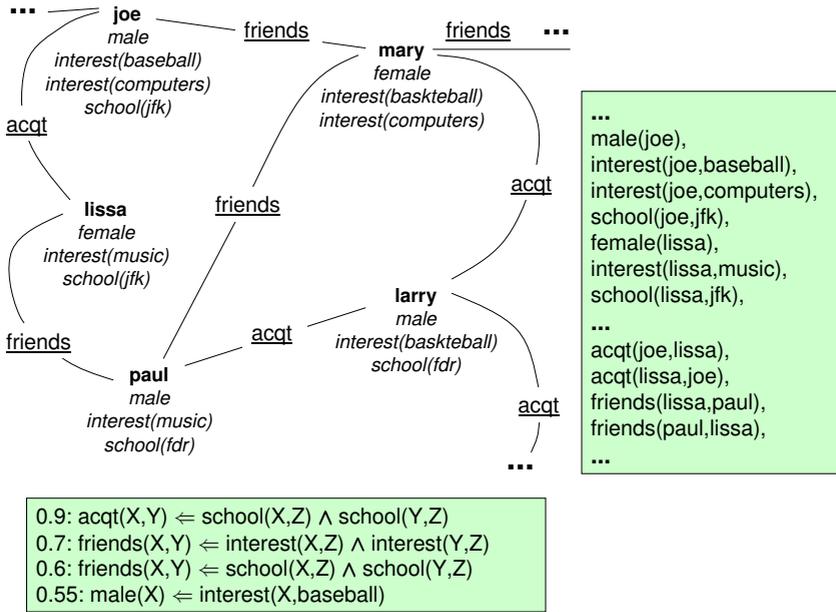


Fig. 12.5. A fragment of a graph structure used as input for relational learning. The same information can be represented as a set of ground literals (right). The hypotheses learned to explain either relations or attributes can be represented as weighted first-order clauses over those literals (below).

however. They instead keep the attribute-value hypothesis and transform novel data to that representation in order to classify it with propositional learners such as Perceptron. For example, in the case of Fig. 12.5, a classifier seeking to learn the relation *acqt* would go through the instances of that predicate and generate suitable attribute-value vectors. The literal *acqt(paul, larry)* would generate an example with label *acqt(X, Y)* and features *male(paul)*, *male(larry)*, *interest(paul, music)*, *school(paul, fdr)*, *interest(larry, basketball)*, *school(larry, fdr)* etc, as well as non-ground ones such as *male(X)*, *male(Y)*, *school(X, fdr)*, *school(Y, fdr)*, *school(X, Z)*, *school(Y, Z)*, *interest(X, music)* etc. The literal *acqt(joe, lissa)* would generate an example with label *acqt(X, Y)* and features *male(joe)*, *female(lissa)*, *interest(joe, baseball)*, *interest(joe, computers)*, *school(joe, jfk)*, etc, as well as non-ground ones such as *male(X)*, *female(Y)*, *school(X, jfk)*, *school(Y, jfk)*, *school(X, Z)*, *school(Y, Z)* etc. Note how this reveals abstractions – the examples above share the features *school(X, Z)* and *school(Y, Z)*, which may be one reason for people being acquaintances in this domain. Should a target depend on specific objects (say, it is much more likely for people at the FDR school to be acquainted to each other) not completely abstracted features such as *school(X, fdr)* would be preferred by the classifier.

There are many different ways of transforming a graph or set of literals into attribute-value vectors for propositional learners. Each of them will represent different learning biases. Some works in this line are LINUS [76], which uses the concept of *ij*-determinacy (a way of restricting the generalizations of literals) in order to construct hypothesis, 1BC [77] and 1BC2 [78], which differentiate between predicates representing attributes or relations and construct multiset attributes (for example, the set of interests among one's friends), and Relational Bayesian Classifier (RBC) [79], which also uses multiset attribute values with a conditional independence assumption similar to the one used in the Naive Bayes classifier.

One disadvantage of propositionalization is that it performs classification of an object at a time. This prevents the use of possible dependencies between object labels and introduces a bias. These dependencies can be used by FOPI algorithms, which perform *joint* inference over several objects at once. Three of FOPI models which have been specifically developed with this in mind are RDNs [65], RMNs [64] and MLNs [70].

12.5.2 Inductive Logic Programming

Inductive Logic Programming (ILP) stemmed from the logic programming community with the goal of learning logic programs from data rather than writing them. The choice of logic programming as a hypothesis language initially restricted the field to deterministic hypotheses; the later incorporation of probabilities to this framework is one of the origins of FOPI models. However, the fact that these algorithms learn from data give them a statistical flavor even in the deterministic case. For example, Progol [36] uses information-theoretic measures for evaluating deterministic hypotheses.

A typical ILP algorithm works by forming hypotheses one clause at a time. For example, if such an algorithm is trying to learn the concept $mother(X, Y)$, it might generate the clause $mother(X, Y) : \neg female(X)$, since $female(X)$ does add some predictive power to whether X is a mother, and may at a latter step refine it to $mother(X, Y) : \neg female(X), child(Y, X)$. This is a top-down approach since the most general clause is successively made more specific according to examples. Two examples of work in this line are [80, 81]. Another approach is bottom-up, best exemplified by Progol [36], where sets of ground literals are successively generalized in order to increase accuracy.

Some ILP algorithms use propositionalization as a subroutine that learns one rule at a time. LINUS [76] transforms its data into attribute-value vectors, applies regular attribute-value learners to it, and then transforms the attribute-value hypothesis into a clause.

Probabilistic ILP (PILP) algorithms (a specific survey can be found at [82]) also often grow clauses and then estimate the probabilities (or parameters) associated with those clauses. Some learn an entire logic program at first, and then the parameters, while others learn the parameters as soon as the clauses are learned. In fact, PILP is simply FOPI with learning done with ILP techniques, and many of these works have been mentioned in previous sections. For example,

MLNs [70], PRMs [58, 59] and BLPs [83] learn the structure of their models (as opposed to their parameters) through techniques similar to those of ILP. RDNs [65] use relational classifiers developed in the ILP community as a subroutine to its model learning.

12.6 Restricted First-Order Probabilistic Models

The models presented so far intended to provide a level of expressivity similar to first-order logic. At a minimum, they provide unary and binary predicates arbitrarily applied to a collection of objects. However, there are some probabilistic models exhibiting a restricted subset of first-order aspects, targeted for particular tasks.

One such model is Hidden Markov Models (HMMs) [84], in which a sequence of pairs of random variables represent a *state* and an *observation*. The model relates observations to the state in the same time slice, as well as states in successive time slides. While essentially propositional, HMMs exhibit the sharing of parameters commonly observed in first-order models, since its parameters equally apply to all transitions from one step to the next. The time indices of random variables of an HMM are a restricted form of treating them as first-order.

The same sharing of parameters can be observed in related models. Stochastic Context-Free Grammars [31] consist of set of production rules were a non-terminal symbol is stochastically replaced by a number of possible sequences of symbols. Again, the rules can be applied at different points and parameters are reused. Dynamic Bayesian Networks [85] generalize HMMs in that each step is represented by a full Bayesian network rather than just a pair of state and observations.

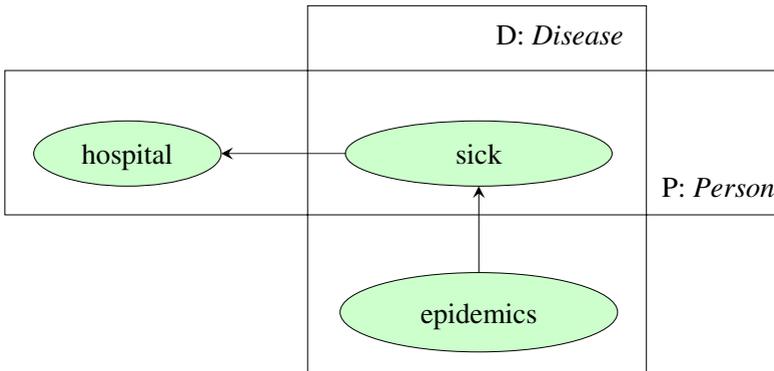


Fig. 12.6. Repeated structure of a graphical model can be indicated by plates. Random variables are implicitly indexed by integer variables associated with the plates inside which they reside. The same CPT is used for all of them.

Other generalizations of these restricted models are Hidden Tree Markov Models [86], where possible states form a tree structure, Logical Hidden Markov Models [66], where each state is represented by a set of logical literals, and Relational Markov Models [87], where each state is also represented by a set of logical literals, but possible arguments follow a taxonomy and induce a lattice on possible literals.

Another simple generalization of propositional models is the plate notation [88], which describes graphical models with parts replicated by a set of indices, indicated by a rectangle involving that part in a diagram. An example is shown in Fig. 12.6. The parameters into these parts are then also replicated. Mjolsness [89] proposes many refinements to this type of notation. The plate notation is commonly used as a tool for descriptions in the literature but has not been typically meant as an input to algorithms.

12.7 Conclusion

First-order probabilistic inference has made much progress in the last twenty years. We believe the main accomplishments have been the clarification the semantics of such languages, as well as a greater understanding on how several different language options relate to each other. In analyzing the works in the area of FOPI, we can distinguish a few main options along which they seem to organize themselves. We now make these aspects more explicit.

12.7.1 Directed vs. Undirected

The decision on using directed or undirected models carries over the the first-order case. While the intelligibility of directed models has favored their use in first-order proposals at first, we can observe a current tendency to use undirected models [70, 90], or at least directed models without the acyclicity requirement [65]. The reason for this shift is that cycles are even more naturally occurring in first-order domains than in propositional ones. A “natural” conditional probability such as $P(\text{smoker}(X) | \text{friend}(X, Y), \text{smoker}(Y))$ creates an underlying network with cycles. The use of undirected models seems to be further justified by the fact that they do not rule out directed models. If the given factors encode conditional probabilities that do not involve cycles, they will still represent the correct distribution even if interpreted as an undirected factor (this however loses structure that could be used to improve efficiency).

12.7.2 The Trade-Off between Language and Algorithm

Some FOPI proposals focus on rich languages that allow the user to indicate domain restrictions which can be exploited for efficiency. Examples of such systems are Ngo and Haddawy’s Probabilistic Logic Programming [23], PRMs [58, 59], LBNs [67]. One example is the treatment of determinism (especially of relations)

being expressed explicitly by the language, as in the case of PRMs and LBNs, as discussed in Sect. 12.4.4.

Other solutions propose simple languages, relying on inference algorithms to exploit domain structure (sometimes guided by extra-language directives). The most typical examples are BLPs [66] and MLNs [70].

This choice reflects a trade-off between language and inference algorithm complexities. Complex languages bring built-in optimization hints; for example, PRMs have attributes (with a value) and relations, even though both could be regarded as binary predicates. By indicating that a binary predicate is an attribute, the user implicitly indicates the most efficient ways of using it, which are distinct from the way a relation is used. To obtain the same effect, an algorithm using only a single general notion of binary predicates would have to either find out or be told (by extra-language directives) how to use each of them in the most efficient manner. In this case, compilation and learning can play important roles.

Our particular view is that FOPI languages will tend to become simpler, leaving the complexities to be dealt with by compilation, learning and directives. This reflects the evolution of programming languages, that have increasingly left efficiency details to be dealt with by compilers and directives rather than by the language itself, leaving the latter at a higher level that can be more easily understood and theoretically related to other approaches. On the other hand, this simplicity should not be such as to prevent the development of specialized libraries and knowledge representations. These are useful but best included on top of simple primitives instead of as primitives themselves.

12.7.3 Infinite Models

First-order models may have a finite description while involving an infinite number of random variables, if the number of objects in the domain is infinite. This poses problems to algorithms presented in this survey, since they may not stop in that case. As pointed out by [91], models with infinite number of random variables can be dealt with by approximate, anytime algorithms. These algorithms will provide arbitrarily precise approximations after a sufficient (but finite) amount of computation is performed. Such an approach would also make sense for processing models that, while involving a finite number of random variables, are too complex for exact inference. Such algorithms would also benefit from *guided* evaluation, that is, by choosing to process first the parts of the model that yield greater amounts of information about the query.

References

1. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice-Hall, Englewood Cliffs (2003)
2. Buchanan, B., Shortliffe, E.: Rule-Based Expert Systems: The MYCIN experiments of the Stanford Heuristic Programming Project. Addison-Wesley, Reading (1984)
3. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, San Mateo (Calif.) (1988)

4. Nilsson, N.J.: Probabilistic logic. *Artificial Intelligence* 28(1), 71–88 (1986)
5. Bacchus, F.: Representing and reasoning with probabilistic knowledge: a logical approach to probabilities. MIT Press, Cambridge (1990)
6. Halpern, J.Y.: An analysis of first-order logics of probability. In: *Proceedings of IJCAI 1989, 11th International Joint Conference on Artificial Intelligence*, Detroit, US, pp. 1375–1381 (1990)
7. Shortliffe, E.H.: Mycin: a rule-based computer program for advising physicians regarding antimicrobial therapy selection. PhD thesis, Stanford University (1975)
8. Clark, K.L., McCabe, F.G.: Prolog: A language for implementing expert systems. In: Hayes, J.E., Michie, D., Pao, Y.H. (eds.) *Machine Intelligence*, Ellis Horwood, Chichester, vol. 10, pp. 455–470 (1982)
9. Shapiro, E.: Logic programs with uncertainties: A tool for implementing expert systems. In: *Proc. IJCAI 1983*, pp. 529–532. William Kaufmann, San Francisco (1983)
10. Heckerman, D.: Probabilistic interpretation for MYCIN's certainty factors. In: Kanal, L.N., Lemmer, J. (eds.) *Uncertainty in Artificial Intelligence*, pp. 167–196. Kluwer Science Publishers, Dordrecht (1986)
11. Lucas, P.: Certainty-factor-like structures in bayesian belief networks. *Knowledge-Based Systems* 14, 325–327 (2001)
12. Carnap, R.: *The Logical Foundations of Probability*. University of Chicago Press, Chicago (1950)
13. Hailperin, T.: Probabilistic logic. *Notre Dame Journal of Formal Logic* 25(3), 198–212 (1984)
14. Fagin, R., Halpern, J.Y., Megiddo, N.: A logic for reasoning about probabilities. *Information and Computation* 87(1/2), 78–128 (1990)
15. Fenstad, J.E.: The structure of probabilities defined on first-order languages. *Studies in Inductive Logic and Probabilities*, pp. 251–262. California Press (1980)
16. Gaifman, H.: Concerning measures in first-order calculi. *Israel Journal of Mathematics* 2, 1–18 (1964)
17. Gaifman, H., Snir, M.: Probabilities over rich languages, testing and randomness. *Journal of Symbolic Logic* 47(3), 495–548 (1982)
18. Kifer, M., Li, A.: On the semantics of rule-based expert systems with uncertainty. In: Gyssens, M., Van Gucht, D., Paredaens, J. (eds.) *ICDT 1988*. LNCS, vol. 326, pp. 102–117. Springer, Heidelberg (1988)
19. Wüthrich, B.: Probabilistic knowledge bases. *IEEE Trans. Knowl. Data Eng.* 7(5), 691–698 (1995)
20. Ng, R.T., Subrahmanian, V.S.: Probabilistic logic programming. *Information and Computation* 101(2), 150–201 (1992)
21. Lakshmanan, L.V.S., Sadri, F.: Probabilistic deductive databases. In: *Symposium on Logic Programming*, pp. 254–268 (1994)
22. Lakshmanan, L.V.S.: An epistemic foundation for logic programming with uncertainty. In: *Foundations of Software Technology and Theoretical Computer Science*, pp. 89–100 (1994)
23. Ngo, L., Haddawy, P.: Probabilistic logic programming and Bayesian networks. In: *Asian Computing Science Conference*, pp. 286–300 (1995)
24. Poole, D.: Probabilistic Horn abduction and Bayesian networks. *Artificial Intelligence* 64(1), 81–129 (1993)
25. Zhang, N.L., Poole, D.: A simple approach to Bayesian network computations. In: *Proceedings of the Tenth Biennial Canadian Artificial Intelligence Conference* (1994)

26. Lukasiewicz, T.: Probabilistic deduction with conditional constraints over basic events. *J. Artif. Intell. Res (JAIR)* 10, 199–241 (1999)
27. Frisch, A.M., Haddawy, P.: Anytime deduction for probabilistic logic. *Artificial Intelligence* 69(1–2), 93–122 (1994)
28. Koller, D., Halpern, J.Y.: Irrelevance and conditioning in first-order probabilistic logic. In: Shrobe, H., Senator, T. (eds.) *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*, vol. 2, pp. 569–576. AAAI Press, Menlo Park (1996)
29. Riezler, S.: Probabilistic constraint logic programming (1997)
30. Jaffar, J., Lassez, J.L.: Constraint logic programming. In: *POPL 1987: Proceedings of the 14th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, pp. 111–119. ACM Press, New York (1987)
31. Lari, K., Young, S.: The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language* 4, 35–56 (1990)
32. Baker, J.: Trainable grammars for speech recognition. In: *Speech communication papers presented at the 97th Meeting of the Acoustical Society*, pp. 547–550 (1979)
33. Baum, L.E.: An inequality and associated maximization technique in statistical estimation for probabilistic functions of markov processes. *Inequalities* 3, 1–8 (1972)
34. Muggleton, S.: Stochastic logic programs. In: De Raedt, L. (ed.) *Proceedings of the 5th International Workshop on Inductive Logic Programming*, Department of Computer Science, Katholieke Universiteit Leuven, vol. 29 (1995)
35. Cussens, J.: Loglinear models for first-order probabilistic reasoning. In: Laskey, K.B., Prade, H. (eds.) *Proceedings of the 15th Annual Conference on Uncertainty in AI (UAI 1999)*, pp. 126–133. Morgan Kaufmann, San Francisco (1999)
36. Muggleton, S.: Inverse entailment and Progol. *New Generation Computing*, Special issue on Inductive Logic Programming 13(3-4), 245–286 (1995)
37. Della Pietra, S., Della Pietra, V.J., Lafferty, J.D.: Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(4), 380–393 (1997)
38. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 39(1), 1–38 (1977)
39. Lukasiewicz, T.: Probabilistic logic programming. In: *European Conference on Artificial Intelligence*, pp. 388–392 (1998)
40. Baral, C., Gelfond, M., Rushton, J.N.: Probabilistic reasoning with answer sets. In: *LPNMR*, pp. 21–33 (2004)
41. Pearl, J.: *Causality: Models, Reasoning and Inference*. Cambridge University Press, Cambridge (2000)
42. Sato, T., Kameya, Y.: Prism: A language for symbolic-statistical modeling. In: *IJCAI*, pp. 1330–1339 (1997)
43. Sato, T., Kameya, Y.: A viterbi-like algorithm and em learning for statistical abduction (2000)
44. Milch, B., Marthi, B., Russell, S., Sontag, D., Ong, D.L., Kolobov, A.: BLOG: Probabilistic models with unknown objects. In: *Proc. IJCAI* (2005)
45. Spiegelhalter, D., Thomas, A., Best, N., Gilks, W.: Bugs: Bayesian inference using gibbs sampling, version 0.30. Technical report, MRC Biostatistics Unit, University of Cambridge (1994)
46. Heckerman, D., Chickering, D.M., Meek, C., Rounthwaite, R., Kadie, C.M.: Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research* 1, 49–75 (2000)

47. de Salvo Braz, R.: Lifted First-Order Probabilistic Inference. PhD thesis, University of Illinois at Urbana-Champaign (2007)
48. de Salvo Braz, R., Amir, E., Roth, D.: Lifted first-order probabilistic inference. In: Getoor, L., Taskar, B. (eds.) *An Introduction to Statistical Relational Learning*, pp. 433–451. MIT Press, Cambridge (2007)
49. Breese, J.S.: Construction of belief and decision networks. *Computational Intelligence* 8, 624–647 (1991)
50. Wellman, M.P., Breese, J.S., Goldman, R.P.: From knowledge bases to decision models. *Knowledge Engineering Review* 7, 35–53 (1992)
51. Horsch, M., Poole, D.: A dynamic approach to probabilistic inference using bayesian networks. In: *Proceedings of the 6th Conference of Uncertainty in Artificial Intelligence*, pp. 155–161. Morgan Kaufmann, San Francisco (1990)
52. Goldman, R.P., Cherniak, E.: A language for construction of belief networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 15(3), 196–208 (1993)
53. Poole, D.: The independent choice logic for modelling multiple agents under uncertainty. *Artificial Intelligence* 94(1-2), 7–56 (1997)
54. Glesner, S., Koller, D.: Constructing flexible dynamic belief networks from first-order probabilistic knowledge bases. In: *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pp. 217–226 (1995)
55. Haddawy, P.: Generating bayesian networks from probability logic knowledge bases. In: de Mantaras, R.L., Poole, D. (eds.) *Uncertainty In Artificial Intelligence 10 (UAI 1994)*, pp. 262–269. Morgan Kaufmann, San Francisco (1994)
56. Koller, D., Pfeffer, A.: Learning probabilities for noisy first-order rules. In: *IJCAI*, pp. 1316–1323 (1997)
57. Jaeger, M.: Relational Bayesian networks. In: Kaufmann, M. (ed.) *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence*, pp. 266–273 (1997)
58. Getoor, L., Friedman, N., Koller, D., Pfeffer, A.: Learning probabilistic relational models. In: Džeroski, S., Lavrac, N. (eds.) *Relational Data Mining*, pp. 307–335. Springer, Heidelberg (2001)
59. Koller, D., Pfeffer, A.: Probabilistic frame-based systems. In: *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI)*, pp. 580–587 (1998)
60. Minsky, M.: A framework for representing knowledge. In: *Computation & intelligence: collected readings*, pp. 163–189. American Association for Artificial Intelligence, Menlo Park (1995)
61. Levesque, H.J., Brachman, R.J.: Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence* 3, 78–93 (1987)
62. Koller, D., Levy, A.Y., Pfeffer, A.: P-CLASSIC: A tractable probabilistic description logic. In: *AAAI/IAAI*, pp. 390–397 (1997)
63. Getoor, L.: Learning probabilistic relational models with structural uncertainty. In: Choueiry, B.Y., Walsh, T. (eds.) *SARA 2000. LNCS (LNAI)*, vol. 1864, pp. 322–329. Springer, Heidelberg (2000)
64. Taskar, B., Abbeel, P., Koller, D.: Discriminative probabilistic models for relational data. In: *Proc. Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, Edmonton, Canada (2002)
65. Neville, J.: Statistical models and analysis techniques for learning in relational data. PhD thesis, University of Massachusetts Amherst (2006)
66. Kersting, K., De Raedt, L.: Bayesian logic programs. In: Cussens, J., Frisch, A. (eds.) *Proceedings of the Work-in-Progress Track at the 10th International Conference on Inductive Logic Programming*, pp. 138–155 (2000)
67. Fierens, D., Blockeel, H., Bruynooghe, M., Ramon, J.: Logical bayesian networks and their relation to other probabilistic logical models. In: *ILP*, pp. 121–135 (2005)

68. Santos Costa, V., Page, D., Qazi, M., Cussens, J.: Clp(bn): Constraint logic programming for probabilistic knowledge. In: Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence (UAI 2003), pp. 517–552. Morgan Kaufmann, San Francisco (2003)
69. Angelopoulos, N.: Probabilistic finite domains: A brief overview. In: Stuckey, P.J. (ed.) ICLP 2002. LNCS, vol. 2401, p. 475. Springer, Heidelberg (2002)
70. Richardson, M., Domingos, P.: Markov logic networks. Technical report, Department of Computer Science, University of Washington (2004)
71. Besag, J.: Statistical analysis of non-lattice data. *The Statistician* 24(3), 179–195 (1975)
72. Laskey, K.B.: First-order Bayesian logic. Technical report, George Mason University Department of Systems Engineering and Operations Research (2005)
73. Poole, D.: First-order probabilistic inference. In: Proceedings of the 18th International Joint Conference on Artificial Intelligence, pp. 985–991 (2003)
74. Mitchell, T.M.: *Machine Learning*. McGraw-Hill Higher Education, New York (1997)
75. Cumby, C., Roth, D.: Relational representations that facilitate learning. In: Cohn, A.G., Giunchiglia, F., Selman, B. (eds.) KR 2000: Principles of Knowledge Representation and Reasoning, pp. 425–434. Morgan Kaufmann, San Francisco (2000)
76. Lavrac, N., Dzeroski, S.: *Inductive Logic Programming: Techniques and Applications*. Routledge, New York, 10001 (1993)
77. Flach, P., Lachiche, N.: 1BC: A first-order Bayesian classifier. In: Dzeroski, S., Flach, P. (eds.) ILP 1999. LNCS (LNAI), vol. 1634, pp. 92–103. Springer, Heidelberg (1999)
78. Lachiche, N., Flach, P.A.: 1BC2: a true first-order Bayesian classifier. In: Matwin, S., Sammut, C. (eds.) ILP 2002. LNCS (LNAI), vol. 2583, pp. 133–148. Springer, Heidelberg (2003)
79. Neville, J., Jensen, D., Gallagher, B.: Simple estimators for relational bayesian classifiers. In: ICDM 2003: Proceedings of the Third IEEE International Conference on Data Mining. IEEE Computer Society Press, Washington (2003)
80. Quinlan, J.: Learning logical definitions from relations. *Machine Learning* 5, 239–266 (1990)
81. Raedt, L.D., Dehaspe, L.: Clausal discovery. *Machine Learning* 26(2-3), 99–146 (1997)
82. Raedt, L.D., Kersting, K.: Probabilistic inductive logic programming. In: ALT, pp. 19–36 (2004)
83. Kersting, K., De Raedt, L.: Towards combining inductive logic programming with Bayesian networks. In: Rouveirol, C., Sebag, M. (eds.) ILP 2001. LNCS (LNAI), vol. 2157, pp. 104–117. Springer, Heidelberg (2001)
84. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. In: *Readings in speech recognition*, pp. 267–296. Morgan Kaufmann Publishers Inc., San Francisco (1990)
85. Murphy, K.P.: *Dynamic bayesian networks: representation, inference and learning*. PhD thesis, University of California Berkeley, Chair-Stuart Russell (2002)
86. Diligenti, M., Frasconi, P., Gori, M.: Hidden tree markov models for document image classification. *IEEE Trans. Pattern Anal. Mach. Intell.* 25(4), 519–523 (2003)
87. Anderson, C.R., Domingos, P., Weld, D.S.: Relational Markov models and their application to adaptive web navigation. In: KDD 2002: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 143–152. ACM Press, New York (2002)

88. Buntine, W.L.: Operations for learning with graphical models. *Journal of Artificial Intelligence Research* 2, 159–225 (1994)
89. Mjolsness, E.: Labeled graph notations for graphical models: Extended report. Technical report, University of California Irvine – Information and Computer Sciences (2004)
90. de Salvo Braz, R., Amir, E., Roth, D.: Lifted first-order probabilistic inference. In: *Proceedings of IJCAI 2005, 19th International Joint Conference on Artificial Intelligence (2005)*
91. Pfeffer, A., Koller, D.: Semantics and inference for recursive probability models. In: *AAAI/IAAI*, pp. 538–544 (2000)

Author Index

- Amir, Eyal 289
- Cooper, Gregory F. 169
- Daryle Niedermayer, I.S.P. 117
- de Salvo Braz, Rodrigo 289
- Flores, M. Julia 251
- Gámez, José A. 251
- Heckerman, David 33
- Holmes, Dawn E. 1, 281
- Jain, Lakhmi C. 1
- Jiang, Xia 169
- Korb, Kevin B. 83
- Lauría, Eitel J.M. 187
- Leray, Philippe 219
- Maes, Sam 219
- Meganck, Stijn 219
- Moral, Serafín 251
- Nagl, Sylvia 131
- Neapolitan, Richard E. 7
- Nicholson, Ann E. 83
- Roth, Dan 289
- Wagner, Michael M. 169
- Williams, Matt 131
- Williamson, Jon 131

Editors



Originally from the UK, Professor D. E. Holmes is now a permanent member of Faculty in the Department of Applied Probability and Statistics at UCSB, California. Her research interests include Artificial Intelligence (the use of the maximum entropy formalism in Bayesian networks), Intuitionist Mathematics (Brouwers programme, intuitionist Markov chains) and some philosophical aspects of Bayesianism.



Professor L.C. Jain is a Director/Founder of the Knowledge-Based Intelligent Engineering Systems (KES) Centre, located in the University of South Australia.

His interests focus on the applications of novel techniques such as knowledge-based systems, virtual intelligent systems, defence systems, intelligence-based medical systems, e-Education and intelligent agents.